Reescritura con Restricciones de Reemplazamiento

Salvador Lucas Alba
Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia

Memoria presentada para optar al título de:
Doctor en Informática

Dirigida por:
María Alpuente Frasnedo

Tribunal de lectura:

Presidente:	**Isidro Ramos Salavert**	**U.P. Valencia**
Secretario :	**Mario Rodríguez Artalejo**	**U.C. Madrid**
Vocales:	**Michael Hanus**	**RWTH Aachen**
	Marisa Navarro	**U. País Vasco**
	Robert Nieuwenhuis	**U.P. Catalunya**

Octubre de 1998

Resumen

La reescritura de términos constituye un mecanismo operacional útil para implementar un buen número de sistemas computacionales cuyo modelo de ejecución se basa en reducir una expresión inicial hasta obtener otra expresión con un contenido semántico bien definido y suficiente para los fines perseguidos.

Para conseguir este objetivo computacional es necesario, en la práctica, *restringir* el espacio de reducciones asociado a la expresión inicial, filtrando las secuencias de reescritura que no conducen a una forma canónica de interés. Esta optimización se puede realizar introduciendo *restricciones de reemplazamiento sintácticas* vinculadas a los términos que participan en las secuencias de reducción. En este trabajo formulamos y caracterizamos una forma de cómputo que explota este tipo de restricciones de la reescritura y que denominamos *reescritura sensible al contexto*.

Empezamos formulando la teoría de la reescritura sensible al contexto y demostramos que conduce a una optimización del mecanismo de la reescritura que permite computar una gran variedad de formas canónicas evitando muchas secuencias de reducción no deseables y preservando aún la completitud de los cálculos. Para mostrar la viabilidad de esta propuesta, estudiamos las propiedades de terminación, confluencia y capacidad de cómputo de la reescritura sensible al contexto y formalizamos diversas estrategias que permiten realizar dichos cómputos de forma eficiente. Presentamos algoritmos y técnicas de transformación de programas que permiten optimizar la relación existente entre la mejora obtenida del comportamiento de terminación de los programas (al restringir con más severidad los reemplazamientos) y las condiciones a imponer para preservar la completitud de los cómputos.

Finalmente, exploramos también la idea de las restricciones de reemplazamiento más allá del marco de la reescritura estándar, investigando su aplicabilidad en el campo del estrechamiento y de la reescritura condicional. Asimismo, presentamos un marco de descripción y análisis de las restricciones de reemplazamiento que permite emplear los resultados obtenidos para la reescritura sensible al contexto como aproximación a las propiedades computacionales de otras clases de restricciones de reemplazamiento más generales.

Agradecimientos

Quisiera aprovechar este espacio para hacer presentes algunas de las muchas personas que, de una forma u otra, han sostenido, con su apoyo, consejo, presencia o discreto silencio, el trabajo que hoy se concreta en esta tesis.

Agradezco a María Alpuente su ayuda constante, incansable y siempre acertada a la hora de orientar, corregir y depurar las ideas y esfuerzos vertidos durante estos años. Además de su apoyo científico, le agradezco profundamente su apoyo personal, desde el momento en que me brindó la oportunidad de integrarme en el grupo, pasando por su ejemplo diario de entusiasmo, seriedad y entrega, y terminando por la confianza con que me ha honrado al alentar la realización y presentación de este trabajo.

A Javier Oliver debo expresarle mi sincera gratitud por proponerme el proyecto fin de carrera del cual surgió la idea central de esta tesis y por su cercanía, simpatía y disponibilidad permanentes.

También quisiera agradecer a Germán Vidal todo el interés que se ha tomado siempre por ayudarme en aquello que he necesitado, desde el principio de mi llegada al grupo hasta el presente.

A María José Ramírez le doy las gracias por su presencia siempre amable, su ejemplo constante, y por animarme a profundizar en este tema.

A Michael Hanus le agradezco el haberme permitido completar mi formación en el RWTH Aachen, tanto mediante la creación de un ambiente de trabajo dinámico y fértil como con sus consejos e indicaciones.

A Javier Piris, Asunción Casanova, Pascual Julián, Vicente Gisbert, Frank Zartmann, Tomás Pérez y Vicente Fuentes les agradezco su entusiasmo y participación en distintas discusiones sobre cuestiones relacionadas en mayor o menor medida con esta investigación.

En general, a todos los miembros del grupo de Extensiones de la Programación Lógica, del Departamento de Sistemas Informáticos y Computación de la UPV y de la Lehr- und Forschungsgebiet für Informatik II del RWTH Aachen, por haber contribuido a crear un ambiente humano y científico adecuado.

No quisiera dejar pasar esta ocasión sin agradecer a mi familia, y en especial a mi madre, todo el cariño, comprensión y afecto con que han sembrado mi vida, no sólo durante los años de trabajo en esta tesis sino, mucho más importante, durante todo el tiempo que hemos pasado juntos. En particular, quiero agradecer a mi hermano Toni, con quien tantas vivencias he compartido, su ejemplo de honestidad intelectual y científica y su entusiasmo sin tregua. También a Amparo, Jero, Cristina, Juan y Alberto por participar en esas discusiones a las que somos tan aficionados y donde la vida, y no sólo las ideas, es lo más importante.

Por último, a mi esposa Sandra le debo mucho más que agradecimiento y mucho menos por esta tesis que por estar siempre a mi lado, con todas las consecuencias, en todo aquello que es importante para mí. Espero saber corresponderte siempre así.

A Sandra,
a Paula,
a mis padres,
a mis hermanos,
a mi familia.

Siendo joven aún, antes de ir por el mundo,
me di a buscar abiertamente la sabiduría en mi oración.
A la puerta delante del templo la pedí
y hasta mi último día la andaré buscando.

Si 51, 13-14

Índice General

1 Introducción **1**

1.1 Restricciones de reemplazamiento sintácticas 3

1.2 Objetivo y estructura de la tesis 8

2 Preliminares **15**

2.1 Definiciones de carácter general 15

2.1.1 Funciones y tuplas 16

2.1.2 Relaciones binarias 17

2.1.3 Secuencias infinitas 18

2.1.4 Conjuntos ordenados 19

2.2 Signaturas y términos 20

2.2.1 Ocurrencias 21

2.2.2 Sustituciones 22

2.2.3 Términos y ocurrencias 22

2.3 Sistemas de reescritura 24

2.3.1 Clases de sistemas de reescritura 26

2.3.2 Formas canónicas en reescritura 27

2.3.3 Multiderivaciones 28

2.3.4 Reescritura infinita 29

2.3.5 Terminación y confluencia 30

2.4 Estrategias de reescritura 31

2.4.1 Estrategias de reducción infinita 34

2.5 Reducciones necesarias 35

2.5.1 Ω-reducción 37

2.5.2 Aproximación de formas estables 38

2.5.3 Índices fuertemente secuenciales 40

2.5.4 Sistemas fuertemente secuenciales 41

I Propiedades computacionales básicas de la reescritura sensible al contexto **43**

3 La reescritura sensible al contexto **45**
3.1 Restricciones de reemplazamiento 45
3.2 Propiedades de las restricciones 49
3.2.1 Restricciones y reducción 52
3.3 Reescritura sensible al contexto 52
3.4 Algunos ejemplos de aplicación 56
3.5 Trabajos relacionados 60

4 Reescritura y *rsc* **65**
4.1 Restricciones en el proceso de reescritura 65
4.2 Reescritura sensible al contexto y reescritura 70
4.3 Caracterización de las formas μ-normales 77
4.4 Trabajos relacionados 81

5 Terminación **83**
5.1 Análisis de μ-terminación 84
5.2 La transformación contractiva 86
5.3 Contracción de sistemas de reescritura 89
5.4 μ-terminación y μ-contracción 92
5.5 Demostración de la μ-terminación 94
5.6 Trabajos relacionados 97
5.7 Ω-reducción con $\mathcal{R}$ y $\mathcal{R}^{\mu}$ 98
5.7.1 Aplicación a la Ω-reducción 101

6 Confluencia **105**
6.1 Condiciones sintácticas básicas de μ-confluencia 106
6.2 Confluencia local 109
6.3 Confluencia fuerte 111
6.4 μ-confluencia 115
6.5 Trabajos relacionados 116

II Estrategias de reducción sensible al contexto **119**

7 Estrategias sensibles al contexto **121**
7.1 Concepto de estrategia sensible al contexto 122
7.2 Cómputos necesarios para estabilización y *rsc* 124
7.3 Índices y estabilización 127

7.4 μ-estrategias y reducción de índices 133
7.5 μ-estrategias de reducción de índices 137
7.6 Trabajos relacionados 141

8 Estrategias genéricas **143**
8.1 Concepto de estrategia genérica 144
8.1.1 Definición de estrategias genéricas 146
8.1.2 Estrategias genéricas sensibles al contexto 149
8.2 Secuencialidad fuerte y contracción de SRTs 151
8.2.1 μ-contracción de SRTEs y decidibilidad de la secuencialidad fuerte 153
8.2.2 Extensión de clases de SRTs fuertemente secuenciales 155
8.2.3 Estrategias *sc* inducidas 158
8.3 Trabajos relacionados 161

9 Implementación **163**
9.1 Implementación de estrategias de reescritura 163
9.2 Index trees 166
9.3 Implementación de μ-estrategias inducidas 167
9.4 Trabajos relacionados 171

10 Ejemplos de aplicación **173**
10.1 Extensión de los SRTs normales por la izquierda 173
10.1.1 Caracterización sintáctica 174
10.1.2 La estrategia inducida 175
10.2 Extensión de los SRTs básicos 177
10.2.1 Caracterización sintáctica 177
10.2.2 La estrategia inducida 178
10.3 SRTs forward-branching 180
10.4 Trabajos relacionados 182

III Formas canónicas y reescritura sensible al contexto **185**

11 Valores y formas estables constructoras **187**
11.1 Obtención directa de valores 188
11.1.1 *Strictness* y completitud de los cómputos con *rsc* 191
11.2 Evaluación eficiente por transformación 192
11.3 Símbolos internos y externos 195
11.4 Aproximación de símbolos definidos externos 197
11.5 Corrección de la aproximación 202
11.6 La transformación V 205

11.7 Corrección y completitud 211
11.8 Trabajos relacionados 217

12 Formas normales y estabilización **219**
12.1 Estabilización y normalización directa 219
12.2 El problema de la normalización 222
12.3 Formas normales y formas μ-normales 223
12.4 Algoritmos de normalización 225
12.5 Trabajos relacionados 229

13 Formas normales infinitas **231**
13.1 Normalización infinita 232
13.2 Estrategias de normalización infinita 233
13.3 Trabajos relacionados 234

IV Extensiones. Restricciones de reemplazamiento generales **235**

14 Extensiones **237**
14.1 Estrechamiento 237
14.1.1 Estrechamiento sensible al contexto 238
14.1.2 Resolución de ecuaciones 240
14.2 SRTs condicionales 242
14.2.1 Reescritura condicional sensible al contexto 245
14.3 Trabajos relacionados 249

15 Restricciones de reemplazamiento generales **251**
15.1 Restricciones de reemplazamiento en reescritura 252
15.2 Descripción de restricciones de reemplazamiento 253
15.2.1 Justificación de las propiedades básicas escogidas 253
15.2.2 Relaciones fundamentales entre las propiedades básicas 256
15.3 Marco algebraico 258
15.3.1 Preservación de las propiedades básicas 258
15.3.2 Aproximación de restricciones de reemplazamiento 259
15.3.3 Aproximación de restricciones de reemplazamiento mediante restricciones sensibles al contexto 261
15.3.4 Caracterización axiomática de las restricciones sensibles al contexto 263
15.3.5 Descomposición de restricciones de reemplazamiento 264
15.3.6 Un dominio de aproximación más preciso 267

15.3.7 Más operaciones sobre restricciones de reemplazamiento 269
15.4 Propiedades de la reescritura restringida 275
15.4.1 Completitud de la reescritura restringida 276
15.4.2 Confluencia de la reescritura restringida 278

Conclusiones y trabajo futuro **283**

Bibliografía **287**

Índice de materias **305**

Capítulo 1

Introducción

La reescritura de términos [DJ90, Klo92] es un principio operacional estándar para la definición de los lenguajes funcionales [FH88, PE93, Pey87, Rea93], algebraicos [BKLOS93, EM85, GTW78] y ecuacionales [HO82b, O'D85, O'D87, O'D95]. La reescritura es importante también para la definición de los mecanismos deductivos que se utilizan para la demostración automática de teoremas en teorías ecuacionales [BD94, Jou94, Pla94, Rus86]. En el área de la programación lógica, algunas propiedades de los programas lógicos son susceptibles de examen mediante la asociación de un sistema de reescritura de términos a éstos y su posterior caracterización operacional. En particular, su terminación o el carácter determinista de los cómputos [GW92, KKS91, Mar94]. El estrechamiento (*narrowing*), es una extensión de la reescritura (donde la unificación sustituye al emparejamiento (*matching*) como mecanismo de paso de parámetros) que se usa para la resolución automática de teoremas ecuacionales [Fay79, Sla74]. El estrechamiento (a veces con pasos de reescritura pura intercalados) [Fri85, Han97, Höl89, Hul80, Red85] y la residuación (una combinación de la reescritura con pasos de instanciación de variables situadas sobre los componentes puramente lógicos del objetivo a evaluar) [ALN87, Han95, Llo98] se utilizan también como mecanismos operacionales para la implementación de algunos lenguajes declarativos multiparadigma: los lenguajes lógico-funcionales, que combinan en un solo marco integrado las características más potentes de la programación lógica y funcional [DP88, Han94, Han95c, MR92, Red95, Red85, Rod95].

Los programas que se ejecutan en estos sistemas computacionales suelen describirse como Sistemas de Reescritura de Términos (SRTs) [DJ90, Klo92], posiblemente condicionales [BK86, DO90, DOS88, Kap84]. La ejecución de los programas consiste en *reducir* las expresiones de entrada hasta obtener una expresión perteneciente a una determinada 'familia' de expresiones canónicas. Ejemplos de familias de expre-

siones canónicas son las formas estables (constructoras o no), fuertemente estables[1], valores finitos o infinitos, formas normales finitas o infinitas, etc. Todas ellas desempeñan un papel decisivo en algún momento del proceso de cálculo asociado a una expresión inicial dada, bien sea como objetivo final de la computación (caso de los valores y las formas normales, tanto finitas como infinitas) o como formas intermedias de dicho cómputo (en el caso de las formas estables y las fuertemente estables) [FH88, KOV96, KPRRS92, PE93, O'D95, Str89].

Dado que el espacio de reducciones asociado a la evaluación[2] de una expresión puede contener muchas secuencias de reducción inútiles (que no culminen en una forma canónica de interés) o incluso peligrosas (como las secuencias de reducción infinitas o las que efectúan pasos de reducción que no contribuyen a obtener una expresión canónica), es de gran interés imponer alguna *restricción* a la relación de reescritura asociada que pueda evitar secuencias de reducción no deseadas, sin arriesgar la completitud. Dependiendo de la aplicación considerada, las secuencias de reducción 'deseables' (o significativas) deberán obtener alguna de las formas canónicas anteriormente enumeradas (relativas al SRT considerado). Por tanto, la restricción deberá ser lo suficientemente fuerte como para eliminar secuencias inútiles o peligrosas (por ejemplo, no terminantes) pero no tanto que evite realizar cómputos útiles para los fines previstos, es decir, capaces de obtener las formas canónicas deseadas.

Una *estrategias de reescritura* es un ejemplo típico de restricción de la relación de reescritura. Por lo general, las estrategias de reescritura son deterministas [AM96, Klo92], es decir: dado un término t, sólo existe una posibilidad de transformación elemental para dicho término. En general, una relación de reducción admite varias vías de transformación elemental que, bajo ciertas circunstancias, pueden hacerse converger a una única forma canónica en caso de que dicho comportamiento (determinista) sea el deseado.

En el siguiente apartado, revisamos las aproximaciones basadas en las restricciones de reemplazamiento *sintácticas*, es decir, aquéllas que están asociadas fundamentalmente a los *términos* que intervienen en el proceso de reescritura y que son, por tanto, esencialmente independientes (para su definición) del programa considerado. En este

[1] Empleamos la terminología *forma estable* y *forma fuertemente estable* como traducción de las expresiones inglesas *head-normal form* y *strong head-normal form*, respectivamente. Básicamente, se dice que un término es estable si no puede ser transformado en un redex mediante reducciones en el sistema de reescritura considerado. De forma análoga, diremos que es fuertemente estable si no puede ser transformado en un redex mediante reducciones realizadas con sistemas de reescritura cuyas reglas tengan las mismas partes izquierdas que el sistema considerado.

[2] En general, emplearemos la palabra *evaluación* para referirnos al proceso de reducciones sucesivas que, aplicado sobre una expresión inicial, tiene como objetivo la obtención de una forma canónica de cualquier tipo. En programación funcional, este vocablo adquiere un significado más preciso relacionado con la obtención de valores, es decir, términos constituidos únicamente por símbolos *constructores*, que representan los elementos del dominio asociado al tipo de datos al que pertenece la expresión inicial.

sentido, difieren de las restricciones introducidas mediante estrategias de reescritura que dependen del SRT concreto que se esté considerando.

1.1 Restricciones de reemplazamiento sintácticas

Tomaremos como base para el estudio de la teoría de las restricciones de reemplazamiento sintácticas una restricción de reemplazamiento particular, la que denominamos *reescritura sensible al contexto* (*rsc*). Para motivar este concepto y enmarcarlo en un contexto computacional conocido, consideremos la evaluación de una llamada a función $f(t_1, \ldots, t_k)$ en un lenguaje funcional genérico. Es habitual distinguir dos modos fundamentales de realizar la evaluación: *voraz* (impaciente o *estricta*) y *perezosa* (o *no estricta*) [FH88, PE93, Rea93]. Una evaluación *voraz* fuerza la evaluación de todos los argumentos $t_1, \ldots, t_k$ antes de proceder a evaluar la llamada a la función más externa f sobre los argumentos ya evaluados. Por otra parte, una evaluación *perezosa* sólo evaluará aquellos argumentos t_i, $1 \leq i \leq k$, cuya transformación sea necesaria para poder efectuar la llamada a la función f. La elección de una u otra modalidad condiciona decisivamente el comportamiento de los cómputos: una evaluación voraz no será capaz de completar la evaluación de una llamada a función si la evaluación de alguno de los argumentos $t_1, \ldots, t_k$ no termina. En cambio, la evaluación perezosa obtendrá el valor de la llamada a función siempre que exista, incluso si alguno de los argumentos no pudiera ser evaluado completamente. Sin embargo, la implementación de esta última modalidad de evaluación consume más recursos computacionales (espaciales y temporales) que la implementación de evaluaciones voraces, que suele ser más simple [Pey87]. Además, las estrategias perezosas plantean un problema más profundo: el de la definición precisa de qué se entiende por "argumento que necesita ser evaluado" y, una vez definido esto, el problema de caracterizar en qué casos (para qué programas y expresiones) es posible decidir si un argumento necesita ser evaluado o no. Esta cuestión no es trivial y ha suscitado un gran esfuerzo de investigación en el campo de la reescritura como técnica de implementación de los distintos modelos de la programación funcional o ecuacional [Ant92, Com95, DM97, HL91, Jac96, Ken89, KM91, O'D77, O'D85, Oya93, SR93, Tha87, TSEP93]. Se debe observar que la evaluación voraz está libre de este problema puesto que, sin más consideraciones, decidimos evaluar todos los argumentos de una llamada a función. Como consecuencia de esta dicotomía, se ha estudiado la posibilidad de combinar ambas modalidades de evaluación empleando, por ejemplo, una estrategia perezosa básica que puede ser modificada eventualmente intercalando pasos de evaluación impaciente de ciertos argumentos hasta alguna forma estable o normal [Bur91, Myc80, MN92, SPR90].

La reescritura sensible al contexto permite formalizar el compromiso entre estas dos

modalidades de evaluación. La restricción de reemplazamiento empleada en la reescritura sensible al contexto se define inductivamente en dos etapas: en un primer nivel (*básico* o puramente *simbólico*) la restricción de reemplazamiento se circunscribe a los símbolos de la signatura Σ mediante una función $\mu : \Sigma \to \mathcal{P}(\mathbb{N})$, llamada la *función de reemplazamiento*, que fija las posiciones argumentales $\mu(f) \subseteq \{1, \ldots, k\}$ que *pueden* reducirse (aunque no *obliga* a ello) en cualquier llamada a la función k-aria denotada por el símbolo f.

Ejemplo 1.1.1 *Consideremos la función condicional* `if` *definida mediante las siguientes reglas:*

```
if(true,x,y)  → x
if(false,x,y) → y
```

Dada una llamada a función:

`if(`$cond$`,`exp`,`exp'`)`

es claro que, en el caso general, no tiene ningún interés evaluar las expresiones exp o exp' *mientras no se conozca el resultado de evaluar la condición cond. Por tanto, es razonable prohibir las reducciones sobre el segundo y el tercer argumento, fomentando así la evaluación del primero. Esta decisión puede expresarse definiendo una función de reemplazamiento* μ *que satisfaga* $\mu(\texttt{if}) = \{1\}$.

En un segundo nivel, que llamamos *estructural*, las restricciones sobre los argumentos de los símbolos de la signatura se extienden de forma inductiva a ocurrencias de los términos construidos con estos símbolos. Al imponer estas restricciones de reemplazamiento obtenemos una restricción de la reescritura que denominamos reescritura sensible al contexto y que es una *relación* sobre la cuál todavía es posible definir refinamientos adicionales (posiblemente deterministas, es decir, estrategias).

Ejemplo 1.1.2 *Consideremos el SRT* $\mathcal{R}$*:*

```
zero    → 0                 first(0,x)       → []
from(x) → x::from(s(x))     first(s(x),y::z) → y::first(x,z)
```

Definimos $\mu(\texttt{s}) = \mu(\texttt{::}) = \mu(\texttt{from}) = \emptyset$ *y* $\mu(\texttt{first}) = \{1,2\}$*, lo cual posibilita realizar la siguiente derivación (donde los redexes reducidos en cada paso aparecen subrayados):*

```
first(zero,from(0)) → first(0,from(0)) → []
```

De esta manera, la siguiente derivación infinita (y carente de significado):

```
first(zero,from(0)) → first(zero,0::from(s(0))) → ···
```

no puede llegar a construirse, en virtud de que $\mu(\texttt{::}) = \emptyset$ *(relevante para evitar el segundo paso).*

Algunos precedentes

El empleo de restricciones de reemplazamiento sintácticas en programación se remonta, más de veinte años atrás [FW76, HM76], a las investigaciones orientadas a la obtención de implementaciones completas (es decir, libres de pérdidas computacionales debidas a ejecuciones no terminantes) del lenguaje de programación *Lisp* [McC60, McC78, WH89]. Dado el papel esencial que el procesamiento de listas desempeña en el contexto del lenguaje *Lisp*, los autores de aquellos trabajos pioneros estudiaron implementaciones del lenguaje en las que el operador de construcción de listas *cons* ('::' en esta memoria) no evaluaba sus argumentos [FW76, HM76]. Dado que el símbolo '::' no está ligado a ningún programa concreto sino más bien al tipo de datos 'lista', esta forma de establecer la restricción puede calificarse de *sintáctica* en el sentido mencionado anteriormente. Esto es similar a imponer $\mu(::) = \emptyset$. Un refinamiento de esta idea puede encontrarse en [HW87], donde los autores estudian la diseminación de anotaciones sintácticas sobre tuplas o listas, que pueden servir también para representar los bloques de parámetros pasados a una función. Sin embargo, el objetivo fundamental del trabajo era representar patrones de *strictness*[3] aplicables a *streams* (listas posiblemente infinitas) que permitieran representar modos de evaluación que combinaran los estilos perezoso y voraz de forma más flexible.

Las restricciones sintácticas de reemplazamiento también se han utilizado en la implementación de otros lenguajes de programación. En el contexto de los lenguajes funcionales (*perezosos*), se han empleado las llamadas *anotaciones de strictness* [Pey87, PE93] (denominadas *anotaciones globales* en [PE93]) para introducir cambios locales de la estrategia perezosa subyacente con el fin de obtener una ejecución más eficiente [Ben92, Bur87, Bur91, MN92, Myc80]. El esquema típico de utilización de estas anotaciones puede esbozarse como sigue [PE93]:

> *Una estrategia perezosa indica si un determinado argumento t_i de una llamada a función $f(t_1, \ldots, t_k)$ debe ser evaluado. Sin embargo, no seguiremos esta regla, y evaluaremos t_i hasta obtener una forma normal (o estable), si el i-ésimo argumento aparece* anotado *en el perfil de f.*

Las anotaciones de *strictness* desempeñan un papel secundario en el mecanismo de ejecución, dado que se asume la existencia de una estrategia subyacente que marca la secuencia de evaluación fundamental. Las anotaciones permiten desviarse, eventualmente, de dicha secuencia básica.

Las anotaciones de *strictness* para los programas se obtienen mediante algún tipo de análisis estático. Ya que este tipo de análisis suele involucrar el cálculo de puntos fijos de funciones derivadas a partir del programa analizado [CP85, Myc80, Wad87],

[3] Los vocablos ingleses cuya traducción castellana resulte forzada se dejarán sin traducir y en cursiva.

dichos análisis son costosos, pero aseguran que desviarse de la estrategia perezosa básica (cuyo carácter normalizante está garantizado) no supone riesgo de incompletitud porque los análisis de *strictness* están basados en la semántica del lenguaje funcional. Por tanto, la modificación de la estrategia no altera el comportamiento de terminación del programa. A veces, se acepta (aunque con reparos) que el programador decida cómo establecer las anotaciones del programa. Sin embargo, en este caso no hay forma de prever qué clase de modificaciones en el comportamiento introducirán dichas anotaciones.

Algunos lenguajes algebraicos, tales como OBJ2 [FGJM85], OBJ3 [GWMFJ93] o CafeOBJ [FN97], admiten la especificación de "estrategias locales" asociadas a los símbolos de función. Dichas estrategias locales se especifican como listas de números naturales y sirven para guiar la estrategia de evaluación: al considerar una llamada a función $f(t_1, \ldots, t_k)$, sólo se evalúan los argumentos cuyos índices, entre 1 y k, figuran en la lista asociada al símbolo f. Además, esto se hace de acuerdo con el orden especificado en la lista de argumentos.

También en la reescritura de términos se han utilizado restricciones sintácticas de reemplazamiento sobre los símbolos de la signatura bajo la denominación de *anotaciones sintácticas* [EP87, KW95, Mar90]. Dichas anotaciones han servido para definir restricciones de la reescritura que permiten realizar reducciones perezosas empleando modos de reducción voraces, mediante técnicas de transformación de programas [KW95], y para implementar reducciones perezosas óptimas que evitan contraer varias veces un mismo redex, empleando aquí técnicas de reducción de grafos [Mar90]. En [OF97], se propone una implementación basada en la reescritura de la estrategia de evaluación empleada en los lenguajes OBJ y CafeOBJ. Estos tipos de restricciones están muy relacionados con la *rsc*. En capítulos posteriores realizamos una presentación más detallada de las técnicas empleadas en estos trabajos junto con una comparación de éstas con nuestra propuesta. En [EP87] se presenta una técnica para diseminar anotaciones sintácticas en los subtérminos constituyentes de las reglas del SRT como un medio para definir estrategias de reducción. No obstante, la semántica de la estrategia deseada debe ser especificada mediante algunas reglas adicionales. Por tanto, el programa ejecutado no es el original sino un programa transformado obtenido a partir del inicial, lo cual aleja esta metodología de la que formalizamos en este trabajo.

Si comparamos el tratamiento de las anotaciones sintácticas en el marco de la *rsc* con el empleo de éstas en la programación funcional, podría decirse que nuestra aproximación al problema es la opuesta. La reescritura sensible al contexto puede considerarse como una mecanización de las anotaciones sintácticas *en sí mismas*. Al contrario de lo que sucede en la programación funcional, no asumimos ningún mecanismo de

evaluación preexistente. Por tanto, cuando un argumento i de una función f no está anotado (es decir, $i \notin \mu(f)$), simplemente *no puede* ser reducido, mientras que un argumento anotado *puede* (o no) ser reducido. Esto difiere sustancialmente de la semántica usual de las anotaciones en programación funcional: un argumento anotado *debe* ser reducido (en alguna medida). Un argumento no anotado *puede* ser reducido o no.

Hemos analizado las cualidades computacionales de esta forma simple de restringir el espacio de búsqueda de la reescritura: la terminación, la confluencia, la completitud de los cómputos, la definición de estrategias, etc. Además, nuestros métodos no dependen de la procedencia de las anotaciones: análisis de *strictness*, indicaciones del programador o cualquier otra fuente. Tal como demostraremos más adelante, las anotaciones influyen sobre las propiedades de la reescritura restringida resultante y nuestros análisis permiten establecer la utilidad de ésta en los cómputos que obtienen formas estables o formas normales infinitas, mientras que las anotaciones basadas en información de *strictness* no tienen utilidad evidente en estos casos. Es más, demostraremos que establecer la función de reemplazamiento de acuerdo (únicamente) con la información de *strictness* procedente del análisis del programa *no* garantiza la completitud de los cómputos mediante la *rsc*, ni siquiera para obtener formas estables. Por tanto, proporcionamos métodos para fijar de forma automática las restricciones de reemplazamiento para un SRT dado sin necesidad de servirnos de los análisis de *strictness*. Los resultados que ofrecemos están parametrizados en función del grado de restricción indicado por la función de reemplazamiento, de manera que es posible determinar qué puede esperarse para cada función de reemplazamiento que se pueda utilizar.

Por lo que respecta a las posibles extensiones de la *rsc* a otros formalismos basados en la reescritura, presentaremos algunos resultados sobre el uso de restricciones de reemplazamiento en sistemas de reescritura condicionales [BK86, DO90, DOS88, Kap84]. Ferreira y Ribeiro han extendido la *rsc* al ámbito de la reescritura módulo teorías con símbolos asociativos y conmutativos (teorías AC) [FR98].

Una propuesta original de nuestro trabajo es la utilización de las restricciones sintácticas de reemplazamiento en extensiones de la reescritura tales como el *estrechamiento* [Hul80, Sla74]. Puesto que el estrechamiento constituye el principio operacional utilizado con mayor frecuencia para implementar los lenguajes lógico-funcionales [AEH94, Fri85, Han94, MR92, Red85], las anotaciones sintácticas pueden ser utilizadas con ventaja en las implementaciones de estos lenguajes basadas en el estrechamiento. Por lo que sabemos, éste es el primer análisis sobre el uso sistemático de las anotaciones sintácticas en un lenguaje de programación lógico-funcional integrado. Hasta el momento, las anotaciones de *strictness* no han sido empleadas en los lenguajes lógico-funcionales puesto que todavía no existe ninguna noción adecuada

de *strictness* que sea aplicable a la modificación del cambio de estrategia de ejecución de los lenguajes lógico-funcionales (ver [Lin88, MKMWH93]).

1.2 Objetivo y estructura de la tesis

En esta tesis nos proponemos demostrar que la consideración de las restricciones sintácticas de reemplazamiento en reescritura y estrechamiento resulta útil y ventajosa por cuanto es posible emplear sistemáticamente la reescritura sensible al contexto en lugar de la reescritura estándar (no restringida) para mejorar el comportamiento de terminación de los programas, sin perder por ello la posibilidad de computar las formas canónicas y respuestas normalmente deseadas.

Organización de la memoria

El Capítulo 2 introduce las nociones y definiciones estándar empleadas a lo largo de la tesis. El resto del material de la tesis se estructura en cuatro partes:

1. Definición de la *rsc* y formalización de los métodos de análisis de sus *propiedades computacionales básicas* (terminación y confluencia).

2. Definición del concepto de *estrategia de reducción sensible al contexto* y estudio de sus propiedades e implementación.

3. Análisis de la capacidad de la *rsc* para realizar cómputos significativos y de las condiciones para obtener distintas *formas canónicas* mediante la *rsc*.

4. Propuesta de *extensión* de la *rsc* al estrechamiento y al ámbito de los sistemas de reescritura condicionales; definición de un marco general para describir y analizar restricciones de reemplazamiento sintácticas basadas directamente en la estructura de los términos, sin asociar restricciones fijas a los símbolos de la signatura.

A continuación resumimos el contenido de los capítulos que integran estas partes.

Propiedades computacionales básicas de la reescritura sensible al contexto

El Capítulo 3 formaliza el concepto de función de reemplazamiento para una signatura, la noción de ocurrencia reemplazante de un término y, finalmente, la relación de reescritura sensible al contexto. Se establecen las propiedades fundamentales de las restricciones cuyo conocimiento es necesario para desarrollar los análisis de las propiedades computacionales básicas en los capítulos posteriores. Se describen con

más detalle algunas propuestas similares a la *rsc* con el fin de facilitar las comparaciones posteriores. Parte del material englobado en este capítulo ha sido publicado en [Luc95, Luc98].

El Capítulo 4 estudia la conexión entre la reescritura estándar y la *rsc*. El problema planteado puede formularse como sigue: ¿hasta qué punto es posible obtener (o aproximar) mediante la *rsc* un término que es computable mediante la reescritura no restringida?. El análisis realizado lleva a la definición de la llamada función de reemplazamiento *canónica*, que condiciona la mayor parte de las propiedades de la *rsc* relacionadas con su capacidad de emular cómputos realizados mediante reescritura no restringida. Se analiza la relación entre las formas μ-normales (términos sobre los cuales no es posible realizar pasos de reducción sensible al contexto bajo una función de reemplazamiento μ). Se establecen condiciones para asegurar que dichas formas μ-normales son términos (fuertemente) estables. Parte del material contenido en este capítulo ha sido publicado en [Luc96, Luc98].

Una característica común a todas las restricciones de la reescritura es que son potencialmente capaces de mejorar las características de terminación de los programas ejecutados mediante dicha restricción. En el caso de la reescritura sensible al contexto, es posible definir métodos que, en ocasiones, permiten *demostrar* formalmente la terminación de toda secuencia de evaluación realizada bajo el control de una función de reemplazamiento determinada. El Capítulo 5 proporciona un método general para demostrar esta propiedad. Parte del material contenido en este capítulo ha sido publicado en [Luc96b]. También se comentan algunas de las técnicas desarrolladas por H. Zantema para demostrar la terminación de la *rsc* [Zan97].

El análisis de la confluencia de una relación de reducción es importante, puesto que asegura el carácter funcional (determinista) de los cómputos realizados. Cuando la relación de reducción se define a partir de un SRT, como es el caso de la *rsc*, resulta interesante establecer *condiciones sintácticas* (es decir, basadas en la forma de las reglas de reescritura y no en la propia relación de reducción) para determinar la confluencia de la relación inducida por éste. El Capítulo 6 ofrece una serie de condiciones sintácticas para garantizar la confluencia de la relación sensible al contexto para un SRT y una función de reemplazamiento dadas. Parte del material contenido en este capítulo ha sido publicado en [Luc96, Luc98].

Estrategias de reducción sensible al contexto

Dada una función de reemplazamiento μ, la elección de una *estrategia* (sensible al contexto) adecuada cobra especial importancia cuando no es posible asegurar un comportamiento terminante bajo μ para el programa considerado. Este problema se aborda en el Capítulo 7. En reescritura estándar, la definición de estrategias normalizantes computables se apoya muchas veces en la noción de *índice*. Los índices

son ocurrencias distinguidas de un término que exhiben una propiedad interesante: cuando apuntan a redexes dentro de un término, la contracción de éstos es *necesaria* para obtener una forma normal de dicho término. Primeramente, demostramos que la reducción de índices es útil también para *estabilizar* términos (es decir, reducirlos a una forma estable). A partir de este resultado, demostramos que es posible emplear ciertas secuencias de reducción de índices para obtener estrategias de reducción sensible al contexto que obtienen la forma μ-normal de un término cuando ésta existe. Parte del material contenido en este capítulo ha sido publicado en [Luc97b, Luc97c, Luc98b].

Aunque una estrategia de reescritura está ligada a un único SRT, es habitual pensar en estrategias de reescritura aplicables a clases de SRTs más que a uno en particular. En el Capítulo 8 hemos formalizado esta idea bajo la denominación de estrategia genérica. La noción de estrategia genérica es de interés teórico y práctico. Desde el punto de vista teórico, las distintas estrategias de reescritura que pueden concretarse como particularización de una estrategia genérica comparten, al fijar un SRT determinado, ciertas propiedades que pueden ser analizadas de forma homogénea y con independencia del programa considerado. Desde un punto de vista práctico, los algoritmos y estructuras de datos que implementan dichas estrategias genéricas también tienen características comunes, como un coste de implementación del mismo orden de complejidad, un algoritmo de recorrido de las estructuras de datos similar o idéntico, etc. En el marco de esta tesis, la noción de estrategia genérica ofrece la posibilidad de estudiar cómo obtener estrategias sensibles al contexto a partir de estrategias genéricas y de demostrar que las propiedades de normalización de las estrategias de reducción de índices se preservan al introducir las restricciones de reemplazamiento, con la ventaja de que las estrategias (sensibles al contexto) pueden aplicarse a una clase más general de SRTs manteniendo sus buenas propiedades. Parte del material presentado en este capítulo ha sido publicado en [Luc97b].

La implementación de las estrategias de reducción de índices también ha sido estudiada extensamente. Aunque la implementación de propósito general es costosa, se han identificado varias clases de SRTs cuya implementación puede realizarse eficazmente. En el Capítulo 9, demostramos que las técnicas de implementación aplicables a las estrategias de reducción de índices pueden adaptarse fácilmente, sin coste adicional destacable, a la implementación de estrategias sensibles al contexto que realizan también reducción de índices.

Para terminar esta segunda parte, en el Capítulo 10 presentamos algunos ejemplos de aplicación de las técnicas descritas en los capítulos anteriores.

Formas canónicas y reescritura sensible al contexto

Desde un punto de vista computacional, nuestro interés en el empleo de un mecanismo de reescritura radica en su capacidad para obtener (a partir de una determinada

expresión inicial) términos derivados (reductos) que pertenezcan a determinadas familias canónicas con una carga semántica precisa.

El Capítulo 11 estudia las condiciones que permiten obtener *formas estables constructoras* y *valores* asociados a una expresión inicial. El resultado fundamental establece la capacidad de la *rsc* para obtener dichos valores cuando la función de reemplazamiento no es más restrictiva que la llamada función de reemplazamiento *de evaluación*. Ésta es menos restrictiva que la función de reemplazamiento *canónica*, que se demuestra suficiente para obtener formas estables constructoras. Esta observación motiva la definición de una transformación de programas que permite evitar los problemas introducidos por la relajación de las restricciones de reemplazamiento inducidas por la función de reemplazamiento canónica, sin perder la posibilidad de evaluar por completo las expresiones iniciales. De esta manera, mantenemos las ventajas relativas a la terminación que se consiguen imponiendo restricciones fuertes sobre los reemplazamientos, junto con la certeza de que estas restricciones no evitarán llegar a los valores asociados a la expresión a evaluar. Parte del material presentado en este capítulo ha sido publicado en [Luc96, Luc98, Luc97d].

En el Capítulo 12 analizamos cómo obtener formas *estables* y *normales* mediante reescritura sensible al contexto. La función de reemplazamiento canónica definida en el Capítulo 4 es suficiente para obtener formas estables de las expresiones iniciales. El cómputo de las formas normales es más intrincado, aunque es posible dar condiciones para obtenerlas directamente usando la *rsc*. Sin embargo, se requiere conocer con antelación información sobre la composición de la forma normal que se va a obtener, lo cual puede ser difícil de prever. Proporcionamos algoritmos que, basados en reducciones sensibles al contexto, añaden cierto grado de control adicional que permite obtener formas normales con seguridad y eficacia. Parte del material presentado en este capítulo ha sido publicado en [Luc98].

El Capítulo 13 realiza un análisis similar, relativo al cómputo de formas normales infinitas a partir de una expresión inicial. Puesto que en el contexto de la normalización infinita no tiene sentido definir procedimientos (que deberían 'terminar' ofreciendo como resultado dicha forma normal infinita, lo cual es imposible), el análisis se centra en la utilización de la *rsc* para definir estrategias de normalización infinita.

Extensiones. Restricciones de reemplazamiento generales

En el Capítulo 14 discutimos algunas posibilidades de extensión del uso de las restricciones de reemplazamiento a otros marcos de programación distintos de la reescritura (incondicional) que hemos analizado en los capítulos anteriores. En particular, definimos el *estrechamiento sensible al contexto* (*esc*) y demostramos que es correcto, completo y capaz de ahorrar derivaciones inútiles, a semejanza de la *rsc*. También

analizamos brevemente la posibilidad de extender la *rsc* a los SRTs condicionales. Parte del material presentado en este capítulo ha sido publicado en [Luc96c, Luc98].

Finalmente, el Capítulo 15 propone un marco general (*descriptivo* y *algebraico*) para el análisis de otras restricciones de reemplazamiento sintácticas más generales que la *rsc*. El marco propuesto se ha definido con la intención de estudiar las restricciones de reemplazamiento desde un punto de vista completamente abstracto, caracterizándolas mediante un conjunto de propiedades abstractas, en lugar de mediante una definición concreta. Se define un álgebra simple de restricciones de reemplazamiento que explota la idea de aproximar unas restricciones por otras y permite analizar las propiedades computacionales de la relación de reescritura inducida por las restricciones de reemplazamiento. Al contemplar la *rsc* como una instancia particular de dicho marco, surge de forma natural una caracterización *declarativa* de la noción de restricción de reemplazamiento sensible al contexto, formulada en términos de tres propiedades fundamentales. Esto permite emplear también la *rsc* como base para aproximar las propiedades computacionales mencionadas más arriba. Parte del material presentado en este capítulo ha sido publicado en [Luc97].

Resumen de aportaciones de la tesis

Esta tesis abunda en el estudio del concepto de restricción de reemplazamiento sintáctica y su aplicación en distintos problemas computacionales en sistemas de programación cuyo principio operacional se basa en la reescritura de términos. Aunque el concepto no ha sido definido como tal anteriormente, existen algunos precedentes que pueden englobarse sin dificultad en él. Sin embargo, nuestro enfoque es novedoso por cuanto no subordina la utilización de una restricción de reemplazamiento sintáctica a la presencia de otros mecanismos de ejecución basados en reescritura sino que asigna un significado computacional propio a estas restricciones y aborda la tarea de estudiar hasta dónde se puede llegar y qué beneficios pueden obtenerse al emplear esta forma simple de reescritura restringida como alternativa a la reescritura pura. La clase de SRTs a que se aplican la gran mayoría de los resultados de esta tesis es la de los SRTs lineales por la izquierda. Sin embargo, un buen número de resultados son aplicables a SRTs arbitrarios.

Los principales resultados obtenidos en este marco son:

1. Definimos el concepto de función de reemplazamiento (Definición 3.1.1), ocurrencia reemplazante (Definición 3.1.9) y relación de reescritura sensible al contexto (Definición 3.3.1), estableciendo sus propiedades computacionales elementales (Proposiciones 3.2.1, 3.2.3, 3.3.6, 3.3.7 y 3.3.8). Dado un SRT, definimos su función de reemplazamiento canónica (Definición 4.2.1) que, entre otras cosas, permite caracterizar las formas normales respecto a la *rsc* asociada a dicho

SRT como formas (fuertemente) *estables* (Teoremas 4.3.7 y 4.3.8).

2. Proporcionamos métodos para demostrar formalmente la terminación de la *rsc* (Teoremas 5.1.3, 5.1.4 y 5.4.3). También establecemos criterios sintácticos para demostrar la confluencia local (Teorema 6.2.3), confluencia fuerte (Teorema 6.3.7) y confluencia (Teoremas 6.4.1 y 6.4.4) de la *rsc*.

3. Refinamos los resultados de Middeldorp [Mid97] acerca de los cómputos necesarios para estabilizar términos, demostrando que las restricciones de reemplazamiento sensibles al contexto permiten precisar mejor la localización de los redexes necesarios para estabilizar un término (Teoremas 7.2.3, 7.2.5, 7.4.10). Demostramos que los índices fuertemente secuenciales de Huet y Lévy [HL79, HL91], hasta ahora empleados como aproximaciones decidibles a los redexes necesarios para la normalización, también pueden emplearse como aproximaciones decidibles a los redexes necesarios para la estabilización (Teorema 7.3.3).

4. Definimos el concepto de estrategia sensible al contexto (Definición 7.1.1) y demostramos que las estrategias sensibles al contexto que realizan reducción de índices (Teorema 7.4.8) son estabilizantes (Teorema 7.5.2) y normalizantes respecto a la *rsc* (Teoremas 7.5.7 y 7.5.8).

5. Definimos la noción de estrategia genérica (Definición 8.1.1) y estudiamos cómo definir estrategias genéricas sensibles al contexto inducidas por una estrategia genérica de reducción de índices (Definición 8.2.22). Demostramos que las propiedades de normalización de ésta última son heredadas (y posiblemente ampliadas) por la primera (Teorema 8.2.27).

6. Demostramos que es posible implementar las estrategias sensibles al contexto inducidas mediante una transformación sencilla de la implementación de la estrategia de reducción de índices correspondiente (Teoremas 9.3.5 y 9.3.6).

7. Establecemos condiciones suficientes para asegurar que la *rsc* es capaz de computar formas estables constructoras (Teorema 11.1.2) y valores (Teorema 11.1.4). Introducimos una transformación de programas que permite obtener el valor de una expresión de forma eficaz (Teorema 11.7.8), combinando simultáneamente las ventajas que ofrece la función de reemplazamiento de evaluación (completitud de las evaluaciones), con las ventajas en cuanto a terminación que ofrece la función de reemplazamiento canónica (que es, en general, más restrictiva).

8. Establecemos condiciones suficientes para asegurar que la *rsc* es capaz de obtener formas estables (Teorema 12.1.1) y formas normales (Teorema 12.1.4). Dado

que el cómputo directo de formas normales mediante la *rsc* es difícil, establecemos un procedimiento de normalización basado en la *rsc* y demostramos su corrección y completitud (Teoremas 12.4.1 y 12.4.2). Establecemos condiciones que garantizan que la μ-terminación de un SRT es una condición suficiente para asegurar la existencia de una secuencia de normalización infinita para cualquier término (Teorema 13.1.1).

9. Definimos el estrechamiento sensible al contexto (Definición 14.1.3) y establecemos su corrección y completitud (Teorema 14.1.7). Definimos la reescritura condicional sensible al contexto (Definición 14.2.7) y demostramos su completitud en el cómputo de formas estables y valores (Teorema 14.2.16).

10. Definimos un marco para el análisis de las propiedades computacionales de restricciones de reemplazamiento sintácticas generales.

Capítulo 2

Preliminares

En este capítulo resumimos las definiciones y notaciones empleadas en el cuerpo de esta tesis. La Sección 2.1 presenta los conceptos matemáticos de carácter general relativos a las funciones, relaciones, secuencias, órdenes, etc.

La Sección 2.2 introduce la terminología y simbología relativa a signaturas y términos de primer orden. En particular, las nociones de ocurrencia, sustitución y término.

La Sección 2.3 se ocupa de los conceptos relacionados con los sistemas de reescritura: clases relevantes de SRTs, relación de reescritura, derivaciones, multiderivaciones, reescritura infinita, terminación y confluencia de los SRTs.

La Sección 2.4 presenta el concepto de estrategia de reescritura y discute algunas de las propiedades que resultan útiles para caracterizarlas.

La Sección 2.5 repasa la teoría de las reducciones necesarias y las nociones de Ω-reducción, índice fuertemente secuencial y SRT fuertemente secuencial como bases para definir aproximaciones decidibles a las reducciones necesarias.

2.1 Definiciones de carácter general

Emplearemos los símbolos usuales $\cup, \cap, \setminus$ para denotar la unión, intersección y diferencia de conjuntos, respectivamente. Escribiremos $A \uplus B$ para representar el conjunto $A \cup B$ cuando deseemos poner de manifiesto que se obtiene por unión de conjuntos disjuntos A y B (esto es, $A \cap B = \emptyset$). Dado un conjunto A, $\mathcal{P}(A)$ denota el conjunto de todos los subconjuntos de A.

Denotamos el conjunto de los números naturales $\{0, 1, 2, \ldots\}$ mediante el símbolo $\mathbb{N}$. El segmento inicial $\{1, 2, \ldots k\}$ del conjunto de los números naturales positivos $\mathbb{N}^+$ se denota $\mathbb{N}^+_k$. Cuando $k = 0$, entonces $\mathbb{N}^+_0 = \emptyset$. El conjunto de los números reales se denota por $\mathbb{R}$.

Dado un conjunto finito A, escribimos $|A|$ para denotar el *cardinal* del conjunto.

Dada una colección $\mathcal{A}$ de conjuntos, denotamos como $\bigcup \mathcal{A}$ al conjunto $\bigcup_{A \in \mathcal{A}} A$, unión de todos los conjuntos de la colección $\mathcal{A}$.

2.1.1 Funciones y tuplas

Dados los conjuntos A y B, una *función* $f : A \rightarrow B$ es un subconjunto $f \subseteq A \times B$ que verifica las siguientes condiciones[1]: (1) para todo $a \in A$, existe un $b \in B$ tal que $(a, b) \in f$ y (2) si $(a, b), (a, b') \in f$, entonces $b = b'$ [BCG75, DS96, DD76, RS84]. Al conjunto A se le denomina el *dominio* de la función y al conjunto B su *codominio*. Cuando $(a, b) \in f$, escribimos $f(a)$ para referirnos al elemento b. Si no se cumple la condición (1), se dice que f es una función *parcial*. Si no se cumple (2), se dice que f es una función *indeterminista*. Por lo general, hablaremos simplemente de 'funciones', dejando que el contexto del discurso aclare, en su caso, los matices específicos que puedan ser relevantes. Dada una función $f : A \rightarrow B$ y un subconjunto $C \subseteq A$, escribimos $f_{|C}$ para denotar la *restricción* $f_{|C} : C \rightarrow B$ de la función f al subconjunto C: $f_{|C}(x) = f(x)$ para todo $x \in C$. Dadas dos funciones $g : B \rightarrow C$ y $f : A \rightarrow B$, la *composición* de ambas funciones se denota como $g \circ f : A \rightarrow C$.

Una función f es *inyectiva* si para todo $a, a' \in A$, $f(a) = f(a')$ implica que $a = a'$. Una función f es *sobreyectiva* si para todo $b \in B$ existe un $a \in A$ tal que $f(a) = b$. Una función *biyectiva* (o correspondencia biunívoca) es una función inyectiva y sobreyectiva. Si $f : A \rightarrow B$ es biyectiva entonces la función $f^{-1} : B \rightarrow A$ definida por $f^{-1}(b) = a$ si y sólo si $f(a) = b$ se denomina función *inversa* de f.

Una función $f : A \rightarrow A$ es *idempotente* si para todo $a \in A$, $f(f(a)) = f(a)$. Un elemento $a \in A$ se denomina un *punto fijo* de f si $f(a) = a$. La función $id_A : A \rightarrow A$ definida por $id_A(a) = a$ para todo $a \in A$ se denomina la función *identidad* para el conjunto A. Dada $f : A \rightarrow A$, se define la *potencia* n-ésima f^n de f como sigue:

$$f^0 = id_A \text{ y } f^{n+1} = f \circ f^n \text{ para } n \geq 0.$$

A veces emplearemos el mismo símbolo f asociado a una función $f : A \rightarrow B$ para denotar su *extensión* a subconjuntos $C \subseteq A$ del dominio de f: $f(C) = \{f(a) \mid a \in C\}$. Dado un conjunto A, denotamos una *k-tupla* $x_1, \ldots, x_k$ de elementos de A como $\tilde{x}$, es decir, $\tilde{x} \in A^k$. Escribiremos $\tilde{x}[y]_i$ para referirnos a la k-tupla que es idéntica a $\tilde{x}$ excepto en el elemento i-ésimo que ha sido reemplazado por y. Dada una k-tupla $\tilde{x}$ y un conjunto de números naturales positivos $J = \{j_1, \ldots, j_q\} \subseteq \mathbb{N}_k^+$, $\tilde{x}_J$ denota la q-tupla de elementos extraídos de $\tilde{x}$ de acuerdo con J, es decir, $\tilde{x}_J = x_{j_1}, \ldots, x_{j_q}$. Observemos que el i-ésimo componente x_i de $\tilde{x}$ es el $|\mathbb{N}_i^+ \cap J|$-ésimo componente de $\tilde{x}_J$. Dado un conjunto B, una función $\psi : A \rightarrow B$ y una k-tupla $\tilde{x} \in A^k$, $\psi(\tilde{x})$ es

[1] El concepto de función que manejamos aquí se denomina a veces *aplicación* o *función total* en la literatura.

la k-tupla $\psi(x_1), \ldots, \psi(x_k)$ de elementos de B obtenidos a partir de $\tilde{x}$ mediante la aplicación de ψ a cada uno de sus elementos.

2.1.2 Relaciones binarias

Sea A un conjunto y $R \subseteq A \times A$ una relación binaria sobre A. Escribiremos $a\ R\ b$ en lugar de $(a,b) \in R$ y $a \not R\ b$ en lugar de $(a,b) \notin R$. Decimos que R es: *reflexiva*, si para todo $a \in A$, $a\ R\ a$; *irreflexiva*, si para todo $a \in A$, $a \not R\ a$; *transitiva*, si para todo $a,b,c \in A$, $a\ R\ b$ y $b\ R\ c$ implican $a\ R\ c$; *antisimétrica*, cuando $a\ R\ b$ y $b\ R\ a$ implican $a = b$; *simétrica*, cuando $a\ R\ b$ implica $b\ R\ a$.

Denotamos como R^ϵ el *cierre*[2] *reflexivo* de la relación R: $R^\epsilon = R \cup \{(a,a) \mid a \in A\}$. R^+ es el *cierre transitivo* de R y R^* el *cierre reflexivo-transitivo* de R.

Dadas dos relaciones R, S sobre A, denotamos como $R \circ S$ la relación sobre A obtenida por *composición de* de ambas: para todo $a,b \in A$, $a\ R \circ S\ b$ si y sólo si existe un $c \in A$ tal que $a\ R\ c$ y $c\ S\ b$. Dados dos conjuntos A, B y relaciones binarias $R \subseteq A \times A$ y $S \subseteq B \times B$, una función $h : A \to B$ es un *homomorfismo* entre las relaciones R y S si para todo $a,b \in A$, $a\ R\ b$ implica que $h(a)\ S\ h(b)$.

Una *secuencia de reducción* finita (de longitud n) asociada a una relación R sobre un conjunto A es una secuencia $a_1, a_2, \ldots, a_{n+1}$ de $n+1$ elementos de A tal que, para todo $1 \leq i \leq n$, $a_i\ R\ a_{i+1}$.

Una relación binaria R sobre A es *confluente* si, para todo $a,b,c \in A$, siempre que $a\ R^*b$ y $a\ R^*c$, existe un $d \in A$ tal que $b\ R^*d$ y $c\ R^*d$ [Hue80, Ros73]. De forma análoga, R es *localmente confluente* si, para todo $a,b,c \in A$, cuando $a\ R\ b$ y $a\ R\ c$, existe un $d \in A$ tal que $b\ R^*d$ y $c\ R^*d$. Decimos que R es *fuertemente confluente* si, para todo $a,b,c \in A$, $a\ R\ b$ y $a\ R\ c$ implican que $b\ R^*d$ y $c\ R^\epsilon d$ para algún $d \in A$.

Lema 2.1.1 [Hue80] *Sean $R, S \subseteq A \times A$ relaciones tales que $R^+ = S^+$. Si S es fuertemente confluente, entonces R es confluente.*

Un elemento $\bar{a} \in A$ se denomina una *forma R-normal* (llamada simplemente forma normal cuando la relación R involucrada queda clara a partir del contexto discursivo) si no existe ningún $b \in A$ tal que $\bar{a}\ R\ b$. Decimos que $\bar{a}$ es una forma R-normal de a, si $\bar{a}$ es una forma R-normal y $a\ R^*\bar{a}$. Decimos que R es *terminante* [Hue80] si y sólo si no existe ninguna secuencia infinita $a_0\ R\ a_1\ R\ a_2 \cdots$. Cuando una relación es terminante, cada elemento $a \in A$ tiene (al menos) una forma normal. Cuando una relación es confluente y terminante, la forma normal existe y es *única*.

Lema 2.1.2 (Lema de Newman [New42]**)** *Una relación terminante R sobre un conjunto A es confluente si y sólo si es localmente confluente.*

[2] Dado un conjunto A incluido en un conjunto E, y una propiedad P sobre elementos de E, el *cierre* de A respecto a P es el menor conjunto incluido en E que contiene A y satisface P.

Si la relación $\equiv \subseteq A \times A$ tiene las propiedades reflexiva, transitiva y simétrica, se denomina una *relación de equivalencia*. El *conjunto cociente* $A/\equiv = \{B \subseteq A \mid \forall a, a' \in A.\ a \equiv a' \Leftrightarrow a, a' \in B\}$ definido por la equivalencia $\equiv$ es una partición del conjunto A.

Dada una relación $R \subseteq A \times A$ y un subconjunto $B \subseteq A$, decimos que B es *cerrado bajo* R si, para todo $a \in B$, siempre que $a\ R\ b$, se tiene que $b \in B$. En particular, si $f : A \rightarrow A$ es una función, $B \subseteq A$ es cerrado bajo f si $f(B) \subseteq B$.

2.1.3 Secuencias infinitas

Una *secuencia infinita* S de elementos tomados de un conjunto A es una función $S : \mathbb{N}^+ \rightarrow A$. Nos referiremos al n-ésimo elemento de la secuencia escribiendo S_n en lugar de $S(n)$. Al manipular secuencias infinitas sobre un conjunto A, es interesante disponer de una noción de *convergencia a un límite* de dichas secuencias. Esto puede conseguirse introduciendo una *distancia* (o *métrica*) entre los elementos del conjunto A. Una distancia en A es una función $d : A \times A \rightarrow \mathbb{R}$ tal que, para toda terna de elementos $x, y, z \in A$, se verifica [Kel75, Man74b]

$$
\begin{aligned}
&d(x, y) \geq 0,\\
&d(x, y) = 0 \text{ si y sólo si } x = y,\\
&d(x, y) = d(y, x)\\
&d(x, z) \leq d(x, y) + d(y, z)
\end{aligned}
$$

Un conjunto A junto con una distancia d en A es un *espacio métrico* (A, d). Sea (A, d) un espacio métrico. Una secuencia S de elementos de A se denomina *convergente* si existe un elemento $\bar{s} \in A$ tal que, para todo $\epsilon > 0$, existe un $n \in \mathbb{N}$ tal que, para todo $p \geq n$, se verifica $d(S_p, \bar{s}) < \epsilon$. Si dicho elemento $\bar{s}$ existe, es único y se denomina el *límite* de la secuencia S. Una secuencia S es una *secuencia de Cauchy* si para todo $\epsilon > 0$ existe un $n \in \mathbb{N}$ tal que para todo $p \geq n$ y para todo $q \geq n$ se verifica $d(S_p, S_q) < \epsilon$. Toda secuencia convergente es una secuencia de Cauchy pero no al revés, es decir, existen secuencias de Cauchy que no tienen límite en A. Un espacio métrico se denomina *completo* si toda secuencia de Cauchy es convergente. Es un hecho bien conocido que todo espacio métrico (A, d) puede ser *completado* hasta obtener un espacio métrico completo $(\widehat{A}, \widehat{d})$ mediante el procedimiento estándar denominado *compleción métrica* [Man74b].

Una *secuencia de reducción* infinita asociada a una relación R sobre un conjunto A es una secuencia infinita $a_1, a_2, \ldots, a_n, \ldots$ de elementos de A tal que, para todo $i \geq 1$, $a_i R\ a_{i+1}$.

2.1.4 Conjuntos ordenados

Una relación $\sqsubseteq \subseteq A \times A$ es una relación de *orden parcial* (sobre A) si es reflexiva, antisimétrica y transitiva [DD76, GHKLMS80, SLG94]. Escribimos $a \sqsubset b$ para indicar que $a \sqsubseteq b \wedge a \neq b$. Escribimos $b \sqsupseteq a$ para expresar que $a \sqsubseteq b$, y $a \parallel b$ para indicar que $a \not\sqsubseteq b$ y $b \not\sqsubseteq a$. Si $\sqsubseteq$ es una relación de orden sobre A, decimos que $(A, \sqsubseteq)$ (o simplemente A) es un *conjunto ordenado*. La relación $\sqsubseteq$ es un orden total si para todo $a, b \in A$, se tiene $a \sqsubseteq b$ o bien $b \sqsubseteq a$. Una relación $\sqsubset$ irreflexiva y transitiva es un *orden estricto* y $(A, \sqsubset)$ un conjunto *estrictamente ordenado*. Un orden $(A, \sqsubseteq)$ se dice *bien fundado* si no existen secuencias infinitas $a_1, \ldots, a_n, \ldots$ tales que $a_{i+1} \sqsubset a_i$ para todo $i \geq 1$.

Un elemento $a \in A$ tal que para todo $b \in A$ siempre que $b \sqsupseteq a$ ($b \sqsubseteq a$) entonces $b = a$ se denomina elemento *maximal* (*minimal*) de A. Un elemento $a \in A$ es una *cota superior* (*inferior*) de $B \subseteq A$ si para todo $b \in B$, $b \sqsubseteq a$ ($a \sqsubseteq b$). Decimos que el conjunto B está *acotado superiormente* (*inferiormente*) si B tiene una cota superior (inferior). Si B está acotado decimos que sus elementos son *compatibles*. Si dos elementos $a, b \in A$ son compatibles, es decir, si $\{a, b\}$ es un conjunto acotado, escribimos $a \uparrow b$. Una cota superior (inferior) $a \in A$ es un *supremo* (*ínfimo*) de $B \subseteq A$ si, para toda cota superior (inferior) $a' \in A$ de B, se tiene $a \sqsubseteq a'$ ($a' \sqsubseteq a$). Si $B \subseteq A$ tiene un supremo se denota a éste como $\bigsqcup B$. Si el supremo (ínfimo) de un subconjunto $B \subseteq A$ pertenece a B, se le denomina el *máximo* (*mínimo*) de B. Si el propio conjunto A tiene un máximo (mínimo) se denota como $\top_A$ ($\bot_A$). Una secuencia (posiblemente infinita) $a_1, a_2, \ldots a_n \ldots$ de elementos de A es una *cadena* si $a_1 \sqsubseteq a_2 \sqsubseteq \cdots \sqsubseteq a_n \sqsubseteq \cdots$.

Un subconjunto $B \subseteq A$ de un conjunto ordenado $(A, \sqsubseteq)$ es un *cono* (inverso) si para todo $b \in B$, siempre que $a \in A$ verifica $a \sqsubseteq b$ ($a \sqsupseteq b$) entonces $a \in B$. Un *orden parcial completo* (*cpo*, del inglés *complete partial order*) es un conjunto ordenado donde toda cadena $B : a_1 \sqsubseteq a_2 \sqsubseteq \cdots \sqsubseteq a_n \sqsubseteq \cdots$ tiene un supremo[3] (también llamado *límite* de la cadena) $\bigsqcup B$. Un *retículo completo* es un conjunto ordenado donde todo subconjunto $B \subseteq A$ tiene un supremo y un ínfimo. En particular, el propio conjunto A tiene un elemento máximo $\top = \bigsqcup A$ y un elemento mínimo $\bot$ que es el ínfimo de A. Se asume, para el caso especial $B = \emptyset$, que $\bigsqcup \emptyset = \bot$ y que el ínfimo del conjunto vacío es $\top$. Además, todo par de elementos $a, b \in A$ tienen un supremo y un ínfimo y escribimos $a \sqcup b$ y $a \sqcap b$ para denotar el supremo y el ínfimo de $\{a, b\}$ respectivamente. Estas operaciones son idempotentes, es decir, para todo $a \in A$, $a \sqcup a = a$ y $a \sqcap a = a$. La estructura $(A, \sqsubseteq, \bot, \top, \sqcup, \sqcap)$ denota el retículo completo. Si, adicionalmente, se

[3] El concepto de *cpo* que manejamos aquí es el que habitualmente se conoce como ω-*cpo* pues caracteriza la completitud del conjunto ordenado respecto a los supremos de las (ω-)cadenas crecientes de elementos. En general, la completitud (y por tanto la definición) de un *cpo* se refiere a la existencia de supremos para *conjuntos dirigidos* incluidos en el conjunto ordenado.

cumple $a \sqcup (b \sqcap c) = (a \sqcup b) \sqcap (a \sqcup c)$ y $a \sqcap (b \sqcup c) = (a \sqcap b) \sqcup (a \sqcap c)$ el retículo es *distributivo*.

Una función $f : A \to B$ entre conjuntos ordenados $(A, \sqsubseteq_A), (B, \sqsubseteq_B)$ es *monótona*, si para todo $a, b \in A$, $a \sqsubseteq_A b$ implica $f(a) \sqsubseteq_B f(b)$. Si dichos conjuntos ordenados son órdenes completos, se dice que f es *continua* si dada una cadena $B : a_1 \sqsubseteq a_2 \sqsubseteq \cdots \sqsubseteq a_n \sqsubseteq \cdots$ se tiene que $f(\bigsqcup B) = \bigsqcup f(B)$.

Teorema 2.1.3 (Teorema del punto fijo [SLG94]**)** *Sea $(A, \sqsubseteq, \bot)$ un* cpo *con elemento mínimo $\bot$ y $f : A \to A$ una función continua. Entonces f tiene un punto fijo mínimo $a \in A$ que puede obtenerse como el supremo de la cadena creciente (secuencia de Kleene):*

$$f^0(\bot) \sqsubseteq f^1(\bot) \sqsubseteq f^2(\bot) \sqsubseteq \cdots \sqsubseteq f^n(\bot) \sqsubseteq \cdots$$

es decir,

$$a = \bigsqcup_{n \geq 0} f^n(\bot)$$

Si los conjuntos ordenados $(A, \sqsubseteq_A)$ y $(B, \sqsubseteq_B)$ tienen elementos mínimos $\bot_A$ y $\bot_B$, respectivamente, decimos que $f : A \to B$ es *estricta* si $f(\bot_A) = \bot_B$. Una función multiargumento $f : A_1 \times \cdots \times A_k \to B$, donde $(A_i, \sqsubseteq_i, \bot_i)$ y $(B, \sqsubseteq, \bot)$ son conjuntos ordenados provistos de sus correspondientes elementos mínimos para $1 \leq i \leq k$, se denomina *i-estricta* si para todo $x_j \in A_j$, $1 \leq j \leq k$, $i \neq j$, se verifica $f(x_1, \ldots, \bot_i, \ldots, x_k) = \bot$. Una función $f : A \to A$ de un conjunto ordenado $(A, \sqsubseteq)$ en sí mismo es *extensiva* si para todo $a \in A$, $a \sqsubseteq f(a)$. Una función f extensiva, idempotente y monótona es un *operador de cierre*.

Dado un conjunto de funciones $F \subseteq A \to B$ donde $(B, \sqsubseteq)$ es un conjunto ordenado, podemos definir el siguiente orden (orden punto a punto) en F: $f \sqsubseteq g$ si y sólo si, para todo $a \in A$, $f(a) \sqsubseteq g(a)$. Dadas las tuplas $(a_1, a_2, \ldots, a_n)$ y $(b_1, b_2, \ldots, b_n)$, donde los elementos a_i, b_i pertenecen a un mismo conjunto ordenado $(A_i, \sqsubseteq_i)$ para $1 \leq i \leq n$, se define la *extensión lexicográfica* $(\sqsubset_1, \sqsubset_2, \ldots, \sqsubset_n)_{lex}$ de los órdenes estrictos correspondientes $\sqsubset_1, \sqsubset_2, \ldots, \sqsubset_n$ a tuplas de elementos como sigue:

$$(a_1, a_2, \ldots, a_n)\ (\sqsubset_1, \sqsubset_2, \ldots, \sqsubset_n)_{lex}\ (b_1, b_2, \ldots, b_n)$$

si $a_1 \sqsubset_1 b_1$, o bien $a_1 = b_1$ y

$$(a_2, \ldots, a_n)\ (\sqsubset_2, \ldots, \sqsubset_n)_{lex}\ (b_2, \ldots, b_n).$$

2.2 Signaturas y términos

En lo que sigue, V representará un conjunto numerable de variables y Σ una signatura, es decir, un conjunto de símbolos de función $\{\mathtt{f}, \mathtt{g}, \ldots\}$, dotados de una aridad que

viene individualizada por una función $ar_\Sigma : \Sigma \to \mathbb{N}$ (o simplemente ar, cuando ello no suponga confusión) [Klo92]. Siempre asumiremos que $\Sigma \cap V = \emptyset$. El conjunto de *términos de primer orden* que se obtiene a partir de una signatura Σ y un conjunto de variables V se denota como $\mathcal{T}(\Sigma, V)$: si x es una variable entonces $x \in \mathcal{T}(\Sigma, V)$; si f es una constante, entonces $f \in \mathcal{T}(\Sigma, V)$; finalmente, si $t_1, \ldots, t_k \in \mathcal{T}(\Sigma, V)$ son términos y $f \in \Sigma$ es un símbolo de función k-ario, entonces $f(t_1, \ldots, t_k) \in \mathcal{T}(\Sigma, V)$. Una k-tupla $t_1, \ldots, t_k$ de términos se denota como $\tilde{t}$, donde el número de componentes k será clarificado a partir del contexto donde la tupla sea empleada.

Dado un término t, $Var(t)$ es el conjunto de los símbolos de variable presentes en t. Si $Var(t) = \emptyset$, decimos que t es un término *básico*; $\mathcal{T}(\Sigma)$ denota el conjunto de términos básicos.

Se utilizará un nuevo símbolo constante Ω (que se asume ausente de todas las signaturas consideradas) para representar términos cualesquiera. Los términos pertenecientes al conjunto $\mathcal{T}(\Sigma \cup \{\Omega\}, V)$ se denominan Ω-términos y se emplean para representar prefijos de términos. Dado un término t, t_Ω es el Ω-término obtenido al reemplazar todas las variables de t por símbolos Ω. Se define un orden $\leq$ sobre Ω-términos como el menor orden que satisface las restricciones: $\Omega \leq t$ para todo t y $f(\tilde{t}) \leq f(\tilde{s})$ si $t_i \leq s_i$ para todo $1 \leq i \leq ar(f)$.

Observación 2.2.1 *Los Ω-términos y el orden $\leq$ entre ellos permiten formalizar la idea de aproximación de un término por otro. Si $t \leq s$, t y s comparten una parte de sus estructuras sintácticas (la que corresponde a las ocurrencias de t que no se refieren a símbolos Ω), siendo s un* refinamiento *de t en el sentido de que s puede obtenerse a partir de t mediante el reemplazamiento de algunas ocurrencias de Ω en t por otros Ω-términos posiblemente más complejos. También decimos que t es un prefijo de s.*

2.2.1 Ocurrencias

Una *ocurrencia* u es una cadena de números enteros positivos $u \in Occ = \{\epsilon\} \cup \{i.v \mid i \in \mathbb{N}^+ \wedge v \in Occ\}$ donde ϵ denota la ocurrencia correspondiente a la cadena vacía y '.' denota la concatenación de símbolos. La longitud de una cadena u se escribe $|u|$. Si u es una ocurrencia y W un conjunto de ocurrencias, $u.W$ es el conjunto $\{u.v \mid v \in W\}$. Establecemos un orden $\leq$ entre ocurrencias empleando el orden de prefijos entre cadenas: $u \leq v$ si y sólo si existe una cadena w tal que $v = u.w$. Escribiremos $u \parallel v$ para indicar que u y v no pueden ser comparadas mediante el orden $\leq$. Diremos entonces que u y v son ocurrencias disjuntas.

También emplearemos el orden total $\leq_L$ (lexicográfico) entre ocurrencias. Viene definido como sigue: $\epsilon \leq_L u$ para toda occurrencia u; si $u = i.u'$ y $v = j.v'$ entonces $u \leq_L v$ si y sólo si $i < j \vee (i = j \wedge u' \leq_L v')$.

2.2.2 Sustituciones

Una *sustitución* es una función $\sigma : V \to \mathcal{T}(\Sigma, V)$. Normalmente se exige que $\sigma(x) \neq x$ sólo para un número finito de variables. Al conjunto (finito) $\mathcal{D}om(\sigma) = \{x \in V \mid \sigma(x) \neq x\}$ de dichas variables se le denomina el *dominio* de la sustitución. A veces se denota una sustitución σ indicando los enlaces elementales $x_i \mapsto t_i$ para $1 \leq i \leq n$ correspondientes a las variables $\{x_1, \ldots, x_n\}$ del dominio $\mathcal{D}om(\sigma)$ escribiendo $\sigma = \{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$. Denotamos la sustitución vacía o identidad mediante la letra griega ε. Tendremos que $\mathcal{D}om(\sigma) = \emptyset$ o, equivalentemente, $\varepsilon(x) = x$ para toda $x \in V$.

La función σ se extiende a una función sobre términos denotada por el mismo símbolo $\sigma : \mathcal{T}(\Sigma, V) \to \mathcal{T}(\Sigma, V)$ de forma que, para todo término $f(\tilde{t})$, se verifica $\sigma(f(\tilde{t})) = f(\sigma(\tilde{t}))$. Se define un orden $\leq$ sobre el conjunto $Subst(\Sigma, V)$ de sustituciones: $\sigma \leq \sigma'$ si existe una sustitución σ'' tal que $\sigma'' \circ \sigma = \sigma'$. Dado un subconjunto de variables $W \subseteq V$, escribiremos $\sigma = \sigma'[\![W]\!]$ si $\sigma_{|W} = \sigma'_{|W}$ y $\sigma \leq \sigma'[\![W]\!]$ si existe una sustitución σ'' tal que $\sigma'' \circ \sigma = \sigma'[\![W]\!]$.

Dadas dos sustituciones $\sigma, \sigma' \in Subst(\Sigma, V)$ tales que $\mathcal{D}om(\sigma) \cap \mathcal{D}om(\sigma') = \emptyset$, definimos la *unión* $\sigma \cup \sigma'$ de ambas sustituciones como $\sigma \cup \sigma'(x) = \sigma(x)$ si $x \in \mathcal{D}om(\sigma)$ y $\sigma \cup \sigma'(x) = \sigma'(x)$ si $x \in \mathcal{D}om(\sigma')$.

Una sustitución σ se denomina *básica* si para todo $x \in \mathcal{D}om(\sigma)$, $\sigma(x)$ es un término básico. Una sustitución σ es un *renombramiento* si existe σ^{-1} tal que $\sigma \circ \sigma^{-1} = \sigma^{-1} \circ \sigma = \varepsilon$.

Un término t *empareja* con un término l si existe una sustitución σ (denominada *sustitución de emparejamiento* o simplemente *emparejamiento*) tal que $t = \sigma(l)$. Cuando $Var(t) \cap Var(l) = \emptyset$, el emparejamiento σ puede descomponerse como la unión $\sigma = \sigma_l \cup \sigma_t$ de sustituciones σ_l y σ_t tales que $\mathcal{D}om(\sigma_l) = Var(l)$ y $\mathcal{D}om(\sigma_t) = Var(t)$. Dos términos t, s *unifican* si existe una sustitución σ tal que $\sigma(t) = \sigma(s)$. A la sustitución σ se le denomina un *unificador* de t y s. Si un unificador θ de los términos t y s verifica que $\theta \leq \sigma$ para cualquier otro unificador σ, decimos que θ es el *unificador más general* (*umg*) de t y s. Dicho *umg* es único salvo composición con sustituciones de renombramiento.

2.2.3 Términos y ocurrencias

Consideramos los términos como árboles etiquetados, de la manera usual. Para identificar un nodo constituyente de un término t empleamos ocurrencias. El conjunto de ocurrencias de un término t es $O(t)$. Escribimos $t|_u$ para denotar el *subtérmino* de t a la occurrencia $u \in O(t)$: $t|_\epsilon = t$ y $f(t_1, \ldots, t_k)|_{i.u} = t_i|_u$ donde $1 \leq i \leq k$. La *profundidad* d en t de un subtérmino $s = t|_u$ situado a la ocurrencia $u \in O(t)$, se define como la longitud de la ocurrencia u: $d = |u|$. Escribimos $t[s]_u$ para referirnos al

término t donde el subtérmino a la ocurrencia u ha sido *reemplazado* por s: $t[s]_\epsilon = s$ y $f(t_1,\ldots,t_i,\ldots,t_k)[s]_{i.u} = f(t_1,\ldots,t_i[s]_u,\ldots,t_k)$ donde $1 \leq i \leq k$. Escribimos $root(t)$ para referirnos al símbolo que etiqueta la *raíz* del (árbol asociado al) término t: $root(x) = x$ si $x \in V$ y $root(f(t_1,\ldots,t_k)) = f$ si $f \in \Sigma$. Dados los términos t, s, escribimos $t \prec s$ para indicar que t es un subtérmino *propio* de s, es decir, que existe $u \in O(s)$, $u \neq \epsilon$ tal que $t = s|_u$.

Observación 2.2.2 *El orden $\leq$ sobre ocurrencias nos permite dar significado a expresiones como* "símbolo (o término) que se encuentra sobre o encima (debajo) de otro símbolo (o término)*": si t es un término y $u, v \in O(t)$ son tales que $u < v$, decimos que el símbolo $root(t|_u)$ (el subtérmino $t|_u$) está sobre (por encima de) $root(t|_v)$ (de $t|_v$).*

De forma análoga, el orden $\leq_L$ nos permite hablar de símbolos y subtérminos a la 'izquierda' y 'derecha' de otros en el seno de un término t así como (en virtud de su carácter de orden total) de "extremo izquierdo" y "extremo derecho" de un término.

El conjunto $O_\Sigma(t) = \{u \in O(t) \mid root(t|_u) \in \Sigma\}$ contiene las ocurrencias de símbolos *no variables* en t y $O_V(t) = O(t) \backslash O_\Sigma(t)$ es el conjunto de ocurrencias de variables. Escribimos $O_s(t)$, para denotar el conjunto de ocurrencias del subtérmino s en t: $O_s(t) = \{u \in O(t) \mid t|_u = s\}$. Dado un Ω-término t, las ocurrencias $u \in O_\Omega(t)$ se denominan Ω-ocurrencias de t. El conjunto de ocurrencias que no localizan símbolos Ω en t es $\overline{O}_\Omega(t) = O(t) \backslash O_\Omega(t)$.

Un término es *lineal* si no contiene múltiples ocurrencias de una misma variable. El conjunto de símbolos de la signatura Σ que están presentes en t es $\Sigma(t) = \{f \in \Sigma \mid \exists u \in O(t).\ root(t|_u) = f\}$. La cadena de símbolos que se encuentran en el camino que va desde la raíz del árbol asociado a un término t hasta la ocurrencia $u \in O(t)$ (inclusive) es $prefix_t(u)$, que se define como sigue: $prefix_t(\epsilon) = root(t)$, $prefix_t(i.u) = root(t).prefix_{t_i}(u)$. El *prefijo estricto* $sprefix_t(u)$ viene dado por $sprefix_t(\epsilon) = \epsilon$, $sprefix_t(u.i) = prefix_t(u)$.

Emplearemos, de forma auxiliar, las siguientes propiedades y resultados estándar [Hue80]. Consideremos $t_1, t_2, t_3 \in \mathcal{T}(\Sigma, V)$.

Si $u \in O(t_1)$ y $v \in O(t_2)$ entonces

- $(t_1[t_2]_u)|_{u.v} = t_2|_v$ (*embedding*),
- $t_1[t_2[t_3]_v]_u = t_1[t_2]_u[t_3]_{u.v}$ (*asociatividad*).

Si $u, v \in O(t_1)$ y $u \parallel v$ tenemos que

- $(t_1[t_2]_u)|_v = t_1|_v$ (*persistencia*) y
- $t_1[t_2]_u[t_3]_v = t_1[t_3]_v[t_2]_u$ (*conmutatividad*).

Si $u \leq v$, haciendo $v = u.w$ obtenemos

- $t_1|_v = (t_1|_u)|_w$ (*cancelación*),
- $t_1[t_2]_v[t_3]_u = t_1[t_3]_u$ (*dominancia*) y
- $(t_1[t_2]_v)|_u = (t_1|_u)[t_2]_w$ (*distributividad*).

Proposición 2.2.3 [Hue80] *Sea Σ una signatura, $t \in \mathcal{T}(\Sigma, V)$ y $\sigma \in Subst(\Sigma, V)$. Entonces,*

$$O(\sigma(t)) = O(t) \cup \bigcup_{t|_u \in V} \{u.v | v \in O(\sigma(t|_u))\}$$

y por tanto, para toda ocurrencia $u \in O(t)$:

1. *Si $t|_u = t' \notin V$, entonces $\sigma(t)|_u = \sigma(t')$;*
2. *Si $t|_u = x \in V$, entonces $\sigma(t)|_{u.v} = \sigma(x)|_v$ para todo $v \in O(\sigma(x))$.*

Proposición 2.2.4 [Hue80] *Sea Σ una signatura, $t, s \in \mathcal{T}(\Sigma, V)$ y $\sigma \in Subst(\Sigma, V)$. Entonces, para toda ocurrencia $u \in O(t)$, tenemos que $\sigma(t)[\sigma(s)]_u = \sigma(t[s]_u)$.*

Un *contexto* C es un término con cero o más 'huecos', es decir, ocurrencias de un símbolo especial $\Box$ (un nuevo símbolo constante). Escribimos $C[\,]_u$ para poner de manifiesto que existe un hueco $\Box$ (normalmente único) situado sobre la ocurrencia u de C. Normalmente escribiremos simplemente $C[\,]$ para denotar un contexto arbitrario, dejando que el número y localización concreta de sus huecos sean clarificados en cada momento. Al contexto $C[\,] = \Box$ se le denomina el contexto *vacío*.

2.3 Sistemas de reescritura

Una *regla de reescritura* es un par ordenado (l, r), escrito $l \to r$ (o bien $\alpha : l \to r$ si se desea asociar una etiqueta α a la regla), donde $l, r \in \mathcal{T}(\Sigma, V)$, $l \notin V$ y $Var(r) \subseteq Var(l)$ [DJ90, Klo92]. Si no se cumple la condición $Var(r) \subseteq Var(l)$, decimos que $l \to r$ es una regla de reescritura *con variables extra*. La *parte izquierda* (*lhs*, del inglés *left-hand side*) de una regla $l \to r$ es l y r es la *parte derecha* (*right-hand side* o *rhs*). Un *Sistema de Reescritura de Términos* (SRT) es un par $\mathcal{R} = (\Sigma, R)$ donde R es un conjunto de reglas de reescritura. Si alguna regla $l \to r \in R$ contiene variables extra, diremos que $\mathcal{R}$ es un SRTE, es decir, un SRT con variables extra.

Observación 2.3.1 *Aunque la noción estándar de SRT prohíbe la presencia de reglas con variables extra, los SRTEs son interesantes en la teoría de reescritura (y sus extensiones) desde muchos puntos de vista [Com95, DOS88b, DM97, KKSV95, Han95b,*

Jac96, SMI95]. En el marco de esta tesis, veremos que muchos resultados importantes son válidos para SRTEs. En particular, los relacionados con la demostración de terminación y completitud de los cómputos mediante la rsc*.*

Aunque muchas de las definiciones relativas a propiedades sintácticas de los SRTs se aplican también a SRTEs, no sucede lo mismo con sus propiedades computacionales. Por tanto, en estos preliminares referimos todas las definiciones a SRTs, asumiendo la extensión natural del concepto para SRTEs. Únicamente cuando la definición no tenga sentido para SRTEs lo haremos constar explícitamente. En cuanto a los resultados estándar, los considerados en estos preliminares se refieren, en general, a SRTs. Puesto que así se encuentran formulados en las referencias apuntadas, no se asumirá su validez al referirlos a SRTEs salvo que ello esté justificado. En el resto de la tesis, cada proposición hará explícito el ámbito de aplicación (SRTs o SRTEs) del resultado correspondiente.

El conjunto de partes izquierdas de un SRT se denota como $L(\mathcal{R})$. Toda instancia $\sigma(l)$ de una *lhs* $l \in L(\mathcal{R})$ de una regla es un *redex* (del inglés *red*ucible *ex*pression). El conjunto de ocurrencias de redexes en un término t es $O_{\mathcal{R}}(t)$. Una *lhs* $l \in L(\mathcal{R})$ cuyas variables han sido reemplazadas por símbolos Ω se denomina un *esquema de redex* [KM91]. El conjunto de esquemas de redex de un SRT $\mathcal{R}$ se denota como $L_\Omega(\mathcal{R})$. Un *preredex* es un Ω-término π tal que $\pi \leq l$ para algún esquema de redex $l \in L_\Omega(\mathcal{R})$. El conjunto de *preredexes estrictos* (aquellos preredexes que ni son Ω ni son esquemas de redexes) de un SRT $\mathcal{R}$ se denota como $L^{<}_\Omega(\mathcal{R})$. También definimos el conjunto $LS_\Omega(\mathcal{R}) = \{\pi|_u \mid \pi \in L_\Omega(\mathcal{R}) \wedge u \in \overline{O}_\Omega(\pi)\}$ de subtérminos de los esquemas de redex de un SRT que no son Ω.

Definición 2.3.2 (Reescritura) *Un término t se* reescribe *al término s (a la ocurrencia u), escrito $t \xrightarrow{u}_{\mathcal{R}} s$ (o simplemente $t \to_{\mathcal{R}} s$, $t \xrightarrow{u} s$ ó $t \to s$), si $t|_u = \sigma(l)$ y $s = t[\sigma(r)]_u$, para alguna regla $l \to r \in R$, ocurrencia $u \in O(t)$ y sustitución σ.*

A veces escribiremos también $t \overset{[u,\alpha]}{\to} s$ para poner de manifiesto la regla concreta $\alpha : l \to r$ que se ha empleado. Decimos que el redex $t|_u$ (o que la ocurrencia del redex u) ha sido contraído(a) cuando se ha realizado un paso de reescritura que reduce éste (a la ocurrencia indicada). La *relación de reescritura en un paso* para el SRT $\mathcal{R}$ es $\to$. El cierre reflexivo y transitivo $\to^*$ de $\to$ se denomina la *relación de reescritura* para $\mathcal{R}$. Si $t \to^* s$, decimos que s es un *reducto* de t. El cierre reflexivo, simétrico y transitivo de $\to$ se denota como $\leftrightarrow^*$. La relación $\downarrow$ (*joinability*) se define como sigue: $t{\downarrow}t'$ si existe un s tal que $t \to^* s$ y $t' \to^* s$. La relación de reescritura interna es $\xrightarrow{>\epsilon} = \to \setminus \xrightarrow{\epsilon}$. Una *secuencia (o derivación) de reescritura* (infinita) es una secuencia de reducción (infinita) asociada a la relación de reescritura en un paso $\to$.

2.3.1 Clases de sistemas de reescritura

Una regla $l' \to r'$ es un *renombramiento* de $l \to r$ si existe una sustitución de renombramiento σ tal que $l' = \sigma(l)$ y $r' = \sigma(r)$. Dos reglas (renombradas si es preciso) $\alpha : l \to r$ y $\alpha' : l' \to r'$ definen un *par crítico* $\langle \sigma(l)[\sigma(r')]_u, \sigma(r) \rangle$ si existe una ocurrencia no variable $u \in O_\Sigma(l)$ tal que $l|_u$ y l' unifican con *umg* σ (excluimos el caso trivial $\alpha = \alpha'$ y cuando $u = \epsilon$). Un par crítico se denomina un *cubrimiento* (*overlay*) si $u = \epsilon$. Un par crítico $\langle t, s \rangle$ es *trivial* si $t = s$. Un par crítico $\langle t, s \rangle$ es *convergente* si $t \downarrow s$.

Denotaremos como $\overline{GR}$ al conjunto de SRTs *básicos* (*ground*), es decir, aquéllos cuyas reglas están constituidas por términos básicos. Denotaremos como $\overline{LL}$ al conjunto de SRTs *lineales por la izquierda* (*left-linear*), es decir, los SRTs $\mathcal{R}$ tales que para toda $l \in L(\mathcal{R})$, l es un término lineal. Existe una relación interesante entre los esquemas de redexes y las *lhs*'s de los SRTEs lineales por la izquierda.

Lema 2.3.3 *Sea $\mathcal{R}$ un SRTE lineal por la izquierda. Sea t un término y $l \in L(\mathcal{R})$. Entonces, $t \geq l_\Omega$ si y sólo si t es un redex de l.*

La linealidad por la izquierda es necesaria para demostrar la validez del sentido *sólo si* de la equivalencia establecida en el lema. Por ejemplo, `f(a,b)` $\geq$ `f(`Ω`,`Ω`)`, pero `f(a,b)` no es un redex de `f(x,x)`.

Un SRT lineal por la izquierda sin pares críticos se denomina *ortogonal* (*orthogonal*). Emplearemos *ORT* para referirnos al conjunto de SRTs ortogonales. Si todos los pares críticos de un SRT son cubrimientos triviales, éste se denomina *cuasi ortogonal* (*almost orthogonal*).

Los SRTs cuasi ortogonales exhiben una propiedad interesante: la contracción de un redex situado bajo otro redex no hace que este último deje de serlo (y respecto de la misma *lhs*) [Mid97]. Esto se concreta en el siguiente lema.

Lema 2.3.4 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT cuasi ortogonal, $t \in \mathcal{T}(\Sigma, V)$ tal que $t = \sigma(l)$ para alguna $l \to r \in R$ y $\sigma \in Subst(\Sigma, V)$. Si $u \in O_\mathcal{R}(t) \backslash \{\epsilon\}$, entonces, para todo $s \in \mathcal{T}(\Sigma, V)$, existe una $\sigma' \in Subst(\Sigma, V)$ tal que $t[s]_u = \sigma'(l)$.*

DEMOSTRACIÓN. Si $u \notin O_\Sigma(l)$ es evidente, pues $t = C[t_1, \ldots, t_n]$ donde $l = C[x_1, \ldots, x_n]$ (por ser $\mathcal{R}$ lineal por la izquierda). Si $u \in O_\Sigma(l)$ entonces, dado que $u \neq \epsilon$, habría un par crítico que no sería un cubrimiento, violándose la cuasi ortogonalidad de $\mathcal{R}$. □

Si un SRT sólo tiene pares críticos triviales se denomina *débilmente ortogonal* (*weakly orthogonal*) [KM91].

Un SRT $\mathcal{R}$ es *normal por la izquierda* (*left-normal*) si toda *lhs* $l \in L(\mathcal{R})$ verifica que a la izquierda de toda ocurrencia de un símbolo de función en l no hay símbolos de

variables, es decir, $\forall l \in L(\mathcal{R}), \forall u \in O_\Sigma(l), \forall v \in O(l)\ v \leq_L u \Rightarrow v \in O_\Sigma(l)$. Denotamos la clase de SRTs lineales por la izquierda y normales por la izquierda como $\overline{LN}$ y al subconjunto ortogonal como LN.

Dado un SRT $\mathcal{R} = (\Sigma, R)$, siempre podemos establecer una partición $\Sigma = \mathcal{C} \uplus \mathcal{F}$ de la signatura Σ en un conjunto $\mathcal{C}$ de símbolos *constructores* $c \in \mathcal{C}$ que no tienen ninguna regla de reescritura asociada y un conjunto $\mathcal{F}$ de símbolos *definidos* (u operaciones) $f \in \mathcal{F}$, que quedan definidos como sigue: $\mathcal{F} = \{f \in \Sigma \mid \exists f(\tilde{l}) \to r \in R\}$ y $\mathcal{C} = \Sigma \backslash \mathcal{F}$. Los términos $\delta \in \mathcal{T}(\mathcal{C}, V)$ obtenidos empleando únicamente símbolos constructores y variables se denominan *términos constructores* (o *valores*[4]).

Al imponer a las reglas de reescritura la *disciplina de constructores* obtenemos una clase importante de SRTs, los SRTs *basados en constructores*. Una regla de reescritura $l \to r \in R$ de un SRT $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ cumple la disciplina de constructores si $l = f(\tilde{\delta})$, donde $f \in \mathcal{F}$ y $\tilde{\delta} \in \mathcal{T}(\mathcal{C}, V)^{ar(f)}$. Si todas las reglas de un SRT $\mathcal{R}$ cumplen la disciplina de constructores, se dice que $\mathcal{R}$ es un SRT basado en constructores (SRT BC).

2.3.2 Formas canónicas en reescritura

Un término t es estable[5] (o una forma normal en cabeza, *head-normal form*) si no existe niguna derivación $t = t_1 \to t_2 \to \cdots$ que reescriba t a un redex, es decir, tal que t_i es un redex para algún $i \geq 1$. En particular, llamaremos *forma estable constructora* a todo término $t = c(\tilde{t})$ para algún símbolo constructor c.

Lema 2.3.5 [Mid97] *Sea $\mathcal{R}$ un SRT cuasi ortogonal. Si t es estable y $s \xrightarrow{>\epsilon}{}^* t$, entonces s es estable.*

Lema 2.3.6 [Mid97] *Sea $\mathcal{R}$ un SRT ortogonal. Si un término t se reescribe a un redex, el patrón del primero de tales redexes es único (es decir, si $t \xrightarrow{>\epsilon}{}^* \sigma(l)$ entonces l es único).*

[4] En el contexto de la programación funcional, suele utilizarse la palabra *valor* refiriéndose, únicamente, a los términos constructores básicos (sin variables).

[5] Creemos que la elección de "forma estable" o "término estable" como traducción del inglés *head-normal form* o *root-stable term* en lugar de una traducción literal como "forma normal en cabeza" está justificada por la necesidad de emplear terminología derivada de la noción de forma normal en cabeza al discutir la noción de cómputo que lleva, de forma *necesaria*, a una forma normal en cabeza (*root-needed computation*, ver Sección 2.5). En castellano parece más natural (y conciso) hablar de secuencias estabilizantes, cómputos necesarios para estabilización, etc., que de secuencias que llevan a forma normal en cabeza, etc.

Por otro lado, otros autores se refieren al proceso de obtención de una *head-normal form* a partir de un término como su *estabilización* [Str89].

Un término es *estabilizable* si puede reducirse a una forma estable. Un término t es una *forma normal*[6] si $O_{\mathcal{R}}(t) = \emptyset$; en caso contrario, decimos que t es *reducible*. Un término es *normalizable*[7] si puede reducirse a una forma normal.

Una secuencia de reescritura (posiblemente infinita) $t_1 \to t_2 \to \cdots \to t_n \to \cdots$ es *estabilizante* (*normalizante*) si existe un $i \geq 1$ tal que t_i es una forma estable (normal). Un SRT es *normalizante* si todo término es normalizable.

2.3.3 Multiderivaciones

En *reescritura paralela*, una *multiderivación elemental* contrae, de forma simultánea, un conjunto U de ocurrencias disjuntas de redexes en t y esto se denota como $t \xrightarrow{U} s$ [HL91]. Escribimos $t \Vdash s$ cuando la información concreta acerca de las ocurrencias de los redexes contraídos no sea relevante. Denotamos la longitud de una multiderivación $A : t \Vdash^* s$ como $|A|$.

Observación 2.3.7 *Cualquier derivación $t \xrightarrow{u_1} t_2 \to \cdots \to t_n \xrightarrow{u_n} t_{n+1}$ puede ser considerada como una multiderivación $t \xrightarrow{\{u_1\}} t_2 \to \cdots \to t_n \xrightarrow{\{u_n\}} t_{n+1}$, lo que significa que podemos tratar las derivaciones como si fueran multiderivaciones. De forma análoga, cualquier multiderivación elemental $t \xrightarrow{U} s$ con $U = \{u_1, \ldots, u_n\}$, $n \geq 0$, tiene asociada una derivación equivalente $t \to^* s$ dada por $t = t_1 \xrightarrow{u_1} t_2 \to \cdots \to t_n \xrightarrow{u_n} t_{n+1} = s$ (puesto que las ocurrencias de los redexes considerados en cada paso elemental de una multiderivación son disjuntas, el orden de los pasos elementales en esta secuencia correspondiente a $t \xrightarrow{u} s$ es irrelevante), lo que significa que también podemos considerar las multiderivaciones como derivaciones.*

Dadas las derivaciones elementales $t \xrightarrow{u} t'$ y $t \xrightarrow{v} t''$ (que también son multiderivaciones elementales si escribimos $t \xrightarrow{\{u\}} t'$ y $t \xrightarrow{\{v\}} t''$), el *parallel moves lemma* proporciona una forma canónica de definir multiderivaciones elementales, denotadas $t' \xrightarrow{U} s$ y $t'' \xrightarrow{W} s$ de forma que ambas *convergen* (en un paso) a un reducto común s [HL91].

Definición 2.3.8 (Residuo [HL91]**)** *Sea $\mathcal{R} = (\Sigma, R)$ un SRT ortogonal y $A : t \xrightarrow{[u,\alpha]} s$ una derivación elemental, donde $\alpha : l \to r \in R$. Sea $v \in O_{\mathcal{R}}(t)$. El conjunto $v \backslash A$*

[6] En el caso de la reescritura incondicional, esta definición de forma normal coincide con la de forma $\to$-normal derivada de considerar $\to$ como una relación de reducción (ver la Sección 2.1.2).

[7] Empleamos este vocablo para traducir el concepto referido en la literatura como término *weakly normalizing* [Klo92] pues la expresión 'normalizable' manifiesta suficientemente que se trata de un término susceptible de ser reducido a una forma normal, aunque ello no tiene porqué suceder con *toda* secuencia de reducción que arranque en dicho término.

de residuos *del redex* $t|_v$ *por* A *es un subconjunto de* $O(s)$ *definido como sigue:*

$$v\backslash A = \begin{cases} \emptyset & si\ v = u \\ \{v\} & si\ v \parallel u \quad (i) \\ \{v\} & si\ v < u \quad (ii) \\ \{u.w_1.v_1 \mid r|_{w_1} = x\} & si\ v = u.w.v_1\ y\ l|_w = x \in V \quad (iii) \end{cases}$$

Para cualquier derivación no elemental A y ocurrencia v de un redex en un término t, definimos $v\backslash A$ como sigue: $v\backslash 0 = \{v\}$ (0 es la derivación vacía), y $v\backslash(AB) = \bigcup_{w \in v\backslash A} w\backslash B$. El cálculo de los residuos $v\backslash A$ de una occurrencia redex v se extiende a conjuntos de ocurrencias disjuntas de redexes como sigue: $U\backslash A = \bigcup_{v \in U} v\backslash A$. Dadas dos derivaciones A y B, cuyo término inicial es t y siendo B una multiderivación elemental que contrae el conjunto $U \subseteq O_{\mathcal{R}}(t)$, la derivación *residual* $B\backslash A$ de B por A es la multiderivación elemental que contrae el conjunto $U\backslash A$.

Lema 2.3.9 (Parallel moves lemma [HL91]**)** *Sea* $\mathcal{R}$ *un SRT ortogonal. Sean* A *y* B *multiderivaciones elementales que parten del mismo término* t*. Entonces,* $B(A\backslash B)$ *y* $A(B\backslash A)$ *son multiderivaciones que parten de* t *y llevan al mismo término* s *y, para toda ocurrencia* $u \in O_{\mathcal{R}}(t)$*, tenemos* $u\backslash B(A\backslash B) = u\backslash A(B\backslash A)$*.*

2.3.4 Reescritura infinita

Existe una noción estándar de distancia entre dos términos que puede emplearse para analizar la convergencia de las secuencias de reescritura infinitas [AN80, Cou83]. Dados dos términos t y s, el mayor número natural k tal que todos los nodos de t y s de profundidad menor o igual a k están etiquetados con el mismo símbolo viene definido por la función $pmc(t,s)$, (*p*rofundidad *m*áxima *c*omún) donde:

$$pmc(t,s) = \begin{cases} 0 & \text{si } root(t) \neq root(s) \\ 1 + min\{pmc(t_i,s_i) \mid 1 \leq i \leq ar(f)\} & \text{si } t = f(\tilde{t}) \text{ y } s = f(\tilde{s}) \end{cases}$$

Una distancia $d : \mathcal{T}(\Sigma,V) \times \mathcal{T}(\Sigma,V) \rightarrow \mathbb{R}$ entre términos finitos se define como sigue:

$$d(t,s) = \begin{cases} 0 & \text{si } t = s \\ 2^{-pmc(t,s)} & \text{en otro caso} \end{cases}$$

La compleción métrica del espacio métrico $(\mathcal{T}(\Sigma,V),d)$ es el conjunto $\mathcal{T}^{\infty}(\Sigma,V)$ de *términos infinitos*[8]. Por tanto, toda secuencia de reescritura infinita de Cauchy que

[8] El concepto de término infinito puede definirse también, de forma equivalente, como supremo de una secuencia creciente de términos finitos en el conjunto ordenado $(\mathcal{T}(\Sigma \cup \{\Omega\},V),\leq)$. Como quiera que dicho conjunto no es un orden parcial completo, no toda secuencia creciente de términos tiene un supremo que identificar con un término infinito. Sin embargo, también existe un procedimiento estándar (la compleción de ideales) para obtener un orden parcial completo $(\mathcal{T}^{\infty}(\Sigma \cup \{\Omega\},V),\leq)$ a partir de $(\mathcal{T}(\Sigma \cup \{\Omega\},V),\leq)$ [Blo76, Cou80, WWT78].

sea convergente tiene un límite que es o bien un término finito o bien un término infinito. Kennaway et al. [KKSV95] han desarrollado la noción de secuencia de reescritura *fuertemente convergente* (para SRTEs). En dichas secuencias, la profundidad de los redexes contraídos tiende a infinito.

Definición 2.3.10 (Secuencia fuertemente convergente [Mid97]) *Una secuencia de reescritura infinita* $t_1 \to t_2 \to \cdots$ *es fuertemente convergente si, para todo* $d \geq 0$*, existe un índice* $i \geq 1$ *tal que la profundidad de todo redex contraído en la subsecuencia* $t_i \to t_{i+1} \to \cdots$ *es, al menos,* d*.*

Todas las secuencias de reducción finitas son fuertemente convergentes. Adviértase que toda secuencia infinita fuertemente convergente $t_1 \to t_2 \to \cdots$ tiene un límite $\bar{t}_\omega$ que, necesariamente, es un término infinito. Si $\bar{t}_\omega$ es una forma normal, entonces, para todo $d \geq 0$, existe un índice $i \geq 0$ tal que la profundidad de todo subtérmino no estable en t_i es, al menos, d.

Definición 2.3.11 [Mid97] *Una secuencia de reescritura se denomina* infinitamente normalizante *si es fuertemente convergente a una forma normal (posiblemente infinita).*

Decimos que un SRTE es *infinitamente normalizante* si todo término t admite una secuencia de reducción infinitamente normalizante.

2.3.5 Terminación y confluencia

Un SRT $\mathcal{R}$ se llama confluente si $\to_\mathcal{R}$ es confluente. Un SRT se llama terminante si $\to_\mathcal{R}$ es terminante. Si un SRT es confluente y terminante, se denomina canónico[9].

De entre los métodos de demostración de la terminación de un SRT, recordamos la técnica que emplea un tipo especial de órdenes sobre términos, los (*rpos*).

Si $>$ es un orden estricto sobre un conjunto A, se define la *extensión lexicográfica* $>^{lex}$ de $>$ sobre tuplas de elementos como sigue:

$$(a_1, a_2, \ldots, a_n) \ >^{lex} \ (b_1, b_2, \ldots, b_n)$$

si y sólo si $a_1 > b_1$, o bien $a_1 = b_1$ y

$$(a_2, \ldots, a_n) \ >^{lex} \ (b_2, \ldots, b_n).$$

recursive path orderings

[9] Algunos autores utilizan 'convergente' [BD94, DO90, Jou94] (o 'completo' [Klo92, Gra96]) para referirse a los SRTs confluentes y terminantes y reservan la palabra 'canónico' para referirse a los SRTs convergentes con algunos requerimientos adicionales [BD94, Jou94].

Definición 2.3.12 (Recursive path ordering [Der87]) *Sea Σ una signatura y $>$ un orden estricto (llamado* precedencia*) sobre Σ. Se define un orden $>_{rpo}$ sobre $\mathcal{T}(\Sigma, V)$ como sigue:*

$$t >_{rpo} s$$

si y sólo si $t = f(t_1, \ldots, t_m)$ y

1. *$t_i = s$ o bien $t_i >_{rpo} s$ para algún t_i, con $1 \leq i \leq m$ o*
2. *$s = g(s_1, \ldots, s_n)$, $t >_{rpo} s_i$ para todo i, $1 \leq i \leq n$ y o bien*
   (a) *$f > g$, o*
   (b) *$f = g$ y $(t_1, \ldots, t_n) >_{rpo}^{lex} (s_1, \ldots, s_n)$*

La propiedad interesante del orden $>_{rpo}$ es que si la precedencia $>$ empleada para definirlo es un orden bien fundado, entonces $>_{rpo}$ puede emplearse para demostrar la terminación de un SRT.

Teorema 2.3.13 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $>$ una precedencia bien fundada sobre Σ. Si*

$$l >_{rpo} r$$

para toda regla de reescritura $l \to r \in R$, entonces $\mathcal{R}$ es terminante.

El Teorema 2.3.13 nos permitirá demostrar la terminación de algunos de los SRTs mostrados en los ejemplos, principalmente en el Capítulo 5.

Con relación a la demostración de la propiedad de confluencia de un SRT, es un hecho bien conocido que los SRTs (débilmente) ortogonales son confluentes [HL91, Klo92].

2.4 Estrategias de reescritura

Una estrategia de reescritura *determinista*[10] monopaso (multipaso) para un SRT $\mathcal{R} = (\Sigma, R)$ es una función $\mathbb{F}_{\mathcal{R}} : \mathcal{T}(\Sigma, V) \to \mathcal{T}(\Sigma, V)$ que verifica

1. $\mathbb{F}_{\mathcal{R}}(t) = t$ si t es una forma normal y
2. $t \to_{\mathcal{R}} \mathbb{F}_{\mathcal{R}}(t)$ ($t \to_{\mathcal{R}}^{+} \mathbb{F}_{\mathcal{R}}(t)$) en otro caso.

Si un término t que no es una forma normal verifica $s = \mathbb{F}_{\mathcal{R}}(t)$ para algún término s, escribiremos $t \to_{\mathbb{F}_{\mathcal{R}}} s$ ($t \to_{\mathbb{F}_{\mathcal{R}}}^{+} s$). Una secuencia de reducción $t_1 \to t_2 \to \cdots$ se denomina una $\mathbb{F}_{\mathcal{R}}$-secuencia si, para todo $i \geq 1$, $t_i \to_{\mathbb{F}_{\mathcal{R}}} t_{i+1}$ ($t_i \to_{\mathbb{F}_{\mathcal{R}}}^{+} t_{i+1}$).

[10] Existen otras formalizaciones de la noción de estrategia donde el determinismo no es asumido implícitamente [BEGKPS87, Mid97].

Observación 2.4.1 (Estrategias monopaso y multipaso) *Siguiendo a Antoy y Middeldorp [AM96] queremos recalcar la importancia de diferenciar entre estrategias monopaso y multipaso, de acuerdo con la noción de estrategia introducida. Tal como señalan los autores, dado que una estrategia descrita como una función de términos en términos no tiene 'memoria' (no almacena información acerca de los pasos de reescritura realizados previamente)* no podremos simular una estrategia multipaso mediante una estrategia monopaso. *Los autores subrayan, en particular, la imposibilidad de simular la estrategia* parallel-outermost*: consideremos el SRT*

$$\texttt{f(x,x)} \to \texttt{x}$$
$$\texttt{a} \to \texttt{b}$$

y el término `f(f(a,a),a)`*. La derivación*

$$\texttt{f}(\underline{\texttt{f(a,a)}},\underline{\texttt{a}}) \to \texttt{f}(\underline{\texttt{a}},\texttt{b}) \to \underline{\texttt{f(b,b)}} \to \texttt{b}$$

procede de acuerdo con la estrategia parallel-outermost*. Si intentamos emularla mediante una estrategia monopaso, por ejemplo mediante una estrategia* leftmost-outermost*, es decir, reduciendo, de entre los más externos, el redex situado en el extremo izquierdo, deberíamos tener esta secuencia de reducción:*

$$\texttt{f}(\underline{\texttt{f(a,a)}},\texttt{a}) \to \texttt{f}(\texttt{a},\underline{\texttt{a}}) \to \texttt{f}(\underline{\texttt{a}},\texttt{b}) \to \underline{\texttt{f(b,b)}} \to \texttt{b}$$

cuyos dos primeros pasos implementan el primer multipaso de la derivación parallel outermost *anterior. Pero observemos que ésta* no *es una derivación* leftmost-outermost *que procedería como sigue:*

$$\texttt{f}(\underline{\texttt{f(a,a)}},\texttt{a}) \to \underline{\texttt{f(a,a)}} \to \underline{\texttt{a}} \to \texttt{b}$$

Esto significa que el estudio de las estrategias multipaso no *es reducible al estudio de las estrategias monopaso.*

Una estrategia $\mathbb{F}_{\mathcal{R}}$ es estabilizante (normalizante) si para todo término t estabilizable (normalizable) existe una $\mathbb{F}_{\mathcal{R}}$-secuencia estabilizante (normalizante).

Observación 2.4.2 (Estrategias secuenciales y paralelas) *En estrecha relación con la dicotomía entre estrategia monopaso y estrategia multipaso (y a veces confundida con ella) se encuentra la distinción entre estrategia secuencial y paralela. Sin embargo, existe una diferencia de matiz importante que normalmente está asociada a alguna noción de eficiencia o utilidad de la estrategia en cuestión. Tal es la noción de reducción normalizante o la de reducción necesaria (*call-by-need*). Como señalan varios autores, los sistemas de cómputo basados en la reescritura no están (casi) nunca 'obligados' a efectuar reducciones secuenciales*[11] *[O'D85]. Es más frecuente encontrar que ciertos sistemas de reducción se ven forzados a ejecutar sus programas en paralelo porque no es posible definir una estrategia de reducción secuencial que garantice un*

[11] Aunque a veces una estrategia paralela puede fracasar a la hora de obtener una forma normal donde una estrategia secuencial no lo hace (ver el Ejemplo 7.4 de [AM96]).

comportamiento correcto (normalizante u óptimo). Este es el sentido asignado a la palabra secuencialidad *en los trabajos de Huet y Lévy y otros, tanto en el campo de la reescritura como en el λ-cálculo [HL79, HL91]. Las estrategias secuenciales son monopaso. El aspecto crucial de éstas es la* posibilidad *de definirlas (recordemos, persiguiendo alcanzar un comportamiento* correcto *para éstas) y ese fue el problema principal analizado por Huet y Lévy en [HL79, HL91].*

Observación 2.4.3 (Estrategias secuenciales sin anticipación) *Otro aspecto de interés en el análisis de las estrategias de reescritura secuenciales es el problema de definir éstas de forma efectiva y empleando la menor cantidad de información posible acerca del 'futuro' de las derivaciones que pueden partir del término sobre el que estamos decidiendo cómo realizar el siguiente paso de reducción. Esto se conoce como el problema de definir estrategias de reescritura sin anticipación (*look-ahead*). Desde un punto de vista formal, esto puede formularse como el problema de definir estrategias de reescritura sin tener en cuenta toda la información de las partes derechas (que podemos entender que representan "el futuro" de la derivación). Dicho problema fue resuelto por Huet y Lévy mediante la introducción del concepto de secuencialidad fuerte [HL91, KM91] (ver la Sección 2.5.3).*

En este trabajo emplearemos la palabra 'secuencial' en lugar de monopaso y la palabra 'paralela' en vez de multipaso al referirnos a estrategias de reescritura.

Descomposición de estrategias de reescritura

Cualquier estrategia de reescritura (que por comodidad asumiremos paralela para formalizar la propiedad siguiente) $\mathbb{F}_{\mathcal{R}}$ para un SRT $\mathcal{R} = (\Sigma, R)$ puede descomponerse en dos componentes: $\mathbb{F}_{\mathcal{R}} = (\mathbb{P}_{\mathcal{R}}, \mathbb{R}_{\mathcal{R}})$. El componente $\mathbb{P}_{\mathcal{R}} : \mathcal{T}(\Sigma, V) \to [Occ]$ selecciona la lista de ocurrencias de redexes que deben reducirse (una lista unitaria cuando la estrategia es secuencial). El componente $\mathbb{R}_{\mathcal{R}} : \mathcal{T}(\Sigma, V) \to [R]$ selecciona la lista de reglas que deben aplicarse [AM96]. Las funciones $\mathbb{P}_{\mathcal{R}}$ y $\mathbb{R}_{\mathcal{R}}$ están sometidas a las siguientes restricciones:

1. $\mathbb{P}_{\mathcal{R}}(t) = [\,]$ si y sólo si $O_{\mathcal{R}}(t) = \emptyset$, es decir, la estrategia siempre reduce algo, salvo cuando t es una forma normal ($O_{\mathcal{R}}(t) = \emptyset$).

2. Si $\mathbb{P}_{\mathcal{R}}(t) = [u_1, \ldots, u_m]$ entonces, asumiendo que $\mathbb{R}(t) = [\alpha_1, \ldots, \alpha_n]$, se tiene que $m = n$, $\{u_1, \ldots, u_n\} \subseteq O_{\mathcal{R}}(t)$, $u_i \parallel u_j$ para $1 \leq i, j \leq n$, $i \neq j$, y $t|_{u_i} = \sigma(l_i)$ para la regla $\alpha_i : l_i \to r_i$ cuando $1 \leq i \leq n$, y $\mathbb{F}_{\mathcal{R}}(t) = t[\sigma(r_1)]_{u_1} \cdots [\sigma(r_n)]_{u_n}$.

Para SRTs débilmente ortogonales, la estrategia $\mathbb{F}_{\mathcal{R}}$ viene determinada por completo por el componente $\mathbb{P}_{\mathcal{R}}$ [AM96], aunque no de forma unívoca en el caso general. Sólo si el SRT $\mathcal{R}$ es ortogonal, la elección de un redex concreto por $\mathbb{P}_{\mathcal{R}}$ fija una y sólo

una regla seleccionada por $\mathbb{R}_{\mathcal{R}}$ para reducir ese redex. En el caso de los SRTs cuasi ortogonales y débilmente ortogonales todavía persiste cierto indeterminismo, aunque se asegura que, cuando se dé la posibilidad de aplicar varias reglas, el término que reemplazará al redex estará unívocamente determinado sea cual sea la regla seleccionada para reducirlo. Por tanto, el indeterminismo en la elección de una regla de reescritura puede romperse sin más que imponer un orden total arbitrario sobre el conjunto de las reglas R del SRT y escoger la primera regla de acuerdo con ese orden.

Por comodidad, en lugar de mantener la formulación de los componentes $\mathbb{P}_{\mathcal{R}}$ y $\mathbb{R}_{\mathcal{R}}$ de una estrategia como funciones que devuelven listas de ocurrencias y listas de reglas respectivamente, supondremos que devuelven conjuntos de ocurrencias y conjuntos de reglas. Esto no supone ningún problema cuando los SRTs involucrados son débilmente ortogonales.

Estrategias estabilizantes y estrategias normalizantes

Bajo ciertas condiciones, las estrategias estabilizantes pueden ser empleadas para obtener la forma normal de un término. Middeldorp ya advirtió sobre la imposibilidad de utilizar *cualquier* estrategia de estabilización como estrategia normalizante: si queremos asegurar esto, debemos restringirnos a las estrategias de estabilización *independientes del contexto* (*context-free*).

Definición 2.4.4 [Mid97] *Una estrategia de reducción secuencial $\mathbb{F}_{\mathcal{R}}$ se denomina* independiente del contexto *si para toda forma estable $t = f(t_1, \ldots, t_i, \ldots, t_k)$ e índice $i \in \{1, \ldots, k\}$, tal que $t \rightarrow_{\mathbb{F}_{\mathcal{R}}} f(t_1, \ldots, t'_i, \ldots, t_k)$, se verifica que $t_i \rightarrow_{\mathbb{F}} t'_i$.*

2.4.1 Estrategias de reducción infinita

En esta sección presentamos las definiciones y resultados sobre estrategias de reescritura infinita que emplearemos en la tesis.

Definición 2.4.5 [Mid97] *Una secuencia de reescritura infinita que no es infinitamente normalizante se denomina* perpetua.

Una estrategia de reescritura $\mathbb{F}_{\mathcal{R}}$ para un SRT $\mathcal{R}$ se llama infinitamente normalizante *si no existen secuencias de $\mathbb{F}_{\mathcal{R}}$-reescritura perpetuas que arranquen de términos que admiten una secuencia de reescritura infinita.*

Definición 2.4.6 [Mid97] *Una secuencia de reescritura infinita $t_1 \rightarrow t_2 \rightarrow \cdots$ se denomina* equitativa *(*fair*) si, para todo $i \geq 1$ y todo subtérmino maximal no estable s de t_i que tiene un reducto estable, existe un $j \geq i$ tal que la ocurrencia del redex contraído en el paso $t_j \rightarrow t_{j+1}$ está por debajo de la ocurrencia de s en t_i.*

Una estrategia de reescritura $\mathbb{F}_{\mathcal{R}}$ para un SRT $\mathcal{R}$ se denomina *infinitamente normalizante y equitativa* (*infinitary fair-normalizing*) si no hay $\mathbb{F}_{\mathcal{R}}$-secuencias de reescritura perpetuas y equitativas que parten de términos que admiten una secuencia de reescritura infinitamente normalizante. Pese a su nombre, pues, las estrategias infinitamente normalizantes y equitativas, no tienen porqué ser equitativas.

Teorema 2.4.7 [Mid97] *Sea $\mathcal{R}$ un SRT confluente. Toda estrategia de reducción estabilizante e independiente del contexto para $\mathcal{R}$ es infinitamente normalizante y equitativa.*

Sin el requerimiento de equidad para una estrategia estabilizante, independiente del contexto e infinitamente normalizante y equitativa, no siempre se obtiene una forma normal, sino sólo una parte estable infinita.

Teorema 2.4.8 [Mid97] *Sea $\mathbb{F}_{\mathcal{R}}$ una estrategia de reescritura estabilizante e independiente del contexto para un SRT confluente $\mathcal{R}$. El límite de toda $\mathbb{F}_{\mathcal{R}}$-secuencia de reescritura infinita fuertemente convergente contiene una parte estable infinita.*

2.5 Reducciones necesarias

Un redex en un término t es *necesario* (*needed*) si es contraído, él mismo o alguno de sus residuos[12], en cualquier secuencia de normalización de t [HL91, KM91]. El resultado fundamental establecido por Huet y Lévy es el siguiente.

Teorema 2.5.1 [HL91] *Sea $\mathcal{R}$ un SRT ortogonal. Todo término que no es una forma normal contiene un redex necesario para su normalización.*

Este teorema abre la posibilidad de definir estrategias necesarias, es decir, que sólo reduzcan redexes necesarios. Las secuencias de reducción que sólo contraen redexes necesarios se denominan secuencias de *reducción necesaria*. Estas secuencias exhiben una propiedad importante[13].

[12] Recientemente Durand y Middeldorp han propuesto una noción de redex necesario que es independiente del concepto de residuo de un redex. Puesto que la definición de residuo de un redex genera algunos problemas cuando los SRTs considerados no son ortogonales (ver [O'D95]), al liberar la definición de redex necesario del concepto de residuo, los autores consiguen aplicarla a SRTs generales. Ambas nociones son equivalentes para SRTs ortogonales [DM97].

[13] En contraste con el análisis realizado por Huet y Lévy, en la teoría desarrollada por Durand y Middeldorp no existen, por el momento, resultados similares a los Teoremas 2.5.1 y 2.5.2 que garanticen la existencia de redexes necesarios (según la nueva noción de redex necesario) en el seno de términos reducibles ni tampoco que la reducción de dichos redexes sea normalizante. Esto complica la definición de estrategias de reducción necesarias y, aún en el caso de que pudieramos hacerlo, el garantizar que sean normalizantes.

Teorema 2.5.2 [HL91] *Sea $\mathcal{R}$ un SRT ortogonal y t un término normalizable. Toda secuencia de reducción repetitiva de redexes necesarios que parta de t es normalizante.*

La reducción necesaria no es útil a la hora de obtener la forma estable o la forma normal infinita de un término puesto que, por definición, todo redex en un término que no tiene forma normal es necesario.

Ejemplo 2.5.3 *Consideremos el SRT $\mathcal{R}$ [Mid97]:*

$\mathtt{f(x)} \rightarrow \mathtt{g(f(x))}$

$\mathtt{a} \rightarrow \mathtt{a}$

El término `f(a)` *no tiene forma normal. Por tanto, el redex* `a` *es necesario. Sin embargo, la reducción reiterada de dicho redex:*

$\mathtt{f(\underline{a})} \rightarrow \mathtt{f(\underline{a})} \rightarrow \cdots$

no produce la forma normal infinita `g(g(g(...)))`*. Aún más, ni siquiera obtiene una forma normal en cabeza de* `f(a)` *como* `g(f(a))`*.*

La noción de *cómputo necesario para la estabilización* (*root-needed computation*) ofrece un marco formal adecuado para definir secuencias de reducción estabilizantes, normalizantes e infinitamente normalizantes [Mid97]. Un redex contenido en un término t es necesario para su estabilización si es contraído, él mismo o alguno de sus residuos, en cualquier secuencia de reducción que, partiendo de t, obtenga una forma estable suya. De forma análoga al resultado establecido por Huet y Lévy para los cómputos necesarios para normalización (Teorema 2.5.1), Middeldorp demuestra lo siguiente.

Teorema 2.5.4 [Mid97] *Sea $\mathcal{R}$ un SRT ortogonal. Todo término no estable contiene un redex necesario para su estabilización.*

Los términos estables no contienen redexes necesarios para su estabilización y todo redex presente en un término que no tiene forma estable es (trivialmente) necesario para estabilizarlo.

Existe una relación clara entre ambas nociones de cómputo necesario (para normalización y para estabilización): los redexes necesarios para estabilizar subtérminos maximales no estables son necesarios para normalizar el término que los contiene.

Dado que para SRTs ortogonales generales no es posible decidir si un redex contenido en un término es necesario para normalizar éste, Sekar y Ramakrishnan sugirieron asociar a cada término un *conjunto necesario de redexes* (*necessary set of redexes*) para la normalización en lugar de un solo redex necesario [SR93]. Este concepto proporciona una noción significativa de cómputo necesario para SRTs cuasi ortogonales. Un conjunto de necesario redexes (para la normalización) es un conjunto de redexes

tal que uno de los redexes de dicho conjunto, o alguno de sus residuos, debe ser reducido en cada secuencia de reducción normalizante[14]. De nuevo tenemos algunos resultados fundamentales acerca de esta nueva idea.

Teorema 2.5.5 [SR93] *Sea $\mathcal{R}$ un SRT BC cuasi ortogonal. Todo término que no es una forma normal contiene un conjunto necesario de redexes para su normalización.*

Las secuencias de reducción que sólo contraen conjuntos necesarios de redexes se denominan secuencias de *reducción de conjuntos necesarios*. Estas secuencias exhiben una importante propiedad.

Teorema 2.5.6 [SR93] *Sea $\mathcal{R}$ un SRT BC cuasi ortogonal y t un término normalizable. Toda secuencia de reducción repetitiva de conjuntos necesarios de redexes que parta de t es normalizante.*

Sekar y Ramakrishnan proporcionaron un algoritmo para obtener conjuntos necesarios de redexes para SRT BCs ortogonales. Por tanto, su noción de necesidad se puede computar de forma efectiva en el caso general (aunque el cálculo es complejo).

Middeldorp aplica la noción de conjunto necesario de redexes a la *estabilización* de términos. Un *conjunto necesario de redexes para la estabilización* es un conjunto de redexes tal que, al menos uno de los redexes de ese conjunto, o uno de sus residuos, será reducido en cada derivación a una forma estable. La contracción reiterada de conjuntos necesarios de redexes para la estabilización se denomina reducción *de conjuntos necesarios para la estabilización.*

Teorema 2.5.7 [Mid97] *Sea t un término estabilizable. No existen secuencias de reducción paralela que, partiendo de t, realicen un número infinito de pasos de reducción de conjuntos necesarios para su estabilización.*

La reducción de conjuntos necesarios para la estabilización es estabilizante para SRTs cuasi ortogonales. En particular, la contracción repetida de redexes necesarios para la estabilización es estabilizante para SRTs ortogonales.

2.5.1 Ω-reducción

Se define la relación de reducción $\rightarrow_\Omega$ (llamada Ω-reducción, [KM91]) sobre Ω-términos como sigue: $C[t] \rightarrow_\Omega C[\Omega]$ si $t \neq \Omega$ y existe un esquema de redex $l \in L_\Omega(\mathcal{R})$ tal que $t \uparrow l$, es decir, existe un $s \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$, tal que $t \leq s$ y $l \leq s$. Esta definición es perfectamente válida si $\mathcal{R}$ es un SRTE.

[14] Por tanto, no es lo mismo un conjunto de redexes necesario (que puede contener redexes necesarios y otros que no lo son) que un conjunto de redexes necesarios (un conjunto que *sólo* contiene redexes necesarios).

Lema 2.5.8 *Sean $t, l \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$. Si $t \uparrow l$, existe un contexto máximo $C[\,]$, tal que $t = C[t_1, \ldots, t_n]$, $l = C[l_1, \ldots, l_n]$ y, para todo j, $1 \leq j \leq n$, o bien $t_j \leq l_j$ (y entonces $t_j = \Omega$) o bien $t_j \geq l_j$ (y entonces $l_j = \Omega$).*

La relación de reducción $\rightarrow_\Omega$ es confluente y terminante para SRTs arbitrarios [HL91, KM91] y dicho resultado se extiende sin dificultad a SRTEs. Sea $\omega(t)$ la forma $\rightarrow_\Omega$-normal de t. Claramente, $\omega(\Omega) = \Omega$ y, siempre que t es un redex, $\omega(t) = \Omega$.

Proposición 2.5.9 [KM91] *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE. Sea $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$. Entonces,*

1. $\omega(t) \leq t$
2. $\omega(t) = \omega(t[\omega(t|_u)]_u)$
3. $t \leq s \Rightarrow \omega(t) \leq \omega(s)$
4. $t \rightarrow^* s \Rightarrow \omega(t) \leq \omega(s)$

DEMOSTRACIÓN. Aunque las propiedades 1 a 3 fueron demostradas en [KM91] para SRTs, la demostración no depende de la ausencia de variables extra. Por tanto, las propiedades se extienden a SRTEs sin problemas.

Demostramos el apartado 4 porque lo hemos formulado de manera un tanto diferente a como aparece en [KM91].

Realizamos la demostración por inducción sobre la longitud n de la derivación $t \rightarrow^* s$. Si $n = 0$, es inmediato. Si $n > 0$, sea $t \stackrel{u}{\rightarrow} t' \rightarrow^* s$. De acuerdo con el apartado 2, y puesto que $t = t[\sigma(l)]_u$ para alguna $l \rightarrow r \in R$, tenemos $\omega(t) = \omega(t[\sigma(l)]_u) = \omega(t[\omega(\sigma(l))]_u) = \omega(t[\Omega]_u)$. De acuerdo con el apartado 3, y puesto que $t[\Omega]_u \leq t[\sigma(r)]_u = t'$, tenemos que $\omega(t[\Omega]_u) \leq \omega(t')$. Por tanto, $\omega(t) \leq \omega(t')$. Por la hipótesis de inducción, $\omega(t') \leq \omega(s)$ y, por transitividad de $\leq$, concluimos la demostración. □

2.5.2 Aproximación de formas estables

Dado que es indecidible si un término es estable o no [Ken94, Mid97], debemos buscar aproximaciones a éstas. La Ω-reducción nos ofrece una oportunidad para ello: un término t es (una forma) *fuertemente estable* (*strong head-normal form*) si $\omega(t) \neq \Omega$.

Proposición 2.5.10 *Sea $\mathcal{R}$ un SRTE. Si t es una forma fuertemente estable, entonces t es estable.*

DEMOSTRACIÓN. Por contradicción. Si t es fuertemente estable, entonces $\omega(t) \neq \Omega$. Si t no es estable, existe una derivación $t \rightarrow^* \sigma(l) \rightarrow \sigma(r)$ para alguna regla $l \rightarrow r$ del

SRTE. Por la Proposición 2.5.9, $\omega(t) \leq \omega(\sigma(l)) = \Omega$. Por tanto $\omega(t) = \Omega$, lo cual es una contradicción. □

En general, la afirmación contraria no es cierta, es decir, no toda forma estable es fuertemente estable.

Ejemplo 2.5.11 *Sea el SRT $\mathcal{R}$:*

$$\texttt{f(a)} \rightarrow \texttt{a}$$
$$\texttt{b} \rightarrow \texttt{c}$$

El término `f(b)` *es estable, pero no fuertemente estable:*

$$\texttt{f(b)} \rightarrow_\Omega \texttt{f}(\Omega) \rightarrow_\Omega \Omega$$

y, por tanto, $\omega(\texttt{f(b)}) = \Omega$.

Puesto que comprobar si un término t es fuertemente estable consiste en obtener su forma $\rightarrow_\Omega$-normal $\omega(t)$ (que debido a que $\rightarrow_\Omega$ es confluente y terminante siempre puede obtenerse en tiempo finito) y comprobar que difiere de Ω, en virtud de la Proposición 2.5.10, la estabilidad fuerte constituye una aproximación decidible a la estabilidad.

Un término t es *rígido* (*rigid*) si $\omega(t) = t$ y *frágil* (*soft*) si $\omega(t) = \Omega$ [Klo92]. También decimos que un contexto $C[\,]$ es rígido si $C[\widetilde{\Omega}]$ es un término rígido.

Proposición 2.5.12 [Klo92] *Todo término $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ puede descomponerse de forma única como $t = C[t_1, \ldots, t_n]$ tal que $C[\Omega, \ldots, \Omega]$ es rígido y cada t_i es frágil para $1 \leq i \leq n$.*

Observemos que el contexto $C[\,]$ a que se refiere la Proposición 2.5.12 no contiene ocurrencias de Ω. A continuación demostramos que la rigidez es composicional, un resultado que emplearemos más adelante.

Lema 2.5.13 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $C[\,], C_1[\,], \ldots, C_n[\,]$ contextos rígidos. Entonces, $C[C_1[\,], \ldots, C_n[\,]]$ es un contexto rígido.*

Demostración. Por contradicción. Si $t = C[C_1[\,], \ldots, C_n[\,]]$ no es rígido, puesto que los $C_1[\,], \ldots, C_n[\,]$ lo son, concluimos que existe una ocurrencia $u \in O_\Sigma(C[\widetilde{\Omega}])$ tal que $t|_u$ es compatible con algún esquema de redex $l \in L_\Omega(\mathcal{R})$. Por tanto, dado que $C[\widetilde{\Omega}]|_u \leq t|_u$, también tenemos $C[\widetilde{\Omega}]|_u \uparrow l$. Por tanto, $\omega(C[\widetilde{\Omega}]) < C[\widetilde{\Omega}]$ lo cual significa que $C[\widetilde{\Omega}]$ no es rígido. □

Los términos rígidos pueden considerarse como generalizaciones de las formas estables. Si $t = f(\tilde{t})$ es estable, el símbolo f en cabeza no puede ser eliminado por reducciones posteriores sobre t. De forma similar, si $C[\,]$ es un contexto rígido y existen términos

$t_1, \ldots, t_n$ tales que $t = C[t_1, \ldots, t_n]$, entonces $C[\,]$ representa una parte invariable de t respecto a reducciones ulteriores. Un contexto rígido $C[\,]$ también tiene la siguiente propiedad: si rellenamos los n huecos de $C[\,]$ con formas normales $t_1, \ldots, t_n$, obtenemos una forma normal $C[t_1, \ldots, t_n]$.

Lema 2.5.14 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $C[\,]$ un contexto rígido. Sean $t_1, \ldots, t_n \in \mathcal{T}(\Sigma, V)$. Entonces, $C[t_1, \ldots, t_n]$ es una forma normal si y sólo si cada $t_1, \ldots, t_n$ es una forma normal.*

DEMOSTRACIÓN. La parte *sólo si* se demuestra fácilmente por contradicción, puesto que si hay un t_i, para algún $1 \leq i \leq n$, que no es una forma normal, entonces $t = C[t_1, \ldots, t_n]$ no es una forma normal. Para la parte *si* advertimos que, siendo formas normales, $t_1, \ldots, t_n$ son contextos rígidos. Así pues, por el Lema 2.5.13, $C[t_1, \ldots, t_n]$ es rígido y, por tanto, una forma normal. □

2.5.3 Índices fuertemente secuenciales

Huet y Lévy definieron la noción de redex *fuertemente* secuencial o redex *fuertemente* necesario como una aproximación decidible a los redexes necesarios (para normalización) [HL91]. Los índices[15] son ocurrencias especiales que localizan dichos redexes fuertemente necesarios (para la normalización). Para discriminar qué ocurrencias corresponden a índices y cuáles no, empleamos la Ω-reducción.

La noción de necesidad, tal como la definieron Huet y Lévy, no tiene sentido fuera del contexto de los SRTs ortogonales puesto que, una vez abandonada la ortogonalidad, no puede asegurarse que todo término tenga un redex necesario si bien algunos autores han propuesto extensiones de la noción de necesidad aplicables a SRTs más generales (y también a SRTEs) [DM97, O'D95, SR93]. Sin embargo, la noción de índice en sí misma (desligada de su carácter de aproximación computable de los redexes necesarios) ha sido empleada por Toyama para definir estrategias normalizantes para SRTs lineales por la izquierda con posible presencia de pares críticos [Toy92].

Definición 2.5.15 [JS94, Toy92] *Sea $\mathcal{R} = (\Sigma, R)$ un SRT, $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ y $u \in O_\Omega(t)$. Sea $\bullet$ un nuevo símbolo constante y $t' = t[\bullet]_u$. Decimos que u es un* índice *de t si y sólo si $\omega(t')|_u = \bullet$. Denotamos como $\mathcal{I}(t)$ el conjunto de índices de t: $\mathcal{I}(t) = \{u \in O_\Omega(t) \mid \omega(t[\bullet]_u)|_u = \bullet\}$.*

Observación 2.5.16 *Dado un término $t \in \mathcal{T}(\Sigma, V)$ (sin Ω-ocurrencias), comprobamos si una ocurrencia $u \in O_\mathcal{R}(t)$ de un redex en t corresponde a un índice, aplicando*

[15] El concepto de índice secuencial, que constituye el punto de partida del análisis de computabilidad de la noción de necesidad realizado por Huet y Lévy corresponde al concepto más general (pero técnicamente más complejo) de índice de un predicado secuencial introducida por Kahn y Plotkin [KP78, KP93].

la definición anterior al término $t[\Omega]_u$. En general, diremos que una ocurrencia cualquiera $u \in O(t)$ es un índice si u es un índice de $t[\Omega]_u$.

Algunas propiedades estándar de los índices serán útiles más adelante.

Proposición 2.5.17 [JS94] *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$.*
(1) *Si $u.v \in \mathcal{I}(t)$, entonces $u \in \mathcal{I}(t[\Omega]_u)$ y $v \in \mathcal{I}(t|_u)$.*
(2) *Si $u \in \mathcal{I}(t)$ y $t \leq t'$, entonces $u \in \mathcal{I}(t'[\Omega]_u)$.*

Proposición 2.5.18 [Klo92, KM91] *Sea $\mathcal{R} = (\Sigma, R)$ un SRT. Consideremos el término $t = C[t_1, \ldots, t_n]$ tal que $C[\Omega, \ldots, \Omega]$ es rígido y los t_i, $1 \leq i \leq n$ son frágiles. Sea $t_i = C'[\Omega]_v = t|_u$. Si $v \in \mathcal{I}(t_i)$, entonces $u.v \in \mathcal{I}(t)$.*

Tanto la Definición 2.5.15 como las Proposiciones 2.5.17 y 2.5.18 son aplicables a SRTEs.

2.5.4 Sistemas fuertemente secuenciales

Los sistemas de reescritura fuertemente secuenciales son la clase de SRTs que verifican la siguiente propiedad: en todo término t que no está en forma normal, siempre podemos encontrar un redex situado sobre un índice. Una forma Ω-normal t es un Ω-término sin redexes (es decir, $O_{\mathcal{R}}(t) = \emptyset$) que contiene al menos una ocurrencia de Ω (es decir, $O_{\Omega}(t) \neq \emptyset$). El reemplazamiento de todos los redexes externos en un término t por símbolos Ω produce una forma Ω-normal.

Definición 2.5.19 [JS94] *Un SRT lineal por la izquierda $\mathcal{R}$ es fuertemente secuencial si toda forma Ω-normal contiene un índice.*

Observación 2.5.20 *La secuencialidad fuerte ha sido estudiada con relación a SRTs con variables extra, es decir SRTEs [Com95, DM97, Jac96]. Por tanto, estamos autorizados a considerar SRTEs fuertemente secuenciales (generalizando las Definiciones 2.5.15 y 2.5.19).*

Jouannaud y Sadfi demostraron que la secuencialidad fuerte de los SRTs lineales por la izquierda era decidible [JS94] (y Comon extendió el resultado a SRTEs lineales por la izquierda [Com95]). Escribiremos $\overline{SS}$ para referirnos a la clase de SRTs fuertemente secuenciales y lineales por la izquierda y SS para referirnos al subconjunto ortogonal estudiado por Huet y Lévy [HL91, KM91]. Una estrategia (tal vez no determinista) que siempre contrae redexes situados sobre índices se llama *reducción de índices*. Dichas estrategias sólo pueden ser definidas para SRTEs fuertemente secuenciales puesto que otras clases de SRTEs no garantizan que todo término que no sea una forma normal contiene un redex situado sobre un índice.

La reducción de índices es normalizante para SRTs lineales por la izquierda, fuertemente secuenciales y convergentes a la raíz. Un SRT es *uniformemente convergente a la raíz* (*root-balanced joinable*) si todos sus pares críticos son uniformemente convergentes a la raíz. Un par crítico $\langle t, s\rangle$ es *uniformemente convergente a la raíz* si $t \xrightarrow{\epsilon}_k t'$ y $s \xrightarrow{\epsilon}_k t'$ para algún t' y $k \geq 0$, donde $\xrightarrow{\epsilon}_0$ es la identidad $id_{\mathcal{T}(\Sigma,V)}$ (considerada como relación), es decir, $\xrightarrow{\epsilon}_0 = id_{\mathcal{T}(\Sigma,V)}$ y $\xrightarrow{\epsilon}_{k+1} = \xrightarrow{\epsilon} \circ \xrightarrow{\epsilon}_k$ [Toy92].

Teorema 2.5.21 [Toy92] *La reducción de índices es normalizante para SRTs lineales por la izquierda, fuertemente secuenciales y uniformemente convergentes a la raíz.*

La clase de SRTs lineales por la izquierda y uniformemente convergentes a la raíz incluye los SRTs débilmente ortogonales y, por tanto, los SRTs cuasi ortogonales y los ortogonales. De esta manera, por lo que respecta a las propiedades de normalización de las secuencias de reducción de índices, este teorema incluye el resultado estándar de Huet y Lévy: "*La reducción de índices es normalizante para SRTs ortogonales y fuertemente secuenciales*" [HL91].

Parte I

Propiedades computacionales básicas de la reescritura sensible al contexto

Capítulo 3

La reescritura sensible al contexto

En este capítulo presentamos la restricción de la reescritura que denominamos reescritura sensible al contexto.

La Sección 3.1 define el concepto de función de reemplazamiento, dota de una estructura algebraica básica al conjunto de funciones de reemplazamiento definibles para una signatura dada e introduce el concepto de ocurrencia reemplazante.

La Sección 3.2 estudia las propiedades fundamentales de las ocurrencias reemplazantes sobre todo en conexión con el proceso de reescritura restringida.

La Sección 3.3 define la relación de reescritura sensible al contexto (*rsc*) y establece sus propiedades básicas.

La Sección 3.4 ofrece algunos ejemplos de aplicación.

La Sección 3.5 describe algunas propuestas de restricción de la reescritura similares a la *rsc*.

3.1 Funciones de reemplazamiento y ocurrencias reemplazantes

En la *rsc* se restringen los reemplazamientos que está permitido realizar en las secuencias de reescritura. Dicha restricción se define en dos etapas. En primer lugar, se establecen las restricciones de reemplazamiento a nivel básico, sobre los *símbolos* de función de la signatura. Sobre esta base, se establecen las restricciones a nivel de la *estructura* de los términos que intervienen en las secuencias de reducción.

Las restricciones de reemplazamiento asociadas a las posiciones argumentales de los símbolos de la signatura se imponen mediante una función de reemplazamiento.

Definición 3.1.1 (Función de reemplazamiento) *Sea Σ una signatura. Una función $\mu : \Sigma \to \mathcal{P}(\mathbb{N})$ es una función de reemplazamiento para Σ si, para todo símbolo $f \in \Sigma$, se verifica $\mu(f) \subseteq \mathbb{N}^+_{ar(f)}$.*

Una función de reemplazamiento determina las posiciones argumentales $\mu(f)$ que *pueden* ser reducidas para cada símbolo f de la signatura Σ. No se permiten reducciones sobre argumentos t_i de una llamada a función $f(t_1, \ldots, t_k)$ tales que $i \notin \mu(f)$.

Notación 3.1.2 *Si asumimos un orden arbitrario $<$ sobre los símbolos de la signatura $\Sigma = \{f_1, \ldots, f_n\}$ (por ejemplo, el inherente a la secuencia en que son escritos al presentar la signatura: $f_1 < f_2 < \cdots < f_n$), podemos expresar una función de reemplazamiento como sigue:*

$$\mu = \langle I_1, \ldots, I_n \rangle, \text{ donde } I_j = \mu(f_j) \text{ para } 1 \leq j \leq n$$

Notación 3.1.3 *El conjunto de todas las posibles funciones de reemplazamiento para una determinada signatura Σ se denota como M_Σ.*

En particular, consideraremos el conjunto $M_\emptyset = \{\mu^\perp_\perp\}$ que consta de la única función de reemplazamiento $\mu^\perp_\perp$ que puede definirse sobre una signatura vacía.

Sobre M_Σ puede establecerse una relación de orden parcial $\sqsubseteq$ que es la extensión punto a punto del orden de inclusión $\subseteq$ definido sobre $\mathcal{P}(\mathbb{N})$, el codominio de las funciones que integran M_Σ.

Definición 3.1.4 (Orden sobre funciones de reemplazamiento) *Sea Σ una signatura y $\mu, \mu' \in M_\Sigma$. Se define el orden parcial $\sqsubseteq$ sobre M_Σ como sigue:*

$$\mu \sqsubseteq \mu' \Leftrightarrow \forall f \in \Sigma.\ \mu(f) \subseteq \mu'(f).$$

La corrección de esta definición está garantizada por el empleo de la operación estándar de extensión puntual a una relación de orden conocida, como es la inclusión conjuntista.

Observación 3.1.5 *Intuitivamente, $\mu \sqsubseteq \mu'$ significa que la función de reemplazamiento μ permite menos reemplazamientos sobre los argumentos de los símbolos de la signatura que la función de reemplazamiento μ'. También decimos que μ es una función de reemplazamiento* más restrictiva *que μ'.*

Dado que el conjunto $(\mathcal{P}(\mathbb{N}), \subseteq, \emptyset, \mathbb{N}, \cup, \cap)$ es un retículo completo distributivo, también la extensión $\sqsubseteq$ del orden $\subseteq$ induce en M_Σ un retículo completo distributivo $(M_\Sigma, \sqsubseteq, \mu_\perp, \mu_\top, \sqcup, \sqcap)$. El elemento mínimo $\mu_\perp$ satisface $\mu_\perp(f) = \emptyset$ para todo $f \in \Sigma$. El elemento máximo $\mu_\top$ satisface $\mu_\top(f) = \mathbb{N}^+_{ar(f)}$ para todo $f \in \Sigma$. La operación $\sqcup$, que obtiene el supremo de dos funciones de reemplazamiento, viene dada por

$(\mu \sqcup \mu')(f) = \mu(f) \cup \mu'(f)$, para todo $f \in \Sigma$. La operación $\sqcap$, que obtiene el ínfimo de dos funciones de reemplazamiento, por $(\mu \sqcap \mu')(f) = \mu(f) \cap \mu'(f)$ para todo $f \in \Sigma$. Escribimos $\mu \parallel \mu'$ para indicar que dos funciones de reemplazamiento no son comparables, es decir: $\mu \not\sqsubseteq \mu'$ y $\mu' \not\sqsubseteq \mu$. El siguiente lema será empleado más adelante.

Lema 3.1.6 *Sea Σ una signatura y $\mu, \mu' \in M_\Sigma$. Entonces, $\mu \not\sqsubseteq \mu'$ si y sólo si*

$$\exists f \in \Sigma, \exists i \in \mathbb{N}^+_{ar(f)}.\ i \in \mu(f) \wedge i \notin \mu'(f).$$

DEMOSTRACIÓN.

$$\neg(\exists f \in \Sigma.\exists i \in \mathbb{N}^+_{ar(f)}.\ i \in \mu(f) \wedge i \notin \mu'(f))$$

si y sólo si

$$\forall f \in \Sigma.\forall i \in \mathbb{N}^+_{ar(f)}.\ i \notin \mu(f) \vee i \in \mu'(f)$$

si y sólo si

$$\forall f \in \Sigma.\forall i \in \mathbb{N}^+_{ar(f)}.\ i \in \mu(f) \Rightarrow i \in \mu'(f)$$

si y sólo si

$$\mu \sqsubseteq \mu'.$$

Esto completa la demostración. □

También estamos interesados en relacionar funciones de reemplazamiento asociadas a signaturas distintas.

Definición 3.1.7 *Sean Σ y Σ' dos signaturas. Definimos una operación genérica $\overline{\ } : M_\Sigma \to M_{\Sigma'}$ que proporciona la inserción $\overline{\mu}$ de una función de reemplazamiento $\mu \in M_\Sigma$, en el espacio de funciones de reemplazamiento $M_{\Sigma'}$: dado $f \in \Sigma'$,*

$$\overline{\mu}(f) = \begin{cases} \mu(f) & \text{si } f \in \Sigma \text{ y } ar_\Sigma(f) = ar_{\Sigma'}(f) \\ \emptyset & \text{en otro caso} \end{cases}$$

Es interesante observar que, al considerar una signatura cualquiera Σ, siempre obtenemos $\overline{\mu}^\perp_\perp = \mu_\perp \in M_\Sigma$. Esto motiva el símbolo $\mu^\perp_\perp$ escogido para la única función de reemplazamiento definible sobre una signatura vacía.

Esta operación permite emplear las operaciones algebraicas del retículo de funciones de reemplazamiento para operar con funciones de reemplazamiento de distintas signaturas. Por ejemplo, dadas las funciones de reemplazamiento $\mu \in M_\Sigma$ y $\mu' \in M_{\Sigma'}$, escribimos $\mu \sqcup \mu' \in M_{\Sigma \cup \Sigma'}$ en lugar de $\overline{\mu} \sqcup \overline{\mu'}$. En realidad, estaríamos empleando las operaciones $\overline{\ } : M_\Sigma \to M_{\Sigma \cup \Sigma'}$ y $\overline{\ } : M_{\Sigma'} \to M_{\Sigma \cup \Sigma'}$, así como la operación supremo del retículo $M_{\Sigma \cup \Sigma'}$. Procederemos de esta manera a menos que indiquemos lo contrario.

Observación 3.1.8 *La función* $\overline{\ }$ *también permite emplear funciones de reemplazamiento asociadas a una signatura* Σ *en contextos donde la signatura involucrada es* Σ'*. Usualmente escribimos simplemente* μ*, en lugar de* $\overline{\mu}$*.*

Una función de reemplazamiento establece las restricciones de reemplazamiento asociadas a los símbolos de la signatura. A continuación emplearemos una función de reemplazamiento para introducir las restricciones de reemplazamiento en la estructura de los términos construidos con los símbolos de esa signatura.

Definición 3.1.9 (Ocurrencias reemplazantes) *Sea* Σ *una signatura,* $\mu \in M_\Sigma$ *y* $t \in \mathcal{T}(\Sigma, V)$*. El conjunto* $O^\mu(t)$ *de* ocurrencias μ-reemplazantes *de* t *se define como sigue:*

$$O^\mu(t) = \begin{cases} \{\epsilon\} & si\ t \in V \\ \{\epsilon\} \cup \bigcup_{i \in \mu(f)} i.O^\mu(t_i) & si\ t = f(\tilde{t}) \end{cases}$$

Si $u \in O^\mu(t)$*, decimos que* u *es una ocurrencia* μ*-reemplazante de* t*. También que* $t|_u$ *es un subtérmino* μ*-reemplazante.*

Cuando la función de reemplazamiento involucrada se sobreentiende a partir del contexto discursivo, eliminamos toda mención explícita a ésta, aludiendo simplemente a las *ocurrencias reemplazantes* de un término.

A continuación, se presenta un ejemplo de cálculo del conjunto de ocurrencias reemplazantes de un término.

Ejemplo 3.1.10 *Consideremos el término* $t =$ `if(and(true,false),0+s(0),0)` *y la función de reemplazamiento* μ *definida como sigue:*

$$\mu(\texttt{0}) = \mu(\texttt{true}) = \mu(\texttt{false}) = \mu(\texttt{s}) = \emptyset \quad y \quad \mu(\texttt{if}) = \mu(\texttt{and}) = \mu(\texttt{+}) = \{1\}$$

El conjunto de ocurrencias reemplazantes de t *es* $O^\mu(t) = \{\epsilon, 1, 1.1\}$*.*

A partir de ahora omitiremos las referencias explícitas al valor de una función de reemplazamiento μ cuando se aplica ésta sobre símbolos constantes $f \in \Sigma$. Siendo la aridad de un símbolo constante nula, el único valor posible para $\mu(f)$ es $\mu(f) = \emptyset$ y así lo asumiremos implícitamente.

A partir de la Definición 3.1.9, introducimos los siguientes conceptos y notaciones derivadas.

Notación 3.1.11 *Consideremos un término* t*. El conjunto de ocurrencias* no reemplazantes *es* $\overline{O^\mu}(t) = O(t) \backslash O^\mu(t)$*. El conjunto de ocurrencias de variables reemplazantes es* $O^\mu_V(t) = O_V(t) \cap O^\mu(t)$*. El conjunto de ocurrencias no variables reemplazantes es* $O^\mu_\Sigma(t) = O_\Sigma(t) \cap O^\mu(t)$*. El conjunto de ocurrencias reemplazantes*

de un subtérmino s dentro de un término t es $O^{\mu}_{s}(t) = O_s(t) \cap O^{\mu}(t)$. Escribimos $Var^{\mu}(t)$ para referirnos al conjunto de variables reemplazantes *de t, es decir, $Var^{\mu}(t) = \{x \in Var(t) \mid O^{\mu}_{x}(t) \neq \emptyset\}$. En $Var^{\mu}(t)$ podemos encontrar variables de t situadas tanto sobre ocurrencias reemplazantes como no reemplazantes; el conjunto de variables* exclusivamente *no reemplazantes es $\overline{Var^{\mu}}(t) = Var(t) \backslash Var^{\mu}(t)$.*

Como era de esperar, el orden $\sqsubseteq$ entre funciones de reemplazamiento también ordena (por inclusión) los conjuntos de ocurrencias reemplazantes asociadas a un mismo término, respecto a distintas funciones de reemplazamiento.

Proposición 3.1.12 *Sea Σ una signatura, $\mu, \mu' \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Si $\mu \sqsubseteq \mu'$, entonces $O^{\mu}(t) \subseteq O^{\mu'}(t)$.*

DEMOSTRACIÓN. Por inducción sobre la estructura de t. Si $t \in \Sigma \cup V$, la conclusión es inmediata, puesto que $O^{\mu}(t) = O^{\mu'}(t) = \{\epsilon\}$. Si $t = f(\tilde{t})$, por la hipótesis de inducción, $O^{\mu}(t_i) \subseteq O^{\mu'}(t_i)$ para $1 \leq i \leq ar(f)$. Por la Definición 3.1.9, y dado que $\mu(f) \subseteq \mu'(f)$, se tiene $O^{\mu}(t) = \{\epsilon\} \cup \bigcup_{i \in \mu(f)} i.O^{\mu}(t_i) \subseteq \{\epsilon\} \cup \bigcup_{i \in \mu(f)} i.O^{\mu'}(t_i) \subseteq \{\epsilon\} \cup \bigcup_{i \in \mu'(f)} i.O^{\mu'}(t_i) = O^{\mu'}(t)$. □

3.2 Propiedades de las restricciones de reemplazamiento sensibles al contexto

El análisis de las propiedades inherentes al cálculo del conjunto de ocurrencias reemplazantes de un término (para una determinada función de reemplazamiento) es importante para utilizar dichas restricciones de reemplazamiento en reescritura. A continuación se establece la propiedad más importante de dicho cálculo, su composicionalidad.

Proposición 3.2.1 (Composicionalidad del cálculo de $O^{\mu}(t)$) *Sea Σ una signatura, $\mu \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Sea $u.v \in O(t)$. Entonces, $u.v \in O^{\mu}(t)$ si y sólo si $u \in O^{\mu}(t) \wedge v \in O^{\mu}(t|_u)$.*

DEMOSTRACIÓN. Por inducción sobre la longitud $|u|$ de la ocurrencia u. Si $|u| = 0$, entonces $u = \epsilon$ y, puesto que se cumple $\epsilon \in O^{\mu}(t)$ para todo término t, se tiene: $u.v \in O^{\mu}(t) \Leftrightarrow \epsilon.v \in O^{\mu}(t|_\epsilon) \Leftrightarrow v \in O^{\mu}(t|_\epsilon) \Leftrightarrow \mathsf{true} \wedge v \in O^{\mu}(t|_\epsilon) \Leftrightarrow \epsilon \in O^{\mu}(t) \wedge v \in O^{\mu}(t|_\epsilon)$ $\Leftrightarrow u \in O^{\mu}(t) \wedge v \in O^{\mu}(t|_u)$.

Si $|u| > 0$, sea $u = i.u'$ y $t = f(\tilde{t})$ para algún $f \in \Sigma$. A partir de la definición 3.1.9, podemos escribir $i.u'.v \in O^{\mu}(t)$ si y sólo si $i \in \mu(f)$ y $u'.v \in O^{\mu}(t_i)$. Por la hipótesis de inducción $u' \in O^{\mu}(t_i)$ y $v \in O^{\mu}(t_i|_{u'})$. Puesto que $i \in \mu(f)$, se tiene $u \in O^{\mu}(t)$ y, dado que $t|_u = t_i|_{u'}$, se obtiene también $v \in O^{\mu}(t|_u)$. □

Observación 3.2.2 *La Proposición 3.2.1 permite realizar el cálculo del conjunto de ocurrencias reemplazantes de un término de forma composicional, es decir, a partir de los subtérminos componentes y del contexto donde éstos se ubican. En un sentido estricto, un cómputo sobre un objeto estructurado es composicional si puede obtenerse a partir de subestructuras estrictamente más simples de dicho objeto. En el caso del cálculo del carácter reemplazante de una ocurrencia $u.v$ en $O(t)$, la Proposición 3.2.1 no expresa exactamente esta propiedad, pues exige conocer el carácter reemplazante de u para lo cual debemos calcular $O^{\mu}(t)$.*

La siguiente proposición permite dar pleno sentido al cómputo composicional del carácter reemplazante de las ocurrencias de un término.

Proposición 3.2.3 *Sea Σ una signatura, $\mu \in M_{\Sigma}$ y $t \in \mathcal{T}(\Sigma, V)$. Si $u \in O(t) \cap O(s)$ y $sprefix_t(u) = sprefix_s(u)$, entonces $u \in O^{\mu}(t) \Leftrightarrow u \in O^{\mu}(s)$.*

DEMOSTRACIÓN. Por inducción en la longitud de la ocurrencia u. Si $u = \epsilon$, la conclusión es inmediata, puesto que $\epsilon \in O^{\mu}(t)$ para todo término t. En otro caso, sea $t = f(\tilde{t})$ y $u = i.u'$. Por la Proposición 3.2.1, $u \in O^{\mu}(t) \Leftrightarrow i \in O^{\mu}(t) \wedge u' \in O^{\mu}(t|_i)$. Por la Definición 3.1.9, $i \in O^{\mu}(t)$ si y sólo si $i \in \mu(root(t)) = \mu(f)$. Como $sprefix_t(u) = sprefix_s(u)$, tenemos que $root(t) = root(s) = f$, luego $s = f(\tilde{s})$ y $sprefix_{t_i}(u') = sprefix_{s_i}(u')$. Por la hipótesis de inducción $u' \in O^{\mu}(t_i) \Leftrightarrow u' \in O^{\mu}(s_i)$. Por tanto, por la Proposición 3.2.1, $u \in O^{\mu}(t) \Leftrightarrow i \in O^{\mu}(t) \wedge u' \in O^{\mu}(t_i) \Leftrightarrow i \in O^{\mu}(s) \wedge u' \in O^{\mu}(s_i) \Leftrightarrow u \in O^{\mu}(s)$. □

La Proposición 3.2.3 establece que las restricciones de reemplazamiento sensibles al contexto *sólo* dependen de los símbolos que etiquetan los nodos del término que se encuentran inmediatamente *por encima* de la ocurrencia considerada. Son, por tanto, independientes tanto de los símbolos que se encuentran por debajo de la ocurrencia considerada como alrededor de ésta. Empleando este resultado no resulta difícil definir un algoritmo que caracterice una ocurrencia de un término como reemplazante o no reemplazante sin necesidad de calcular exhaustivamente el conjunto de ocurrencias reemplazantes de un término.

La Proposición 3.2.1 expresa una propiedad muy básica del conjunto de ocurrencias reemplazantes que vamos a emplear para demostrar algunos resultados de interés.

La primera consecuencia es que el conjunto de ocurrencias reemplazantes $O^{\mu}(t)$ de un término t es un *cono* respecto al orden $(O(t), \leq)$.

Corolario 3.2.4 *Sea Σ una signatura, $\mu \in M_{\Sigma}$ y $t \in \mathcal{T}(\Sigma, V)$. Sean $u, v \in O(t)$. Si $u \leq v$ y $v \in O^{\mu}(t)$, entonces $u \in O^{\mu}(t)$. De forma equivalente, si $u \leq v$ y $u \notin O^{\mu}(t)$, entonces $v \notin O^{\mu}(t)$.*

DEMOSTRACIÓN. Si $u \leq v$, podemos escribir $v = u.v'$ para alguna $v' \in Occ$. Dado que $v \in O^\mu(t)$, la Proposición 3.2.1 asegura que $u \in O^\mu(t)$. Una vez probado esto, es sencillo ver que si no se cumpliera la sentencia que se afirma equivalente, incurriríamos en contradicción con la primera de ellas. □

Como consecuencia de este último resultado, introducimos el concepto de *contexto reemplazante máximo* de un término respecto a una función de reemplazamiento: se trata del contexto que encierra todos los símbolos reemplazantes del término (y sólo éstos).

Definición 3.2.5 (Contexto reemplazante máximo de un término) *Sea Σ una signatura, $t \in \mathcal{T}(\Sigma, V)$ y $\mu \in M_\Sigma$. El* contexto μ-reemplazante máximo *(o CMR para abreviar) de t es el contexto $CMR^\mu(t)$ donde:*

$$CMR^\mu(t) = \begin{cases} x & si\ t = x \\ f(C_1[\,], \ldots, C_k[\,]) & si\ t = f(t_1, \ldots, t_k),\ k = ar(f)\ y \\ & \quad C_i[\,] = \begin{cases} CMR^\mu(t_i) & si\ i \in \mu(f) \\ \Box & si\ i \notin \mu(f) \end{cases} \\ & para\ 1 \leq i \leq k \end{cases}$$

Adviértase que, siendo $C[\,] = CMR^\mu(t)$, tenemos que $t = C[t_1, \ldots, t_n]$ es tal que $\{u_1, \ldots, u_n\} = minimal(\overline{O^\mu}(t))$, y $t_i = t|_{u_i}$, $1 \leq i \leq n$. Que $O^\mu(t)$ sea un cono del conjunto ordenado $(O(t), \leq)$ implica que, salvo las constantes $\Box$ que representan los huecos del contexto, cualquier otro subtérmino de $CMR^\mu(t)$ es μ-reemplazante. Esto justifica la denominación de "contexto reemplazante máximo".

Proposición 3.2.6 *Sea Σ una signatura, $t \in \mathcal{T}(\Sigma, V)$, y $\mu \in M_\Sigma$. Entonces, $\overline{O^\mu}(CMR^\mu(t)) = O_\Box(CMR^\mu(t))$. Si $t \notin V$, entonces $O^\mu(CMR^\mu(t)) = O^\mu(t) = O_\Sigma(CMR^\mu(t))$.*

También es interesante advertir que, dada una función de reemplazamiento μ cualquiera, el contexto reemplazante máximo de un término t nunca está vacío (es decir, $CMR^\mu(t) \neq \Box$).

Ejemplo 3.2.7 *Consideremos el término $t =$* `if(and(true,false),0+s(0),0)` *del Ejemplo 3.1.10 y la función de reemplazamiento μ dada en dicho ejemplo. El contexto reemplazante máximo de t es $CMR^\mu(t) =$* `if(and(true,□),□,□)`.

Una consecuencia inmediata del Corolario 3.2.4 es la siguiente: no existen ocurrencias reemplazantes por debajo de ocurrencias no reemplazantes.

Proposición 3.2.8 *Sea $t \in \mathcal{T}(\Sigma, V)$ y $\mu \in M_\Sigma$. Si $u \in O^\mu(t)$ y $v \in \overline{O^\mu}(t)$, entonces $v \not\leq u$.*

3.2.1 Las restricciones de reemplazamiento durante el proceso de reducción

Las propiedades fundamentales presentadas más arriba son importantes al abordar la implementación de sistemas de reducción basados en la *rsc*. Los mecanismos de *instanciación* y *reemplazamiento* de (sub)términos son esenciales en las operaciones de reescritura y ambos introducen cambios locales en la estructura de los términos que participan en el proceso de reducción. Es necesario estudiar cómo evolucionan las restricciones de reemplazamiento que conciernen a dichos términos a medida que avanza el proceso de reescritura. La Proposición 3.2.3 permite realizar el cálculo del conjunto de ocurrencias reemplazantes de forma *incremental* respecto a dicho proceso de reducción. La incrementalidad del cálculo respecto a las operaciones de instanciación y reemplazamiento también es consecuencia de dicha proposición: puesto que tanto la instanciación de las variables de un término como el reemplazamiento de un subtérmino por otro dejan inalterados los símbolos que se encuentran por encima de las variables instanciadas o del subtérmino reemplazado, resulta sencillo emplear la Proposición 3.2.3 para demostrar los siguientes resultados.

Proposición 3.2.9 (Persistencia respecto a la instanciación) *Sea Σ una signatura, $\mu \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Si $u \in O(t)$ y $\sigma \in Subst(\Sigma, V)$, entonces $u \in O^\mu(t) \Leftrightarrow u \in O^\mu(\sigma(t))$.*

DEMOSTRACIÓN. Claramente, $sprefix_t(u) = sprefix_{\sigma(t)}(u)$ para todo término t y toda ocurrencia $u \in O(t)$. Por tanto, puesto que $O(t) \subseteq O(\sigma(t))$, se tiene que $u \in O(t) \cap O(\sigma(t))$ y la Proposición 3.2.3 asegura el resultado deseado. □

Proposición 3.2.10 (Persistencia respecto al reemplazamiento) *Sea Σ una signatura, $\mu \in M_\Sigma$, $t, s \in \mathcal{T}(\Sigma, V)$ y $u, v \in O(t)$. Si $u \geq v$ o bien $u \parallel v$, entonces $u \in O^\mu(t) \Leftrightarrow u \in O^\mu(t[s]_v)$.*

DEMOSTRACIÓN. Similar a la Proposición 3.2.9. □

3.3 La relación de reescritura sensible al contexto

Una vez establecido cómo asociar un conjunto de ocurrencias reemplazantes $O^\mu(t)$ a cada término t, la idea de restringir el proceso de reescritura a las ocurrencias de redexes *reemplazantes* surge de forma natural.

Definición 3.3.1 (Reescritura sensible al contexto) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Un término t es μ-reescrito al término s, lo cual se denota como $t \overset{u}{\hookrightarrow}_{\mathcal{R}(\mu)} s$, si $t \overset{u}{\rightarrow}_{\mathcal{R}} s$ y $u \in O^\mu(t)$.*

La *relación de reescritura sensible al contexto en un paso*, inducida por el SRTE $\mathcal{R}$ y la función de reemplazamiento μ es $\hookrightarrow_{\mathcal{R}(\mu)}$. La *relación de reescritura sensible al contexto* (*rsc*) es $\hookrightarrow^*_{\mathcal{R}(\mu)}$, el cierre reflexivo y transitivo de la relación de reescritura en un paso. Normalmente eliminaremos las referencias al SRTE o a la función de reemplazamiento (o a ambos) involucrados en la definición de $\hookrightarrow_{\mathcal{R}(\mu)}$, escribiendo $\hookrightarrow_\mu$ o simplemente $\hookrightarrow$, cuando ello no induzca a confusión. Una secuencia de μ-reescritura $t_1 \hookrightarrow t_2 \hookrightarrow \cdots \hookrightarrow t_n$ se denomina también una μ-derivación.

Observación 3.3.2 *Observemos que, de acuerdo con la definición declarativa de relación de reescritura dada en [DJ90, Jou94], una relación de reescritura es la mínima relación de reducción que contiene todas las instancias de las reglas del SRT y que es cerrada respecto a las propiedades de reflexividad, transitividad y aplicación sobre contextos arbitrarios. La propiedad que se relaja en la* rsc *es esta última, aunque, como veremos, se verifica una versión más débil, pero todavía útil, de la misma (Proposición 3.3.6).*

A continuación presentamos un primer ejemplo de reescritura sensible al contexto.

Ejemplo 3.3.3 *Consideremos el siguiente SRT $\mathcal{R}$:*

```
if(true,x,y)  → x        and(true,x)  → x
if(false,x,y) → y        and(false,y) → false
0+x           → x
s(x)+y        → s(x+y)
```

y la función de reemplazamiento μ definida como sigue[1]*:*

$$\mu(\mathtt{s}) = \emptyset \quad y \quad \mu(\mathtt{if}) = \mu(\mathtt{and}) = \mu(\mathtt{+}) = \{1\}$$

Consideremos la expresión inicial $t =$ `if(and(true,false),0+s(0),0)`*. Entonces, dado que $1 \in O^\mu(t)$, podemos realizar el siguiente paso de μ-reducción (el redex seleccionado aparece subrayado):*

`if(`<u>`and(true,false)`</u>`,0+s(0),0)` $\hookrightarrow$ `if(false,0+s(0),0)`

No obstante,

`if(and(true,false),`<u>`0+s(0)`</u>`,0)` $\not\hookrightarrow$ `if(and(true,false),s(0),0)`

puesto que 2 no es una ocurrencia reemplazante de t, es decir, $2 \notin O^\mu(t)$.

Ejemplo 3.3.4 *Ahora podemos formular el Ejemplo 1.1.2 empleando la notación y terminología introducidas. La primera secuencia de reducción presentada correspondía a la μ-derivación*

[1] Recordemos que, tal como acordamos en la Sección 3.1, no daremos explícitamente el valor de la función de reemplazamiento aplicada sobre símbolos constantes puesto que siempre se devuelve el conjunto vacío.

$$\texttt{first(}\underline{\texttt{zero}}\texttt{,from(0))} \hookrightarrow \underline{\texttt{first(0,from(0))}} \hookrightarrow \texttt{[]}$$

mientras que en la segunda secuencia tenemos

$$\texttt{first(zero,}\underline{\texttt{from(0)}}\texttt{)} \hookrightarrow \texttt{first(zero,0::}\underline{\texttt{from(s(0))}}\texttt{)} \not\hookrightarrow$$

lo cual evita que la μ-derivación desarrolle una secuencia de reescritura infinita.

Observación 3.3.5 *Dado un SRTE $\mathcal{R} = (\Sigma, R)$, al considerar la función de reemplazamiento $\mu_\top$, que es el máximo del retículo M_Σ, la relación de reescritura sensible al contexto $\hookrightarrow_{\mathcal{R}(\mu_\top)}$ coincide con la relación de reescritura estándar, es decir: $\hookrightarrow_{\mathcal{R}(\mu_\top)} = \rightarrow_{\mathcal{R}}$.*

Escribimos $O^\mu_{\mathcal{R}}(t)$ para referirnos al conjunto de ocurrencias de redexes *reemplazantes* (o μ-redexes) existentes en el término t: $O^\mu_{\mathcal{R}}(t) = O_{\mathcal{R}}(t) \cap O^\mu(t)$. De forma análoga al caso no restringido, cuando $O^\mu_{\mathcal{R}}(t) = \emptyset$, decimos que t es una *forma μ-normal*. En general, a menos que se establezca de forma diferente, extenderemos las denominaciones de las propiedades usuales de la reescritura estándar al caso de la *rsc* en la forma natural. En particular, diremos que un SRTE $\mathcal{R}$ es μ-terminante, μ-confluente, etc., para indicar que la relación de reescritura $\hookrightarrow_{\mathcal{R}(\mu)}$ es terminante, confluente, etc.

A continuación analizamos las propiedades básicas de la relación de reescritura sensible al contexto.

Proposición 3.3.6 (Cierre bajo contextos reemplazantes) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\mu \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Sea $C[\,]_u$ un contexto y $u \in O^\mu(C[t]_u)$. Si $t \hookrightarrow^* s$ entonces $C[t]_u \hookrightarrow^* C[s]_u$.*

DEMOSTRACIÓN. Por inducción sobre la longitud n de la μ-derivacion. Si $n = 0$ el resultado es inmediato. Si $n > 0$, sea $t \hookrightarrow t' \hookrightarrow^* s$. Si $t \hookrightarrow t'$, existe una occurrencia $v \in O^\mu(t)$ y una regla $l \rightarrow r \in R$ tal que $t|_v = \sigma(l)$ para alguna sustitución σ. Dado que $u \in O^\mu(C[t]_u)$, por la Proposicion 3.2.1, $u.v \in O^\mu(C[t]_u)$. Así pues, la misma regla $l \rightarrow r$ puede aplicarse a la ocurrencia $u.v$ de $C[t]_u$ y por tanto $C[t]_u \hookrightarrow C[t']_u$. Por la Proposición 3.2.10, $u \in O^\mu(C[t']_u)$. Teniendo en cuenta la hipótesis de inducción, concluimos el resultado deseado. □

La siguiente proposición establece la relación existente entre relaciones de reescritura sensible al contexto inducidas por funciones de reemplazamiento comparables.

Proposición 3.3.7 (Monotonía de $\hookrightarrow$ respecto al orden $\sqsubseteq$) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu, \mu' \in M_\Sigma$. Si $\mu \sqsubseteq \mu'$, entonces $\hookrightarrow_{\mathcal{R}(\mu)} \subseteq \hookrightarrow_{\mathcal{R}(\mu')}$.*

DEMOSTRACIÓN. Si $t \hookrightarrow_{\mathcal{R}(\mu)} s$, entonces existe una occurrencia $u \in O^\mu(t)$ y una regla $l \rightarrow r \in R$ tal que $t|_u = \sigma(l)$ para alguna sustitución σ. Puesto que $u \in O^\mu(t)$, por la Proposicion 3.1.12 se verifica $u \in O^{\mu'}(t)$, por lo que $t \hookrightarrow_{\mathcal{R}(\mu')} s$. □

Desde un punto de vista computacional, esta proposición formaliza el hecho intuitivamente claro de que la *rsc* es una restricción de la reescritura estándar (recordemos la Observación 3.3.5) y tiene asociado un espacio de reducción menor que la reescritura estándar.

También podemos probar que la *rsc* es estable respecto a la aplicación de sustituciones sobre los términos involucrados en la secuencia de reducción.

Proposición 3.3.8 (Estabilidad de la *rsc*) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\mu \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Si $\sigma \in Subst(\Sigma, V)$ y $t \hookrightarrow^* s$, entonces $\sigma(t) \hookrightarrow^* \sigma(s)$.*

DEMOSTRACIÓN. Por inducción sobre la longitud n de la μ-derivación. Si $n = 0$, es inmediato. Si $n > 0$ escribimos $t \hookrightarrow t' \hookrightarrow^* s$. Si $t \hookrightarrow t'$, existe una occurrencia $u \in O^\mu(t)$ y una regla $l \to r \in R$ tal que $t|_u = \sigma'(l)$ y $t' = t[\sigma'(r)]_u$ para alguna sustitución σ'. Puesto que $l \notin V$, por la Proposición 2.2.3, tenemos $t|_u \notin V$ y $\sigma(t)|_u = \sigma(\sigma'(l)) = \sigma''(l)$. Por la Proposición 3.2.9, $u \in O^\mu(\sigma(t))$. Entonces, $\sigma(t) \hookrightarrow \sigma(t)[\sigma(\sigma'(r))]_u$, y, por la Proposición 2.2.4, $\sigma(t) \hookrightarrow \sigma(t[\sigma'(r)]_u)$, es decir, $\sigma(t) \hookrightarrow \sigma(t')$. Por la hipótesis de inducción concluimos el resultado deseado. □

La extensión a la *rsc* de la noción de multiderivación es también inmediata.

Definición 3.3.9 (μ-multiderivación elemental) *Denotamos mediante $A^\mu : t \overset{U}{\hookrightarrow} s$ una μ-multiderivación elemental que contrae los redexes a las occurrencias (disjuntas) $U \subseteq O^\mu_{\mathcal{R}}(t)$, es decir, $\forall u, u' \in U.\ u \neq u' \Rightarrow u \parallel u'$.*

Escribimos $t \Vdash s$ cuando la información acerca de las ocurrencias concretas que se han contraído no sea relevante.

Lema 3.3.10 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Entonces, $\hookrightarrow^+ = \Vdash^+$.*

DEMOSTRACIÓN. Que $\hookrightarrow^+ \subseteq \Vdash^+$ es inmediato, puesto que $\hookrightarrow \subseteq \Vdash$. Para demostrar que $\Vdash^+ \subseteq \hookrightarrow^+$, procedemos por inducción sobre la longitud n de la multiderivación $t \Vdash^+ s = t \overset{U}{\hookrightarrow} t'' \Vdash^* s$.

1. Si $n = 1$, entonces $t'' = s$ y $t \Vdash^+ s = t \overset{U}{\hookrightarrow} t''$. Para demostrar lo que deseamos en este caso particular, procedemos por inducción sobre el número de ocurrencias contraídas, $|U| \geq 1$.

   (a) Si $U = \{u\}$, entonces $t \overset{\{u\}}{\hookrightarrow} s \Leftrightarrow t \overset{u}{\hookrightarrow} s$.

   (b) Si $U = \{u\} \uplus U'$ con $U' \neq \emptyset$, entonces, puesto que, para todo $u, u' \in U$, tenemos que $u \neq u' \Rightarrow u \parallel u'$, podemos escribir $t \overset{U}{\hookrightarrow} s \Leftrightarrow t \overset{\{u\}}{\hookrightarrow} t' \overset{U'}{\hookrightarrow} s$. Demostremos que esto es correcto: $t' = t[\sigma(r)]_u$ para alguna regla $l \to r$. Puesto que para toda ocurrencia $u' \in U'$ se verifica $u \parallel u'$ y $u' \in O^\mu(t)$, por

la Proposición 3.2.10, $U' \subseteq O^{\mu}(t')$. Las reducciones empleando ocurrencias en U' no interfieren la reducción que emplea u. Por este motivo, ninguna ocurrencia de un redex en U' puede ser destruida por la contracción de un redex situado sobre u. Por esta razón, $U' \subseteq O^{\mu}_{\mathcal{R}}(t')$ y $t' \overset{U'}{\hookrightarrow} s$. Por la hipótesis de inducción, $t' \hookrightarrow^{+} s$, es decir, $t \hookrightarrow^{+} s$.

2. Si $n > 1$, tenemos que $t \overset{U}{\hookrightarrow} t'' \Vdash^{+} s$. Hemos demostrado en el párrafo anterior que $t \hookrightarrow^{+} t''$. Por lo tanto, por la hipótesis de inducción, $t'' \hookrightarrow^{+} s$ y $t \hookrightarrow^{+} s$.

Esto completa la demostración. □

3.4 Algunos ejemplos de aplicación

Con el fin de ilustrar de forma práctica el tipo de mejoras computacionales que cabe esperar del empleo de la *rsc* presentamos los siguientes ejemplos.

Optimización de la evaluación de expresiones

Tal como se puso de manifiesto en el Ejemplo 3.3.3, en algunos casos es razonable evaluar algún argumento fijado de antemano hasta obtener su forma normal (o su valor) en detrimento de la evaluación de otros argumentos de una llamada a función. El motivo es que la evaluación (más o menos completa) de dicho argumento es necesaria para determinar la regla que debe aplicarse para contraer un redex situado sobre una ocurrencia más externa.

Ejemplo 3.4.1 *Consideremos las reglas que definen los operadores booleanos and/or tal como vienen implementados en el lenguaje Lisp:*

```
and(true,x)  → x        or(true,x)  → true
and(false,x) → false    or(false,x) → x
```

Si el primer argumento t de una llamada a función `and`(t,s) *(*`or`(t,s)*) se evalúa a* `false` *(*`true`*), de acuerdo con estas reglas cualquier reducción realizada sobre el segundo argumento de la llamada a función es inútil y no contribuye a obtener el valor final asociado a la misma.*

Si utilizamos una función de reemplazamiento μ haciendo que

$$\mu(\texttt{and}) = \mu(\texttt{or}) = \{1\}$$

la evaluación de redexes situados sobre ocurrencias del segundo argumento puede ser evitada (o retrasada).

Otras situaciones interesantes surgen al estudiar la implementación de funciones sobre listas.

Ejemplo 3.4.2 *Las reglas siguientes definen los operadores estándar (*proyecciones*)* `head` *y* `tail` *para la selección de componentes de una lista:*

```
head(x::y) → x
tail(x::y) → y
```

La evaluación de una llamada a la función `head` *sólo requiere reducir la cabeza de la lista, mientras que una llamada a la función* `tail` *evaluaría el resto de la lista. Estas observaciones sólo serán relevantes si la implementación del* constructor de listas *(*`::`*) no evalúa sus argumentos de forma sistemática. En efecto, se han desarrollado lenguajes de programación donde el constructor de listas no evalúa todos sus argumentos [FW76, HM76, HW87, Rea93].*

Este comportamiento se obtiene fácilmente empleando la rsc *sin más que imponer* $\mu(\texttt{::}) = \emptyset$.

Manipulación de estructuras de datos infinitas

El operador perezoso de construcción de listas discutido en el Ejemplo 3.4.2 es adecuado para implementar cómputos con estructuras de datos (listas en particular) infinitas [FW76, Rea93].

Ejemplo 3.4.3 *Consideremos el siguiente programa, que puede utilizarse para definir expresiones capaces de seleccionar un elemento de una lista infinita.*

```
sel(0,x::y)    → x          from(x) → x::from(s(x))
sel(s(x),y::z) → sel(x,z)
```

Definimos $\mu(\texttt{::}) = \{1\}$ *y* $\mu(f) = \mathbb{N}^{+}_{ar(f)}$ *para cualquier otro símbolo* f. *Consideremos la siguiente* μ*-derivación:*

```
sel(s(s(0)),from(0))
   ↪ sel(s(s(0)),0::from(s(0)))
   ↪ sel(s(0),from(s(0)))
   ↪ sel(s(0),s(0)::from(s(s(0))))
   ↪ sel(0,from(s(s(0))))
   ↪ sel(0,s(s(0))::from(s(s(s(0)))))
   ↪ s(s(0))
```

Como puede observarse, a pesar de la presencia de una expresión `from(0)` *que denota una estructura de datos infinita (la lista de todos los números naturales), es posible operar sobre ella (mediante la función* `sel`*) y seleccionar uno de los elementos de dicha lista infinita.*

Podemos considerar otro ejemplo interesante de aplicación de la *rsc* en cómputos que involucran estructuras de datos infinitas.

Ejemplo 3.4.4 *Sea el SRT $\mathcal{R}$ que permite computar y manipular números primos.*

```
from(x)        → x::from(s(x))
primes         → sieve(from(s(s(0))))
sieve(x::y)    → x::sieve(filter(x,y))
filter(x,y::z) → if(x|y,filter(x,z),y::filter(x,z))
sel(0,x::y)    → x            if(true,x,y)  → x
sel(s(x),y::z) → sel(x,z)     if(false,x,y) → y
```

Supongamos que `|` *es un símbolo predefinido y que* `x|y` *significa* `x` divide a `y` *(evaluándose, en ese caso a* `true`*). Fijamos* $\mu(\mathtt{s}) = \mu(\mathtt{::}) = \mu(\mathtt{from}) = \emptyset$, $\mu(\mathtt{sieve}) = \mu(\mathtt{if}) = \{1\}$, $\mu(\mathtt{filter}) = \{2\}$, $\mu(\mathtt{sel}) = \{1,2\}$. *Más adelante veremos que este SRT pertenece a una clase de SRTs que poseen propiedades interesantes que permiten definir estrategias sensibles al contexto que siempre obtienen formas μ-normales con una implementación muy simple. Esto permite realizar cálculos que involucren números primos.*

En muchos casos, la capacidad de la *rsc* para realizar cómputos con estructuras de datos infinitas procede de la posibilidad de obtener un comportamiento terminante para la relación de reescritura sensible al contexto. La *rsc* mejora el comportamiento de terminación de forma análoga al efecto obtenido por las estrategias perezosas. De hecho, existen métodos formales para establecer la terminación de $\hookrightarrow$ (ver el Capítulo 5).

Cómputos seguros

La noción de *computación segura* [Lal93, Vui74] es la contrapartida operacional de la noción semántica de *strictness* [Myc80, PE93]. Al interpretar (como es habitual) el elemento mínimo $\bot$ de un dominio de Scott[2] D, como la representación de los cómputos *no terminantes*, la información de *strictness* puede emplearse para definir una estrategia de reducción: si evaluamos un argumento estricto (empleando una estrategia normalizante), el hecho de que dicha evaluación no termine *no impide que la evaluación de la llamada a función principal produzca un valor* (porque es seguro que dicha llamada a función no tiene un valor definido) [Lal93, PE93].

Una estrategia de reducción cuyos pasos de reducción siempre se realizan sobre contextos seguros se denomina una estrategia *segura*. Un contexto $C[\,]_u$ es seguro, si $C[\bot]_u = \bot$. Si definimos la función de reemplazamiento μ de modo que $\mu(f)$ sea el conjunto de índices de los argumentos *estrictos* de f, para cualquier símbolo f de la signatura, es evidente que cualquier contexto reemplazante $C[\,]_u$, con $u \in O^{\mu}(C)$, es un contexto seguro. Por tanto, cualquier secuencia de reducción sensible

[2] Cualquier dominio de Scott tiene un elemento mínimo que denotamos como $\bot$, ver [Sco70, Sco82, SLG94].

al contexto resulta ser una derivación segura. La información sobre la *strictness* de las funciones puede obtenerse empleando alguna clase de análisis de *strictness* [CP85, Myc80, Wad87].

Sin embargo, en el marco computacional de la *rsc*, la especificación de las restricciones de reemplazamiento a partir de información de *strictness no* garantiza la completitud de los cómputos (ver las Secciones 4.2 y 11.1).

Mecanización de sistemas de inferencia

De forma análoga a la reescritura (ver [Lal93, Pla94]), la relación de reescritura sensible al contexto puede definirse como el cierre deductivo de un sistema de inferencia [Smu71] cuyos teoremas son de la forma $t \hookrightarrow^* s$, donde t y s son términos, y que consta de un esquema de axioma

$$\frac{}{l \hookrightarrow r}$$

para cada regla $l \to r$ (que se concreta en tantos axiomas como instancias puedan realizarse) y de tantas reglas de propagación contextual (*context-passing rules* [Lal93]) para cada símbolo f de la signatura como índices reemplazantes imponga la función de reemplazamiento. Es decir, para cada posición argumental *reemplazante* $i \in \mu(f)$, de cada $f \in \Sigma$, definimos el esquema de regla:

$$\frac{t \hookrightarrow^* s}{f(t_1, \ldots, t_{i-1}, t, \ldots, t_k) \hookrightarrow^* f(t_1, \ldots, t_{i-1}, s, \ldots, t_k)}$$

(que de nuevo se concreta en tantas reglas de inferencia como instancias del esquema podamos realizar) además de las reglas usuales que expresan la reflexividad y transitividad de la relación.

Muchos sistemas computacionales han sido formalizados mediante sistemas de inferencia y, en algunos casos, las reglas de inferencia respetan el formato expresado por las reglas anteriores. Este es el caso de la β-reducción débil, las estrategias call-by-name del λ-cálculo [Lal93], el λ-cálculo etiquetado paralelo [Oli96, LO96] y también el π-cálculo [Mil91]. Por tanto, es posible emplear la *rsc* para implementar dichos sistemas computacionales. Al tratar estos sistemas en el marco formal de la reescritura podremos explotar la variedad de máquinas de reducción existentes (máquinas de reescritura [She94], de reducción de grafos [GKM87], etc.) para obtener implementaciones eficientes de dichos sistemas. Una ventaja adicional es que todos los resultados acerca de propiedades computacionales de la *rsc* pueden utilizarse para analizar el comportamiento en ejecución de dichos sistemas.

3.5 Trabajos relacionados

En los trabajos de Maranget [Mar90] y Kamperman y Walters [KW95], la especificación de restricciones de reemplazamiento también tiene lugar a dos niveles: en un primer lugar se asocian restricciones de reemplazamiento a las posiciones argumentales de los símbolos de la signatura. A continuación, se utilizan éstas para establecer una restricción sobre los reemplazamientos que pueden realizarse en el seno de los términos construidos con estos símbolos. Presentamos con cierto detalle estas dos restricciones de la reescritura que serán comparadas con la *rsc* a lo largo de la tesis. También se discute la relación existente entre la *rsc* y la especificación de anotaciones sintácticas en lenguajes funcionales y algebraicos.

Reducción condicional

En [Mar90], Maranget analiza el uso de la llamada *reducción condicional* (*conditional reduction*, aunque esta denominación no tiene que ver con la relación de reducción asociada a los SRTs *condicionales* de [BK86, DOS88, Kap84]) para implementar reducciones perezosas óptimas.

Para definir la reducción condicional, Maranget introduce restricciones sintácticas en los reemplazamientos mediante una *función de dominio* que en esencia coincide con la función de reemplazamiento. Maranget denomina SRTs *condicionales* a los SRTs provistos de una función de dominio. La *reducción condicional* de Maranget coincide con la reescritura sensible al contexto *paralela*. Sin embargo, debido a su interés en asegurar un comportamiento verdaderamente perezoso, donde cada redex sólo se contraiga una vez, necesita imponer algunas condiciones adicionales sobre los SRTs *condicionales* aceptables. De hecho, Maranget considera SRTs μ-ortogonales (ver el Capítulo 6) cuya función de dominio (que podemos denotar como μ) verifica la siguiente restricción adicional: $V_a(r) \subseteq V_a(l)$ y $V_f(r) \subseteq V_f(l)$ para toda regla $l \to r$ del SRT. Aquí, $V_a(t)$, el conjunto de variables *permitidas* (en la terminología de Maranget) puede definirse como $V_a(t) = Var^{\mu}(t)$, donde la función de reemplazamiento μ es la función de dominio. El conjunto de variables *prohibidas* $V_f(t)$ viene dado por $V_f(t) = \{x \in Var(t) \mid \exists u \in \overline{O^{\mu}}(t).\ t|_u = x\}$. Estas condiciones adicionales son necesarias porque Maranget estudia la aplicación de técnicas de reducción de grafos para implementar el comportamiento perezoso en las evaluaciones.

Esto introduce una limitación de principio en el alcance de los resultados obtenidos por Maranget respecto a los que presentamos aquí. De hecho, estas limitaciones originan cierta confusión en el análisis y la comprensión de los factores que influyen en el comportamiento de la relación restringida introducida, en particular, en el estudio de las condiciones que aseguran la confluencia de la reducción condicional (ver Capítulo 6).

Reescritura perezosa

Las restricciones en los reemplazamientos definidas por Kamperman y Walters [KW95] emplean un predicado Λ sobre símbolos de función y posiciones argumentales que guarda una relación simple con nuestra función de reemplazamiento[3]:

$$\neg\Lambda(f,i) \Leftrightarrow i \in \mu(f)$$

para todo $f \in \Sigma$ y $1 \leq i \leq ar(f)$. Kamperman y Walters definen los *caminos voraces* (*eager paths*), que coinciden con las ocurrencias reemplazantes. Su noción de *camino perezoso* (*lazy path*) coincide con nuestras ocurrencias no reemplazantes. Sin embargo, la reducción perezosa y la *rsc* no coinciden. De hecho, la *rsc* es un subconjunto propio de la *reescritura perezosa* puesto que Kamperman y Walters permiten realizar pasos de reescritura sobre ocurrencias voraces pero también sobre ocurrencias no voraces bajo ciertas condiciones (que por otra parte dependen del SRT concreto que se esté considerando).

Anotaciones de *strictness*

Los lenguajes de programación funcional perezosos como Clean [ENPS92, NSEP92, PE93], Gofer [Jon92] y Haskell [Dav92, PH97] admiten la especificación de anotaciones sintácticas en el programa, cuyo fin es guiar la introducción de modificaciones en la estrategia de evaluación perezosa.

Estas anotaciones pueden introducirse en el nivel sintáctico elemental, asociadas a los argumentos de los símbolos de función de la signatura o bien pueden quedar ligadas a algunos *subtérminos* de las ecuaciones de definición de las funciones, bien en la parte izquierda o en la derecha de las mismas (en este segundo caso la noción de *strictness* subyacente se refiere a contextos y huecos dentro de ellos más que a funciones y posiciones argumentales [Ben92, Bur87, Bur91]). Siguiendo la terminología empleada en [PE93], las primeras se denominan anotaciones *globales* y las segundas anotaciones *locales*. Las anotaciones globales se indican en la declaración del tipo de la función que se desea anotar (por ejemplo, situando un símbolo especial, tal como '`!`', junto al tipo del argumento que se desea anotar en el perfil de la función).

Ejemplo 3.5.1 *La siguiente declaración*

```
if :: !Bool a a -> a
if True  x y = x
if False x y = y
```

constituye una definición anotada de la función `if` *que indica la obligación de evaluar el primer argumento de cada llamada a dicha función.*

[3] En la definición del predicado $\Lambda(f,i)$, los autores permiten que i oscile entre 0 y $ar(f)$ aunque esta posibilidad nunca es utilizada en el artículo.

Las anotaciones globales exhiben una clara correspondencia con las restricciones de reemplazamiento expresadas mediante una función de reemplazamiento μ. En particular, en el ejemplo 3.5.1 escribiríamos $\mu(\texttt{if}) = \{1\}$.

Finalmente, tal como se comentó en la introducción, la semántica de las restricciones o anotaciones introducidas por la *rsc* difiere sustancialmente de la semántica usual de las anotaciones en programación funcional donde un argumento anotado *debe* ser reducido (en alguna medida) y un argumento no anotado *puede* ser reducido o no.

Estrategias locales

Algunos lenguajes de programación como OBJ2 [FGJM85], OBJ3 [GWMFJ93] o CafeOBJ [FN97], admiten la especificación de "estrategias locales" asociadas a los símbolos de función. Dichas estrategias locales se especifican como listas de números naturales y sirven para guiar la estrategia de evaluación.

Ejemplo 3.5.2 *La siguiente especificación [OF97]:*

```
sorts: Zero NzNat Nat List.
order: Zero<Nat
       NzNat<Nat
ops:   0:-> Zero
       s: Nat -> NzNat
       cons: Nat List -> List { strat: (1 0) }
       from: Nat -> List
       sel: Nat List -> Nat
vars:  X Y: Nat
       L: List
rules: from(X) -> cons(X,from(s(X))
       sel(0,cons(X,L)) -> X
       sel(s(X),cons(Y,L)) -> sel(X,L)
```

constituye una versión OBJ3 del programa del Ejemplo 3.4.3. Podemos observar la presencia de anotaciones locales explícitas para el símbolo `cons`*. La ausencia de anotaciones para los demás símbolos (*`0`*,* `s`*,* `from` *y* `sel`*) supone que cada uno de ellos adopta la estrategia local* por defecto*. Dado un símbolo k-ario f, su estrategia local por defecto es* $(1, 2, \ldots, k, 0)$*.*

Al considerar una llamada a función $f(t_1, \ldots, t_k)$, las estrategias locales asignadas a cada símbolo condicionan la evaluación como sigue: sólo se evalúan los argumentos cuyos índices, entre 1 y k, figuran en la lista asociada al símbolo f. Además, esto se hace de acuerdo con el orden especificado en la lista de argumentos. Si al recorrer la estrategia local se encuentra el índice 0 se intenta ejecutar la llamada a función

sobre los argumentos evaluados hasta el momento. La estrategia definida mediante estas pautas de comportamiento se denomina *estrategia E* (E-strategy, de *Evaluation strategy*).

En [OF97] se proporciona una implementación por reescritura de la estrategia E *regular* que asume una limitación adicional para las estrategias locales admisibles. Bajo la estrategia regular, toda estrategia local está vacía (es decir, la lista de índices de parámetros que la describe está vacía) o bien termina con cero y contiene como máximo una ocurrencia de cada entero positivo y no contiene ocurrencias sucesivas de cero [OF97]. La relación de reducción $\overset{ER}{\rightarrow}$ que implementa las secuencias de reescritura controladas por la estrategia E regular es un caso particular de la *rsc* si a partir de cada estrategia local $(i_1, \ldots, i_p)$, explícita o implícitamente asociada a un símbolo f, obtenemos $\mu(f)$ como sigue: $\mu(f) = \{i_1, \ldots, i_p\} \backslash \{0\}$. Esto se debe a que la ocurrencia $u \in O(t)$ del redex considerado por $\overset{ER}{\rightarrow}$ para efectuar el paso de reducción $t \overset{ER}{\rightarrow} s$ pertenece siempre al conjunto de ocurrencias μ-reemplazantes de t de acuerdo con la correspondencia establecida anteriormente entre estrategias locales y función de reemplazamiento (ver [OF97]). De hecho, las *formas E-normales* (*ENF*, *E-Normal Forms*) de [OF97] son siempre formas μ-normales para la función de reemplazamiento μ que expresa las estrategias locales asociadas a los símbolos de acuerdo con la correspondencia anterior.

Capítulo 4

Reescritura y reescritura sensible al contexto

Este capítulo está dedicado a estudiar si, al emplear la *rsc* en lugar de la reescritura, perdemos capacidad de cálculo y hasta qué punto. Para analizar este problema, resulta imprescindible caracterizar la relación existente entre las derivaciones de reescritura y las de reescritura sensible al contexto.

En la Sección 4.1 investigamos cómo influye la imposición de restricciones de reemplazamiento sensibles al contexto en el desarrollo normal del proceso de reescritura.

En la Sección 4.2 introducimos la función de reemplazamiento canónica asociada a un SRTE dado. Demostramos que, dada una derivación de reescritura que lleva a una forma estable, la función de reemplazamiento canónica es la mínima función de reemplazamiento que asegura la capacidad de la *rsc* para obtener un término cuyo símbolo a la raíz coincida con el de la forma estable obtenida.

En la Sección 4.3 demostramos que las formas μ-normales respecto a un SRTE y una función de reemplazamiento μ mayor o igual que la función de reemplazamiento canónica asociada a dicho SRTE son formas fuertemente estables y, por tanto, estables.

4.1 Compatibilidad de las restricciones de reemplazamiento con el proceso de reescritura

Debido a que la *rsc* es una restricción de la reescritura, toda derivación sensible al contexto corresponde, trivialmente, a una derivación de reescritura. La cuestión de interés es la contraria: ¿bajo qué condiciones es posible asociar una derivación sensible al contexto que sea 'equivalente' (en algún sentido) a una derivación de reescritura?

Para responder a esta cuestión, analizamos la repercusión de la imposición de restricciones de reemplazamiento en el mecanismo de reescritura y en la construcción de una derivación de reescritura. Debemos considerar tres aspectos principales: el *emparejamiento de patrones*, la *estrategia de reducción* y el *reemplazamiento de subtérminos* [O'D85]. El emparejamiento de patrones permite detectar la presencia de redexes en el término a reducir. La estrategia de reducción selecciona el redex que será reducido y la regla que será aplicada y, finalmente, tendrá lugar el reemplazamiento de ese redex por la correspondiente instancia de la parte derecha de la regla aplicada.

En la práctica, todos estos aspectos del mecanismo de reducción se interrelacionan dando lugar a un único algoritmo que combina las tres etapas. Por ejemplo, si consideramos una regla $f(\tilde{l}) \rightarrow r$ y un término $t = f(\tilde{t})$, podemos decir que, sean cuales sean los términos $t_1, \ldots, t_k$, donde $k = ar(f)$, ya tiene lugar un emparejamiento *parcial* entre $f(\tilde{l})$ y $f(\tilde{t})$, por el hecho de que ambos términos tienen el mismo símbolo f en cabeza. Es posible que algún subtérmino l_i, $1 \leq i \leq ar(f)$, también empareje con el correspondiente t_i de t, es decir, $t_i = \sigma(l_i)$ para alguna sustitución σ. Si deseamos 'extender' este emparejamiento parcial, deberemos reducir (en alguna medida) otros subtérminos t_j que no emparejan todavía. Por tanto, las restricciones de reemplazamiento *deberían ser compatibles con esta necesidad de extender el emparejamiento de un término con una regla con la cual empareja parcialmente.*

Ejemplo 4.1.1 *Consideremos el SRT del Ejemplo 3.3.3 y el término a reducir* $t =$ `if(`$cond$`,`s`,`s'`)`. *Observemos que existe un emparejamiento parcial entre* t *y las reglas que definen la función* `if`. *Si consideramos la forma de las reglas asociadas a la función* `if` *se advierte que, para extender el emparejamiento parcial, sólo necesitamos reducir la condición cond a* `true` *o* `false`. *Si escogemos* μ *de forma que* $1 \notin \mu(\texttt{if})$, *esta reducción queda prohibida y el emparejamiento parcial no puede extenderse en la dirección deseada.*

La estrategia de reducción debe seleccionar el subtérmino inmediato t_i sobre el cual es necesario actuar para intentar obtener el emparejamiento total. Si consideramos las restricciones de reemplazamiento introducidas por una función de reemplazamiento μ, podemos definir una propiedad que una *lhs* l de una regla $l \rightarrow r$ debe satisfacer para asegurar que un emparejamiento parcial de un término puede ser extendido (mediante μ-reducciones) siempre que sea necesario. Claramente, lo que necesitamos imponer es que las ocurrencias no reemplazantes de l correspondan a variables. En ese caso, no será necesario actuar sobre subtérminos que emparejen con esa parte de la *lhs* puesto que una variable empareja con cualquier término. Para ilustrar esto, volvamos al ejemplo 4.1.1. Podríamos continuar la evaluación del término t reduciendo *cond*, o bien s o bien s'. Sin embargo, dado que 2 y 3 son ocurrencias variables de las *lhs*'s `if(true,x,y)` e `if(false,x,y)` de las reglas para la función `if`, las reducciones sobre

s y s' no mejoran el emparejamiento. Sólo la reducción del subtérmino *cond* puede conseguirlo en su caso. Por tanto, tiene sentido imponer $\mu(\texttt{if}) = \{1\}$. Esto motiva la siguiente definición.

Definición 4.1.2 (Término μ-compatible) *Sea Σ una signatura y $\mu \in M_\Sigma$. Un término $t \in \mathcal{T}(\Sigma, V)$ es* μ-compatible *(escribiremos $\mathsf{comp}_\mu(t)$) si todas las ocurrencias no reemplazantes de t son variables: $\overline{O^\mu}(t) \subseteq O_V(t)$. De forma equivalente, t es μ-compatible si las ocurrencias no variables son reemplazantes: $O_\Sigma(t) \subseteq O^\mu(t)$.*

Esta propiedad se extiende a conjuntos de términos: un conjunto $T \subseteq \mathcal{T}(\Sigma, V)$ es μ-compatible (y escribiremos $\mathsf{comp}_\mu(T)$) si y sólo si $\forall t \in T.\ \mathsf{comp}_\mu(t)$. El predicado comp_μ es monótono respecto al orden $\sqsubseteq$ sobre M_Σ:

Proposición 4.1.3 *Sea Σ una signatura, $\mu, \mu' \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Si $\mu \sqsubseteq \mu'$, entonces $\mathsf{comp}_\mu(t) \Rightarrow \mathsf{comp}_{\mu'}(t)$.*

Demostración. Por definición, $\mathsf{comp}_\mu(t)$ si y sólo si $O_\Sigma(t) \subseteq O^\mu(t)$. Puesto que $\mu \sqsubseteq \mu'$, por la Proposición 3.1.12, $O^\mu(t) \subseteq O^{\mu'}(t)$. Por tanto, $O_\Sigma(t) \subseteq O^{\mu'}(t)$, es decir, $\mathsf{comp}_{\mu'}(t)$. □

La siguiente proposición establece que la μ-compatibilidad de un término $t \in \mathcal{T}(\Sigma, V)$ sólo depende de los símbolos $\Sigma(t)$ presentes en dicho término.

Proposición 4.1.4 *Sea Σ una signatura, $\mu \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Entonces, $\mathsf{comp}_\mu(t)$ si y sólo si $\mathsf{comp}_{\mu_{|\Sigma(t)}}(t)$.*

Demostración. Puesto que $O_\Sigma(t) = O_{\Sigma(t)}(t)$ y $O^\mu(t) = O^{\mu_{|\Sigma(t)}}(t)$, obtenemos la conclusión al considerar la Definición 4.1.2. □

La μ-compatibilidad es una propiedad estructural.

Proposición 4.1.5 *Sea Σ una signatura y $\mu \in M_\Sigma$. Sea $t \in \mathcal{T}(\Sigma, V)$ y $u \in O(t)$. Si $\mathsf{comp}_\mu(t)$, entonces $\mathsf{comp}_\mu(t[\,]_u)$ y $\mathsf{comp}_\mu(t|_u)$.*

Demostración. En primer lugar demostramos que $\mathsf{comp}_\mu(t)$ implica $\mathsf{comp}_\mu(t[\,]_u)$. Tenemos que $\mathsf{comp}_\mu(t) \Leftrightarrow O_\Sigma(t) \subseteq O^\mu(t)$. Puesto que $O_\Sigma(t[\,]_u) \subseteq O_\Sigma(t)$, dado que se verifica $\mathsf{comp}_\mu(t)$, se tiene que $u \in O_\Sigma(t[\,]_u)$ implica $u \in O^\mu(t)$. Por la Proposición 3.2.10, $u \in O^\mu(t[\,]_u)$ y se obtiene el resultado deseado[1].

Ahora demostramos que $\mathsf{comp}_\mu(t)$ implica $\mathsf{comp}_\mu(t|_u)$. Emplearemos el hecho de que $\mathsf{comp}_\mu(t) \Leftrightarrow \overline{O^\mu}(t) \subseteq O_V(t)$. Sea $v \in \overline{O^\mu}(t|_u)$. Por la Proposición 3.2.1, tenemos

[1] Observemos que, tal como se ha formulado el resultado, los huecos del contexto son considerados como *variables* (respecto a la signatura Σ).

$u.v \in \overline{O^\mu}(t)$. Por $\mathsf{comp}_\mu(t)$, se tiene $u.v \in O_V(t)$. Por tanto, $v \in O_V(t|_u)$, es decir, $\overline{O^\mu}(t|_u) \subseteq O_V(t|_u)$. De esta manera, se verifica $\mathsf{comp}_\mu(t|_u)$. □

Dado un término t, podemos asociarle una función de reemplazamiento μ_t que haga dicho término μ_t-compatible.

Definición 4.1.6 *Sea Σ una signatura y $t \in \mathcal{T}(\Sigma, V)$. Se define la función de reemplazamiento $\mu_t \in M_{\Sigma(t)}$ como sigue: si $t \in V$, entonces $\mu_t = \mu^\perp_\perp$. Si $t = f(t_1, \ldots, t_k)$, sea $\mu^\epsilon_t \in M_{\{f\}}$ como sigue: $\mu^\epsilon_t(f) = \{i \mid t_i \notin V,\ 1 \leq i \leq ar(f)\}$. Entonces, $\mu_t = \overline{\mu}^\epsilon_t \sqcup \overline{\mu}_{t_1} \sqcup \cdots \sqcup \overline{\mu}_{t_k}$.*

Veamos un ejemplo de cálculo de μ_t para algunos términos.

Ejemplo 4.1.7 *Dado el término $t = \mathtt{first(0,x)}$, tenemos $\Sigma(t) = \{\mathtt{0}, \mathtt{first}\}$. Obtenemos:*

$$\begin{array}{rcl} \mu_{\mathtt{first(0,x)}} & = & \overline{\mu}^\epsilon_{\mathtt{first(0,x)}} \sqcup \overline{\mu}_{\mathtt{0}} \sqcup \overline{\mu}_{\mathtt{x}} \\ & = & \langle \emptyset, \{1\} \rangle \sqcup \langle \emptyset, \emptyset \rangle \sqcup \langle \emptyset, \emptyset \rangle \\ & = & \langle \emptyset, \{1\} \rangle \end{array}$$

Para el término $t' = \mathtt{first(s(x),y::z)}$, tenemos $\Sigma(t') = \{\mathtt{s}, \mathtt{::}, \mathtt{first}\}$, y obtenemos

$$\begin{array}{rcl} \mu_{\mathtt{first(s(x),y::z)}} & = & \overline{\mu}^\epsilon_{\mathtt{first(s(x),y::z)}} \sqcup \overline{\mu}_{\mathtt{s(x)}} \sqcup \overline{\mu}_{\mathtt{y::z}} \\ & = & \langle \emptyset, \emptyset, \{1,2\} \rangle \sqcup \langle \emptyset, \emptyset, \emptyset \rangle \sqcup \langle \emptyset, \emptyset, \emptyset \rangle \\ & = & \langle \emptyset, \emptyset, \{1,2\} \rangle \end{array}$$

A continuación demostramos que μ_t caracteriza la μ-compatibilidad de un término t respecto a cualquier otra función de reemplazamiento μ. Para ello utilizamos algunos resultados auxiliares.

Lema 4.1.8 *Sea Σ una signatura, $t \in \mathcal{T}(\Sigma, V)$ y $f \in \Sigma(t)$. Sea $s = f(\tilde{s}) = t|_u$ con $u \in O(t)$. Si $i \in \mu_s(f)$, entonces $i \in \mu_t(f)$.*

DEMOSTRACIÓN. Por inducción sobre la estructura de la ocurrencia u. Si $u = \epsilon$, el resultado es inmediato. Si $u = j.u'$ y $t = g(\tilde{t})$, por la hipótesis de inducción, $i \in \mu_{t_j}(f)$, para algún $1 \leq j \leq ar(g)$. Puesto que $\mu_t(f) = \mu^\epsilon_t(f) \cup \mu_{t_1}(f) \cup \cdots \cup \mu_{t_{ar(g)}}(f)$, obtenemos la conclusión deseada. □

Lema 4.1.9 *Sea Σ una signatura, $t \in \mathcal{T}(\Sigma, V)$ y $f \in \Sigma(t)$. Si $i \in \mu_t(f)$, existe un $s = f(s_1, \ldots, s_k) = t|_u$ con $u \in O^{\mu_t}(t)$ tal que $s_i \notin V$.*

DEMOSTRACIÓN. Por inducción sobre la estructura de t. Si $t \in \Sigma \cup V$, es trivialmente cierto. Sea $t = g(t_1, \ldots, t_l)$. (1) Si $f = g$, tomamos $u = \epsilon$. Entonces, si $i \in \mu_t(f)$, se tiene que $l = k$ e $i \in \mu^\epsilon_t(f) \cup \mu_{t_1}(f) \cup \cdots \cup \mu_{t_k}(f)$. Si $i \in \mu^\epsilon_t(f)$, entonces $t_i \notin V$ y

concluimos lo afirmado. Si $i \notin \mu_t^{\epsilon}(f)$, entonces $i \in \mu_{t_j}(f)$ para algún t_j, $1 \leq j \leq k$, y, por la hipótesis de inducción, existe un término $s = f(s_1, \ldots, s_k) = t_j|_v$ con $v \in O^{\mu_{t_j}}(t_j)$ que verifica $s_i \notin V$. Puesto que $s_i \notin V$, tenemos que $j \in \mu_{t_j}^{\epsilon}(f)$, y, de hecho, que $j \in \mu_t^{\epsilon}(f)$; por tanto $j \in \mu_t(f)$. Dado que $i \in \mu_t(f)$, concluimos $j.v.i \in O^{\mu_t}(t)$ y con ello demostramos nuestra afirmación. (2) Si $f \neq g$, entonces $\mu_t^{\epsilon}(f) = \emptyset$ y $\mu_t(f) = \mu_{t_1}(f) \cup \cdots \cup \mu_{t_l}(f)$. Por la hipótesis de inducción se sigue la conclusión. □

A continuación establecemos el resultado anunciado: la compatibilidad de un término t respecto a una función de reemplazamiento μ viene determinada por μ_t.

Proposición 4.1.10 *Sea Σ una signatura, $t \in \mathcal{T}(\Sigma, V)$ y $\mu \in M_{\Sigma(t)}$. Entonces, $\mathsf{comp}_{\mu}(t)$ si y sólo si $\mu_t \sqsubseteq \mu$.*

DEMOSTRACIÓN. Para establecer que $\mu_t \sqsubseteq \mu$ implica $\mathsf{comp}_{\mu}(t)$, primero demostramos, por contradicción, que se cumple $\mathsf{comp}_{\mu_t}(t)$. Si $\neg\mathsf{comp}_{\mu_t}(t)$, entonces $O_{\Sigma}(t) \not\subseteq O^{\mu_t}(t)$. Luego existe $u \in O_{\Sigma}(t)$ tal que $u \notin O^{\mu_t}(t)$. Por tanto, debe cumplirse $O_{\Sigma}(t) \neq \emptyset$, es decir, $t \notin V$. Dado que $\epsilon \in O^{\mu_t}(t)$ para todo término t, podemos escribir $u = v.w$, donde $v \in O^{\mu_t}(t)$ es una ocurrencia reemplazante maximal, y $w \neq \epsilon$ (porque $u = v.w \notin O^{\mu_t}(t)$). Puesto que $u \in O_{\Sigma}(t)$, tenemos $v \in O_{\Sigma}(t)$, y podemos escribir t como $t = C[f(\tilde{t})]_v$. Dado que $w \neq \epsilon$, $w = i.w'$, para algún i, $1 \leq i \leq ar(f)$, y siendo v una ocurrencia reemplazante maximal, se tiene $i \notin \mu_t(f)$. Sin embargo, puesto que $u \in O_{\Sigma}(t)$ y $v.i \leq u$, se tiene $v.i \in O_{\Sigma}(t)$. Por tanto, si $s = f(\tilde{t})$, tenemos $i \in \mu_s(f)$. Por el Lema 4.1.8, $i \in \mu_t(f)$. Obtenemos una contradicción. Por tanto, concluimos $\mathsf{comp}_{\mu_t}(t)$. Dado que $\mu_t \sqsubseteq \mu$, por la Proposición 4.1.3, se sigue la conclusión.

Para demostrar la implicación en sentido contrario procederemos también por contradicción. Si $\mu_t \not\sqsubseteq \mu$, por el Lema 3.1.6, existe un símbolo $f \in \Sigma(t)$, tal que $i \in \mu_t(f)$ y $i \notin \mu(f)$. Por el Lema 4.1.9, existe un subtérmino $s = f(s_1, \ldots, s_k) = t|_u$ con $u \in O^{\mu_t}(t)$ tal que $s_i \notin V$. Puesto que $u.i \notin O^{\mu}(t)$, obtenemos $u.i \in \overline{O^{\mu}}(t)$ pero $t|_{u.i} \notin V$. Así pues, t no es μ-compatible. □

Las Proposiciones 4.1.10 y 4.1.4 implican que la función de reemplazamiento μ_t caracteriza por completo la μ-compatibilidad de un término t respecto a cualquier función de reemplazamiento μ.

De manera análoga, dado un conjunto de términos $T \subseteq \mathcal{T}(\Sigma, V)$, podemos definir una función de reemplazamiento $\mu_T = \sqcup_{t \in T} \mu_t$ que hace que T sea μ_T-compatible. Esta función de reemplazamiento también caracteriza la μ-compatibilidad de T con relación a cualquier otra función de reemplazamiento μ.

Proposición 4.1.11 *Sea Σ una signatura, $T \subseteq \mathcal{T}(\Sigma, V)$ y $\mu \in M_{\Sigma(T)}$. Entonces, $\mathsf{comp}_{\mu}(T)$ si y sólo si $\mu_T \sqsubseteq \mu$.*

DEMOSTRACIÓN. Para demostrar que $\mu_T \sqsubseteq \mu$ implica $\mathsf{comp}_\mu(T)$, observemos que, si $\mu_T \sqsubseteq \mu$, por definición de μ_T tenemos que para todo $t \in T$, $\mu_t \sqsubseteq \mu_T \sqsubseteq \mu$. Por la Proposición 4.1.10, $\mathsf{comp}_\mu(t)$ para todo $t \in T$. Por tanto, $\mathsf{comp}_\mu(T)$.

Para demostrar la implicación inversa, procedemos por contradicción. Supongamos que $\mu_T \not\sqsubseteq \mu$. Entonces, existe un $t \in T$ tal que $\mu_t \not\sqsubseteq \mu$. Si no fuera así, puesto que μ_T es el supremo de $\{\mu_t \mid t \in T\}$, tenemos que $\mu_T \sqsubseteq \mu$. Sin embargo, si $\mu_t \not\sqsubseteq \mu$, por la Proposición 4.1.10, $\neg\mathsf{comp}_\mu(t)$. Pero entonces, por definición de $\mathsf{comp}_\mu(T)$, se tiene $\neg\mathsf{comp}_\mu(T)$. □

A partir de la definición de μ_T, es fácil observar que si $T = T' \cup T''$, entonces $\mu_T = \mu_{T'} \sqcup \mu_{T''}$.

4.2 Reescritura sensible al contexto y reescritura

En esta sección aplicamos los conceptos y resultados introducidos en la sección anterior al estudio de los cómputos realizados por reescritura (sin restricciones) y la posibilidad de aproximar éstos mediante la *rsc*. El concepto de término compatible fue introducido considerando la necesidad de extender un emparejamiento parcial existente entre un término t y una parte izquierda l de una regla de reescritura $l \to r$. Si, dada una función de reemplazamiento μ, la parte izquierda l de la regla de reescritura involucrada en el emparejamiento parcial es un término μ-compatible, será posible, en cuanto a las limitaciones que las restricciones de reemplazamiento imponen, extender el emparejamiento parcial. Si deseamos que esta propiedad se verifique para cualquier regla de reescritura del SRTE, tiene sentido establecer una función de reemplazamiento como la descrita por la siguiente definición.

Definición 4.2.1 (Función de reemplazamiento canónica) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE. La función de reemplazamiento $\mu_\mathcal{R}^{com}$ definida como*

$$\mu_\mathcal{R}^{com} = \mu_{L(\mathcal{R})} = \sqcup_{l \in L(\mathcal{R})}\ \mu_l$$

es la función de reemplazamiento canónica *para el SRTE $\mathcal{R}$.*

Teniendo en cuenta la Proposición 4.1.11, $\mu_\mathcal{R}^{com}$ es la *mínima* función de reemplazamiento que hace que todas las partes izquierdas de las reglas del SRTE sean términos compatibles. Por tanto, allá donde empleemos dicha función de reemplazamiento asumiremos este hecho, interpretando, de forma implícita, que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$ significa $\mathsf{comp}_\mu(l)$ para toda $l \in L(\mathcal{R})$, es decir, $O_\Sigma(l) \subseteq O^\mu(l)$ y también $\overline{O^\mu}(l) \subseteq O_V(l)$. Tal como mostraremos más adelante, esta función de reemplazamiento tiene propiedades muy importantes relacionadas con la obtención de formas canónicas mediante la *rsc*. Antes de proseguir, veamos algunos ejemplos de cálculo de la función de reemplazamiento canónica.

Ejemplo 4.2.2 *Consideremos el SRT $\mathcal{R}$:*

```
first(0,x)       → []                  from(x) → x::from(s(x))
first(s(x),y::z) → y::first(x,z)
```

De acuerdo con la Definición 4.2.1,

$$\mu_{\mathcal{R}}^{com} = \mu_{\texttt{first(0,x)}} \sqcup \mu_{\texttt{first(s(x),y::z)}} \sqcup \mu_{\texttt{from(x)}}$$

Teniendo en cuenta el Ejemplo 4.1.7 y puesto que $\mu_{\texttt{from(x)}} = \mu_{\perp}$, obtenemos la siguiente función de reemplazamiento canónica

$$\mu_{\mathcal{R}}^{com}(\texttt{s}) = \mu_{\mathcal{R}}^{com}(\texttt{::}) = \mu_{\mathcal{R}}^{com}(\texttt{from}) = \emptyset \text{ y } \mu_{\mathcal{R}}^{com}(\texttt{first}) = \{1,2\}$$

Ejemplo 4.2.3 *Consideremos el SRT $\mathcal{R}$ obtenido al reunir las reglas que constituyen los SRTs de los Ejemplos 3.3.3 y 4.2.2.*

```
if(true,x,y)     → x                   and(true,x)  → x
if(false,x,y)    → y                   and(false,y) → false
first(0,x)       → []                  0+x          → x
first(s(x),y::z) → y::first(x,z)       s(x)+y       → s(x+y)
from(x)          → x::from(s(x))
```

Para las reglas que proceden del SRT del Ejemplo 3.3.3, obtenemos las siguientes funciones de reemplazamiento asociadas a las distintas lhs's de dichas reglas:

- $\mu_{\texttt{if(true,x,y)}}(\texttt{if}) = \{1\}$.
- $\mu_{\texttt{if(false,x,y)}}(\texttt{if}) = \{1\}$.
- $\mu_{\texttt{and(true,x)}}(\texttt{and}) = \{1\}$.
- $\mu_{\texttt{and(false,x)}}(\texttt{and}) = \{1\}$.
- $\mu_{\texttt{0+x}}(\texttt{+}) = \{1\}$.
- $\mu_{\texttt{s(x)+y}}(\texttt{+}) = \{1\}$ *y* $\mu_{\texttt{s(x)+y}}(\texttt{s}) = \emptyset$.

De acuerdo con lo establecido en el Ejemplo 4.2.2, obtenemos la siguiente función de reemplazamiento canónica:

$$\mu_{\mathcal{R}}^{com}(\texttt{s}) = \mu_{\mathcal{R}}^{com}(\texttt{::}) = \mu_{\mathcal{R}}^{com}(\texttt{from}) = \emptyset,$$

$$\mu_{\mathcal{R}}^{com}(\texttt{if}) = \mu_{\mathcal{R}}^{com}(\texttt{and}) = \mu_{\mathcal{R}}^{com}(\texttt{+}) = \{1\} \text{ y } \mu_{\mathcal{R}}^{com}(\texttt{first}) = \{1,2\}.$$

Resulta interesante comparar el establecimiento de las restricciones de reemplazamiento asociadas a un símbolo de función k-ario, f, (definido mediante las reglas de un programa $\mathcal{R}$), cuando tomamos como criterio de selección de los índices reemplazantes la propiedad de compatibilidad de las partes izquierdas de las reglas de

reescritura de $\mathcal{R}$ (obteniendo así la función de reemplazamiento $\mu_{\mathcal{R}}^{com}$), con el método que emplea información de *strictness*, imponiendo que un argumento i para $1 \leq i \leq k$ sea reemplazante sólo si la función f es i-estricta (obteniendo así una función de reemplazamiento que denotaremos como $\mu_{\mathcal{R}}^{str}$). Mediante algunos ejemplos, podemos demostrar que no existe una correspondencia clara entre $\mu_{\mathcal{R}}^{com}$ y $\mu_{\mathcal{R}}^{str}$.

Ejemplo 4.2.4 *Consideremos la función* `f` *definida mediante el programa*

```
f(x) → x
```

La función `f` *es estricta y, por tanto,* $1 \in \mu_{\mathcal{R}}^{str}(\mathtt{f})$. *Sin embargo,* $1 \notin \mu_{\mathcal{R}}^{com}(\mathtt{f})$.

Ejemplo 4.2.5 *Consideremos la función* `first` *definida mediante el programa del Ejemplo 4.2.2. Tenemos que* $2 \in \mu_{\mathcal{R}}^{com}(\mathtt{first})$ *pero* `first` *no es 2-estricta, puesto que* `first(0,`$\perp$`)=[]`$\neq \perp$. *Luego* $2 \notin \mu_{\mathcal{R}}^{str}(\mathtt{first})$.

Ahora investigaremos las propiedades de la *rsc* cuando se consideran funciones de reemplazamiento μ mayores o iguales que la función de reemplazamiento canónica $\mu_{\mathcal{R}}^{com}$, es decir, $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$.

Observación 4.2.6 *Como veremos a continuación, y en los capítulos 11, 12 y 13, la completitud de los cómputos mediante la* rsc *viene determinada por la función de reemplazamiento canónica* $\mu_{\mathcal{R}}^{com}$. *Dado un SRTE* $\mathcal{R}$, *sólo podremos asegurar que los cómputos realizados bajo el control de una función de reemplazamiento* μ *obtienen formas estables, valores, etc. si, como mínimo, se cumple* $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. *El Ejemplo 4.2.5 demuestra que, al fijar la función de reemplazamiento empleando información de* strictness, *no siempre obtenemos una función de reemplazamiento mayor que* $\mu_{\mathcal{R}}^{com}$. *Como consecuencia, el empleo de información de* strictness no es un método válido para definir funciones de reemplazamiento que aseguren la completitud de los cómputos realizados mediante la *rsc*. *Esto contrasta de manera significativa con la práctica, habitual en la programación funcional, de establecer las anotaciones asociadas a los símbolos de programa empleando información de* strictness.

Básicamente, la siguiente proposición establece condiciones suficientes para que toda (multi)derivación que lleva a una instancia de un término lineal l pueda ser simulada por *rsc* obteniéndose otra instancia del mismo término lineal.

Proposición 4.2.7 *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRTE lineal por la izquierda,* $l \in \mathcal{T}(\Sigma, V)$ *un término lineal y* $\mu \in M_\Sigma$ *tal que* $\mu_{\mathcal{R}}^{com} \sqcup \mu_l \sqsubseteq \mu$. *Si existe una multiderivación* $A : t \Vdash^* \sigma(l)$, *para alguna sustitución* $\sigma \in Subst(\Sigma, V)$, *entonces existe* $\theta \in Subst(\Sigma, V)$ *tal que* $t \Vdash^* \theta(l)$ *y, para todo* $x \in Var(l)$, *tenemos que* $B_x : \theta(x) \Vdash^* \sigma(x)$ *y* $|B_x| \leq |A|$.

DEMOSTRACIÓN. Por inducción noetheriana, empleando el siguiente orden estricto bien fundado $<_M$ entre multiderivaciones: sean $A : t \Vdash^* s$ y $A' : t' \Vdash^* s'$ dos multiderivaciones. Escribimos $A <_M A'$ si $(|A|, s)\ (<, \prec)_{lex}\ (|A'|, s')$, es decir, o bien $|A| < |A'|$

(la derivación A es más corta que A') o bien $|A| = |A'|$ y $s \prec s'$ (ambas derivaciones son de igual longitud pero el término obtenido mediante A es un subtérmino propio del obtenido mediante A').

El caso base (derivaciones minimales respecto a $<_M$) que considera multiderivaciones A cuya longitud es cero ($|A| = 0$) es inmediato.

Para el paso de inducción, consideremos $n = |A| > 0$. Sea $A : t \Vdash^* t' \xrightarrow{U} \sigma(l)$, donde la longitud de $A' : t \Vdash^* t'$ es $|A'| = n - 1$, y $U \subseteq O_{\mathcal{R}}(t')$, es tal que $\forall u, u' \in U$, $u \neq u' \Rightarrow u \parallel u'$. Consideramos dos casos: $\epsilon \in U$ y $\epsilon \notin U$.

1. Si $\epsilon \in U$, entonces, puesto que las ocurrencias en U son disjuntas dos a dos, debe tenerse $U = \{\epsilon\}$. Entonces t' es un redex, y existe una sustitución σ' y una regla $l' \to r'$ tal que $A : t \Vdash^* \sigma'(l') \xrightarrow{\{\epsilon\}} \sigma'(r') = \sigma(l)$. Dado que $A' <_M A$, y $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ (puesto que $l' \in L(\mathcal{R})$, $\mathcal{R}$ es lineal por la izquierda y $\mu_{l'} \sqsubseteq \mu^{com}_{\mathcal{R}}$, aseguramos las asunciones iniciales para el paso de inducción) por la hipótesis de inducción existe una θ' tal que $t \Vdash^* \theta'(l')$ y, para toda $x \in Var(l')$, tenemos $B'_x : \theta'(x) \Vdash^* \sigma'(x)$, y $|B'_x| \leq |A'|$. Puesto que las variables extra pueden instanciarse arbitrariamente en cada paso de reducción, sin pérdida de generalidad podemos asumir que, para toda variable extra $y \in Var(r') \backslash Var(l')$ de la regla $l' \to r'$ que se aplica, se verifica que $\theta'(y) = \sigma'(y)$. Observemos que $\theta'(l') \xhookrightarrow{\{\epsilon\}} \theta'(r')$. Definimos la multiderivación $B' : \theta'(r') \Vdash^* \sigma'(r')$ como sigue. Sea $r' = C[x_1, \ldots, x_m]$, donde $x_1, \ldots, x_m$ son todas las variables en $Var(r') \cap Var(l')$, quizás con algunas repeticiones, es decir, $m \geq |Var(r') \cap Var(l')|$. Sean $w_1, \ldots, w_m$ las ocurrencias (disjuntas) de variables $x_1, \ldots, x_m$ en r': $x_i = r'|_{w_i}$ para $1 \leq i \leq m$.

   Primero, escribimos cada multiderivación $B'_{x_i} : \theta'(x_i) \Vdash^* \sigma'(x_i)$ como $\theta'(x_i) = s_i^1 \xrightarrow{V_i^1} s_i^2 \to \cdots \to s_i^{p_i} \xrightarrow{V_i^{p_i}} s_i^{p_i+1} = \sigma'(x_i)$ para $1 \leq i \leq m$. Aquí, $p_i = |B'_{x_i}|$. Entonces, B' es $B' : r^1 \xrightarrow{W_1} r^2 \to \cdots \to r^p \xrightarrow{W_p} r^{p+1}$, donde $p = max\{p_i \mid 1 \leq i \leq m\}$, y:

   - $r^1 = C[s_1^1, \ldots, s_m^1] = C[\theta'(x_1), \ldots, \theta'(x_m)]$.
   - Para todo j, $1 < j \leq p + 1$, $r^j = C[r_1^j, \ldots, r_m^j]$ es tal que, para todo i, $1 \leq i \leq m$, $r_i^j = s_i^j$ si $j \leq p_i + 1$ y $r_i^j = r_i^{j-1}$ en otro caso.
   - Para todo j, $1 \leq j \leq p$, sea $RedOc(j) = \{i \mid 1 \leq i \leq m \wedge j \leq p_i\}$ en la expresión $W_j = \bigcup_{i \in RedOc(j)} w_i.V_i^j$.

   Las variables extra $\{y_1, \ldots, y_{m'}\} \subseteq Var(r') \backslash Var(l')$ presentes en r' son instanciadas por θ' haciendo $\theta'(y_j) = \sigma'(y_j)$ para $1 \leq j \leq m'$. Puesto que ocupan posiciones disjuntas entre sí y respecto a cualquier otra variable en r', dicha instanciación no resulta alterada durante la derivación B'. Claramente,

$\theta'(r') = r^1$ y $\sigma'(r') = r^{p+1}$. Puesto que $|B'_{x_i}| \leq |A'|$ y $|A'| = n-1$, tenemos $p_i \leq n-1$ para cada i, $1 \leq i \leq m$, y la longitud de la multiderivación B' es $|B'| = p = max\{p_i \mid 1 \leq i \leq m\} < n$. Entonces, puesto que $\sigma'(r') = \sigma(l)$, tenemos una multiderivación $B' : \theta'(r') \Vdash^* \sigma(l)$ tal que $B' <_M A$. Por la hipótesis de inducción, $\theta'(r') \Vdash^* \theta(l)$, y para todo $x \in Var(l)$, $B_x : \theta(x) \Vdash^* \sigma(x)$ y $|B_x| \leq |B'| < |A|$. Puesto que, $t \Vdash^* \theta'(l') \overset{\{\epsilon\}}{\hookrightarrow} \theta'(r') \Vdash^* \theta(l)$, es decir, $t \Vdash^* \theta(l)$, se sigue el resultado deseado.

2. Si $\epsilon \notin U$, descomponemos el conjunto U en un conjunto $U_{ov} = \{u \in U \mid u \in O_\Sigma(l)\}$ de ocurrencias de U que 'solapan' l, y un conjunto $\overline{U}_{ov} = U \backslash U_{ov}$. Entonces:

   - Para todo $u \in \overline{U}_{ov}$, o bien $u \notin O(l)$ o bien $u \in O_V(l)$. Podemos descomponer $\overline{U}_{ov}$ como sigue: $\overline{U}_{ov} = \overline{U}_1 \uplus \cdots \uplus \overline{U}_p$ tal que, para todo j, $1 \leq j \leq p$, existe $y_j \in Var(l)$ y $v_j \in O_V(l)$ tales que $l|_{v_j} = y_j$, y $\overline{U}_j = \{u \in \overline{U}_{ov} \mid \exists w.\ u = v_j.w\} = v_j.W_j$. Sea $Y = \{y_1, \ldots, y_p\}$.
   - Considerando $U_{ov} = \{u_1, \ldots, u_q\}$, definimos $l' = l[z_1, \ldots, z_q]$, donde las z_i son variables nuevas y distintas entre sí, tales que $z_i \notin Var(l)$ y $l'|_{u_i} = z_i$ para todo i, $1 \leq i \leq q$. Sea $Z = \{z_1, \ldots, z_q\}$. Observemos que l' es lineal. Aún más, por la Proposición 4.1.5 y la Proposición 4.1.10, $\mu_{l'} \sqsubseteq \mu_l \sqsubseteq \mu$. También, observemos que $Y \subseteq Var(l')$, porque las occurrencias en U_{ov} y $\overline{U}_{ov}$ son disjuntas entre sí.

   Entonces, podemos expresar el término t' en $A : t \Vdash^* t' \overset{U}{\rightarrow} \sigma(l)$ utilizando una sustitución σ', tal que $t' = \sigma'(l')$. La sustitución σ' se define como sigue:

   (a) $\sigma'(x) = \sigma(x)$ para todo $x \in Var(l') \backslash (Y \cup Z)$,

   (b) $\sigma'(y_j) = t'|_{v_j}$ para $1 \leq j \leq p$. Por tanto,
   $C_{y_j} : \sigma'(y_j) = t'|_{v_j} \overset{W_j}{\rightarrow} \sigma(l)|_{v_j}$. Adviértase que $|C_{y_j}| = 1$. Recordemos que $W_j = \{w \mid v_j.w \in \overline{U}_j\}$.

   (c) $\sigma'(z_i) = t'|_{u_i}$ para $1 \leq i \leq q$.

   Por tanto, teniendo en cuenta 2a y 2b, se sigue que, para todo $x \in Var(l[\,])$ (Z y $Var(l)$ son disjuntos), tenemos derivaciones $C_x : \sigma'(x) \Vdash^\epsilon \sigma(x)$ tales que $|C_x| \leq 1$ (recordemos que $t \Vdash^\epsilon s$ significa que $t \Vdash s$ o bien $t = s$).

   Ahora escribimos $A : t \Vdash^* \sigma'(l') \overset{U}{\rightarrow} \sigma(l)$. Sea $A' : t \Vdash^* \sigma'(l')$. Puesto que $|A'| < |A|$, por la hipótesis de inducción existe una sustitución θ' tal que $t \Vdash^* \theta'(l')$ y para toda $x \in Var(l')$, existe una $B'_x : \theta'(x) \Vdash^* \sigma'(x)$ tal que $|B'_x| \leq |A'|$. Adviértase que, para toda $x \in Var(l[\,])$, tenemos una derivación $B_x : \theta'(x) \Vdash^* \sigma'(x) \Vdash^\epsilon \sigma(x)$. Por tanto, $|B_x| = |B'_x C_x| \leq |B'_x| + 1 \leq |A'| + 1 = |A|$

para cada $x \in Var(l[\ ])$. Para las variables $z_i \in Z$, tenemos derivaciones $C_{z_i} : \theta'(z_i) \Vdash^* \sigma'(z_i)$ tales que $|C_{z_i}| \leq |A'|$. Puesto que $\sigma'(z_i) = t'|_{u_i} \rightarrow \sigma(l|_{u_i})$, tenemos que $C'_{z_i} : \theta'(z_i) \Vdash^* \sigma(l|_{u_i})$ y $|C'_{z_i}| \leq |A|$. Puesto que $u_i \neq \epsilon$, tenemos $\sigma(l|_{u_i}) \prec \sigma(l)$. Por tanto, $C'_{z_i} <_M A$. Por la hipótesis de inducción, para cada i, $1 \leq i \leq q$, existe una sustitución θ_i tal que $\theta'(z_i) \Vdash^* \theta_i(l|_{u_i})$ y, para toda $x \in Var(l|_{u_i})$, existen derivaciones $B_x : \theta_i(x) \Vdash^* \sigma(x)$ tales que $|B_x| \leq |C'_{z_i}| \leq |A|$. A partir de estos hechos:

- Definimos θ como $\theta = \theta' \cup \theta_1 \cup \cdots \cup \theta_q$. Puesto que $Var(l) = Var(l[\ ]) \uplus Var(l|_{u_1}) \uplus \cdots \uplus Var(l|_{u_q})$, y l es lineal por la izquierda, θ está bien definida, y cubre todos los enlaces para las variables en $Var(l)$. Por tanto, obtenemos que cada $B_x : \theta(x) \Vdash^* \sigma(x)$ verifica $|B_x| \leq |A|$ para todo $x \in Var(l)$.
- Tenemos que $t \Vdash^* \theta'(l')$. Puesto que $l' = l[z_1, \ldots, z_q]$, obtenemos $\theta'(l') = \theta'(l[z_1, \ldots, z_q]) = \theta'(l)[\theta'(z_1), \ldots, \theta'(z_q)]$. Por tanto, obtenemos la derivación $t \Vdash^* \theta'(l)[\theta'(z_1), \ldots, \theta'(z_q)]$. También tenemos $\theta'(z_i) \Vdash^* \theta_i(l|_{u_i})$ para todo i, $1 \leq i \leq q$. Puesto que $\mu_l \sqsubseteq \mu$, $O_\Sigma(l) \subseteq O^\mu(l)$. Por la Proposición 3.2.9, $O^\mu(l) \subseteq O^\mu(\theta'(l))$. Por tanto, dado que $u_i \in O_\Sigma(l)$, para todo i, $1 \leq i \leq q$, obtenemos $u_i \in O^\mu(\theta'(l))$ y por las Proposiciones 3.2.10 y 3.3.6, $\theta'(l)[\theta'(z_1), \ldots, \theta'(z_q)] \Vdash^* \theta'(l)[\theta_1(l|_{u_1}), \ldots, \theta_q(l|_{u_q})]$.
  Por definición de θ, $\theta'(l)[\theta_1(l|_{u_1}), \ldots, \theta_q(l|_{u_q})] = \theta(l)$. Por tanto, $t \Vdash^* \theta(l)$.

Esto completa la demostración. □

En la siguiente proposición escribimos $\Vdash^*_{>\epsilon}$ para denotar una *multiderivación* que no considera redexes a la ocurrencia vacía ϵ. El siguiente teorema establece la capacidad de la *rsc* para, dado un término t que se reduce a una forma estable s cuyo símbolo a la raíz es f, obtener un término con el mismo símbolo a la raíz f.

Teorema 4.2.8 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda, y $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma, V)$ y $s \in \mathcal{T}(\Sigma, V)$ una forma estable. Si $t \Vdash^* s$, entonces existe un término s' tal que $t \Vdash^* s'$, $root(s) = root(s')$, y $s' \Vdash^*_{>\epsilon} s$.*

DEMOSTRACIÓN. Si t es estable, el resultado es inmediato. Si t no es estable, y se reduce a una forma estable s, existe una derivación $A : t \Vdash^* \sigma(l) \overset{\{\epsilon\}}{\rightarrow} \sigma(r) \Vdash^*_{>\epsilon} s$ para alguna regla $l \rightarrow r$. Escribimos $A' : t \Vdash^* \sigma(l)$ y $A'' : \sigma(r) \Vdash^*_{>\epsilon} s$. Adviértase que $|A| = |A'| + 1 + |A''|$. Procedemos por inducción en la longitud $|A| = n \geq 1$ de la derivación A. Si $n = 1$, entonces t es un redex. Puesto que $t = \sigma(l) \hookrightarrow \sigma(r) = s$, tomamos $s' = \sigma(r) = s$. Si $n > 1$, aplicamos la Proposición 4.2.7 a la derivación A'. Por tanto, existe una sustitución θ tal que $t \Vdash^* \theta(l)$ y $B_x : \theta(x) \Vdash^* \sigma(x)$ para todo $x \in Var(l)$ tal que $|B_x| \leq |A'|$. Procediendo como en el caso 1 de la demostración de la Proposición 4.2.7, podemos definir una multiderivación $B' : \theta(r) \Vdash^* \sigma(r)$ tal

que $|B'| \leq |A'|$. De esta manera, la derivación $B : \theta(r) \Vdash^* \sigma(r) \Vdash^*_{>\epsilon} s$ verifica $|B| = |B'|+|A''| \leq |A'|+|A''| < |A|$. Si $r = g(\tilde{r})$, entonces $root(s) = g$ y, por construcción, la derivación B' no reduce una ocurrencia ϵ de un redex. Entonces, tomamos $s' = \theta(r)$, y, puesto que $s' = \theta(r) \Vdash^*_{>\epsilon} \sigma(r) \Vdash^*_{>\epsilon} s$, $t \Vdash^* \theta(l) \overset{\{\epsilon\}}{\hookrightarrow} \theta(r) = s'$, y $root(s') = g$ se sigue la conclusión. Si $r \in V$, entonces, si $\theta(r)$ es estable, también tenemos $\theta(r) \Vdash^*_{>\epsilon} \sigma(r) \Vdash^*_{>\epsilon} s$ y tomamos $s' = \theta(r)$. Claramente, $root(s') = root(s)$. Si $\theta(r)$ no es estable, puesto que la derivación B verifica $|B| < |A|$, por la hipótesis de inducción existe un s' tal que $\theta(r) \Vdash^* s'$, $s' \Vdash^*_{>\epsilon} s$, y $root(s) = root(s')$. Por ello, puesto que $t \Vdash^* \theta(l) \overset{\{\epsilon\}}{\hookrightarrow} \theta(r) \Vdash^* s'$, concluimos la demostración. □

Tal como muestra el siguiente ejemplo, la exigencia de linealidad por la izquierda no puede relajarse sin comprometer el resultado anterior.

Ejemplo 4.2.9 *Consideremos el SRT $\mathcal{R}$:*

```
f(x,x) → 0        h(0) → 0
g(0,x) → 0
```

Este SRT no es lineal por la izquierda. El término $t =$ `f(g(s(0),h(0)),g(s(0),0))` *no es una forma estable, puesto que podemos escribir la siguiente derivación:*

`f(g(s(0),`$\underline{\texttt{h(0)}}$`),g(s(0),0))` → $\underline{\texttt{f(g(s(0),0),g(s(0),0))}}$ → `0`

que reduce t *a un redex y, finalmente, a un reducto estable* `0`*. Tenemos que* $\mu^{com}_{\mathcal{R}}(\texttt{f}) = \emptyset$, $\mu^{com}_{\mathcal{R}}(\texttt{g}) = \mu^{com}_{\mathcal{R}}(\texttt{h}) = \{1\}$*. Puesto que* 1.2 *no es una ocurrencia reemplazante de* t*, tenemos que*

`f(g(s(0),`$\underline{\texttt{h(0)}}$`),g(s(0),0))` $\not\hookrightarrow$

Esto impide que la rsc *obtenga un reducto cuyo símbolo a la raíz coincide con* `0`*, violándose el Teorema 4.2.8.*

El siguiente ejemplo muestra que, en general, la condición $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ no puede ser eliminada de las premisas del Teorema 4.2.8.

Ejemplo 4.2.10 *Consideremos el SRT $\mathcal{R}$:*

```
f(x,y) → g(x,y)        a → b
g(b,b) → b
```

y la función de reemplazamiento μ *tal que* $\mu(\texttt{f}) = \mu(\texttt{g}) = \{1\}$ *(que no verifica* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ *debido a la regla* `g(b,b) → b`*). Tenemos la siguiente* μ*-derivacion:*

$\underline{\texttt{f(b,a)}} \hookrightarrow$ `g(b,a)`

donde `g(b,a)` *es una forma* μ*-normal de* `f(b,a)`*, pero* no *es una forma estable de* `f(b,a)` *puesto que*

`g(b,`$\underline{\texttt{a}}$`)` → $\underline{\texttt{g(b,b)}}$ → `b`

Dado que la μ-derivación no continúa mas allá del término `g(b,a)`*, no obtiene un término cuyo símbolo a la raíz es* `b`*.*

Sin embargo, también podemos dar un ejemplo de que, aunque la condición $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$ es suficiente, no es necesaria.

Ejemplo 4.2.11 *Consideremos el SRT $\mathcal{R}$:*

```
g(b,b)      → b        f(x,y) → g(x,y)
g(b,a)      → b        a → b
g(b,f(x,y)) → b
g(b,g(x,y)) → b
```

En este caso es posible, incluso si establecemos μ(`g`*) = {1}, aproximar una derivación de estabilización como*

$$\underline{\texttt{f(b,a)}} \rightarrow \texttt{g(b,}\underline{\texttt{a}}\texttt{)} \rightarrow \underline{\texttt{g(b,b)}} \rightarrow \texttt{b}$$

Esto es debido a que no va a ser necesario reducir el segundo argumento para extender un emparejamiento parcial de las reglas para la función `g` *puesto que hemos proporcionado de forma* explícita *todas las posibles formas de extenderlo, con lo que cualquier término situado allí hará progresar el emparejamiento. Este comportamiento es (casi) equivalente al que tendría una única regla*

```
g(b,x) → b
```

que sustituyera las cuatro reglas que definen `g`*.*

A partir de aquí, utilizaremos la Proposición 4.2.7 y el Teorema 4.2.8 aplicándolos a derivaciones, más que a multiderivaciones. La correspondencia existente entre multiderivaciones y derivaciones (Observación 2.3.7) nos autoriza a proceder así. Pueden realizarse, además, consideraciones similares para $\hookrightarrow$ y $\Vdash$, teniendo en cuenta las propiedades correspondientes de las ocurrencias reemplazantes (ver el Lema 3.3.10).

4.3 Caracterización de las formas μ-normales

Nuestro primer resultado establece que los contextos reemplazantes máximos de las formas μ-normales son contextos rígidos.

Proposición 4.3.1 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma, V)$ una forma μ-normal y $C[\,] = CMR^{\mu}(t)$. Entonces, $C[\widetilde{\Omega}]$ es un término rígido y fuertemente estable.*

Demostración. En primer lugar demostramos que $C[\widetilde{\Omega}]$ es rígido, es decir, $\omega(C[\widetilde{\Omega}]) = C[\widetilde{\Omega}]$. Procedemos por inducción en la estructura de $s = C[\widetilde{\Omega}]$. El caso $C[\widetilde{\Omega}] = \Omega$ no es posible puesto que $CMR^{\mu}(t)$ no puede ser el contexto vacío. Si $C[\widetilde{\Omega}] = x \in V$, el resultado es inmediato.

Si $C[\widetilde{\Omega}] = f(C_1[\widetilde{\Omega}], \ldots, C_{ar(f)}[\widetilde{\Omega}])$, y dado que, de acuerdo con la Definición 3.2.5, $C_i[\widetilde{\Omega}] = CMR^{\mu}(t_i)$ si $i \in \mu(f)$ y $C_i[\widetilde{\Omega}] = \Omega$ si $i \notin \mu(f)$, podemos asumir en cualquiera de los dos casos, que cada $C_i[\widetilde{\Omega}]$ es rígido para cada $1 \leq i \leq ar(f)$ (bien por la hipótesis de inducción o porque, trivialmente, $\omega(\Omega) = \Omega$). Por tanto, si $C[\widetilde{\Omega}]$ no es rígido, existe un esquema de redex $l \in L_{\Omega}(\mathcal{R})$ tal que $s \uparrow l$. Sea $C'[\,]$ un contexto máximo tal que $s = C'[s_1, \ldots, s_n]$ y $l = C'[l_1, \ldots, l_n]$, es decir, $C'[\,]$ es el prefijo común máximo de s y l. Dado que $s \uparrow l$, por el Lema 2.5.8, para todo $1 \leq j \leq m$, o bien $s_j \leq l_j$ (y entonces $s_j = \Omega$) o bien $s_j \geq l_j$ (y entonces $l_j = \Omega$). Existe al menos un i, $1 \leq i \leq n$ tal que $s_i < l_i$. En otro caso, tendríamos $s_j \geq l_j$ para todo $1 \leq j \leq n$ y, por tanto, $s \geq l$. Pero en ese caso, puesto que $t \geq s \geq l$, y $\mathcal{R}$ es lineal por la izquierda, por el Lema 2.3.3, t también es un redex. Ahora bien, dado que $\epsilon \in O^{\mu}(t)$, t no es una forma μ-normal. Puesto que $s_i < l_i$, tenemos que $s_i = \Omega = s|_v$ y $l_i \neq \Omega$. Por tanto, $v \in O_{\Sigma}(l)$. Puesto que t no contiene Ω-ocurrencias, de acuerdo con la definición de s, y por la Proposición 3.2.6, tenemos $v \in \overline{O^{\mu}}(s)$. No obstante, dado que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, y debido a que $v \in O_{\Sigma}(l)$, tenemos que $v \in O^{\mu}(l)$. Puesto que $sprefix_s(v) = sprefix_l(v)$, por la Proposición 3.2.3, tenemos que $v \in O^{\mu}(s)$ lo cual supone una contradicción.

Ahora demostramos que $\omega(C[\widetilde{\Omega}]) > \Omega$, es decir, que $C[\widetilde{\Omega}]$ es fuertemente estable. Dado que $C[\,]$ no puede ser el contexto vacío, tenemos que $C[\widetilde{\Omega}] > \Omega$. Por tanto, puesto que $C[\widetilde{\Omega}]$ es rígido, $\Omega < C[\widetilde{\Omega}] = \omega(C[\widetilde{\Omega}])$, lo cual completa la demostración. □

La linealidad por la izquierda no se ha impuesto arbitrariamente, tal como muestra el siguiente ejemplo.

Ejemplo 4.3.2 *El SRT*

`f(x,x)` $\rightarrow$ `x`

impone que $\mu^{com}_{\mathcal{R}}$(`f`) $= \emptyset$. *Por tanto,* `f(`$\Box$`,`$\Box$`)` *es el contexto reemplazante máximo de* `f(a,b)`. *Sin embargo, a pesar de que* `f(a,b)` *es una forma* $\mu^{com}_{\mathcal{R}}$*-normal, el* Ω*-término* `f(`Ω`,`Ω`)` *no es rígido ni tampoco fuertemente estable, puesto que* `f(`Ω`,`Ω`)`$\rightarrow_{\Omega} \Omega$.

La condición $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ tampoco se ha exigido en vano:

Ejemplo 4.3.3 *Consideremos el SRT del Ejemplo 1.1.2. Si hacemos* μ(`first`) $= \emptyset$, *tenemos que* $\mu \sqsubset \mu^{com}_{\mathcal{R}}$. *Entonces* `first(zero,x)` *es una forma* μ*-normal. Su contexto reemplazante máximo es* `first(`$\Box$`,`$\Box$`)`. *No obstante,* `first(`Ω`,`Ω`)` *no es rígido puesto que* `first(`Ω`,`Ω`)` $\uparrow$ `first(0,`Ω`)`. *Por tanto,* ω(`first(`Ω`,`Ω`)`$) = \Omega$.

Por último, si t no es una forma μ-normal, $CMR^{\mu}(t)$ no tiene porqué ser rígido.

Ejemplo 4.3.4 *Si consideramos el SRT* $\mathcal{R}$ *del Ejemplo 2.5.11, el término* $t =$ `f(b)` *no es una forma* $\mu^{com}_{\mathcal{R}}$*-normal (puesto que* $\mu^{com}_{\mathcal{R}}$(`f`)$ = \{1\}$ *y* `b` *es un redex). Dado que* $C[\,] = CMR^{\mu^{com}_{\mathcal{R}}}(t) = t = C[\widetilde{\Omega}]$ *y* $\omega(t) = \Omega < t$, *concluimos que* $CMR^{\mu^{com}_{\mathcal{R}}}(t)$ *no es rígido.*

Como consecuencia de la Proposición 4.3.1, obtenemos la siguiente propiedad que establece la capacidad de las formas μ-normales para *aproximar* cualquier posible reducto suyo (mediante reescritura no restringida).

Proposición 4.3.5 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma, V)$ una forma μ-normal y $C[\,] = CMR^\mu(t)$. Si $t \to^* s$, entonces $C[\widetilde{\Omega}] \leq s$, y s es fuertemente estable.*

DEMOSTRACIÓN. Por la Proposición 4.3.1, $\omega(C[\widetilde{\Omega}]) = C[\widetilde{\Omega}]$. Puesto que $t \to^* s$, por la Proposición 2.5.9, tenemos que $\omega(t) \leq \omega(s) \leq s$. Dado que $C[\widetilde{\Omega}] \leq t$, de nuevo por la Proposición 2.5.9 tenemos $\Omega < C[\widetilde{\Omega}] = \omega(C[\widetilde{\Omega}]) \leq \omega(t) \leq \omega(s) \leq s$, lo que concluye la demostración. □

Las formas μ-normales contienen prefijos estables comunes a cualquier reducto (alcanzado mediante reescritura no restringida) que pueda obtenerse a partir de ellas. A continuación obtenemos algunos corolarios sencillos de la Proposición 4.3.5. La primera proposición establece las condiciones que aseguran que el conjunto de formas normales es cerrado bajo reescritura (no restringida).

Proposición 4.3.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma, V)$ una forma μ-normal. Si $t \to^* s$, entonces s es una forma μ-normal.*

DEMOSTRACIÓN. Si $t \to^* s$, por la Proposición 4.3.5, $C[\widetilde{\Omega}] \leq s$. Por la Proposición 3.2.6 y la Proposición 3.2.8, toda ocurrencia de s que no se encuentra en $C[\widetilde{\Omega}]$ es no reemplazante. Siendo $C[\widetilde{\Omega}]$ rígido, no existen ocurrencias de redexes en s que solapen $C[\,]$ (ver la demostración de la Proposición 4.3.1), luego todo redex en s es no reemplazante, es decir, s es una forma μ-normal. □

La condición $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$ es necesaria para asegurar que las formas μ-normales son cerradas bajo reducciones arbitrarias (el Ejemplo 4.2.10 muestra que si no se cumple esta condición, el conjunto de formas μ-normales puede tener reductos que no sean μ-normales). Observemos que la Proposición 4.3.6 no significa que si $\bar{t}$ es una forma μ-normal de t y s es un reducto de $\bar{t}$, entonces s es una forma μ-normal de t. En general no se tendrá que $t \hookrightarrow^* s$, condición indispensable para hablar de s como de una forma μ-normal *de* t.

Los dos siguientes resultados caracterizan las formas μ-normales como términos fuertemente estables (y, por tanto, estables).

Teorema 4.3.7 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Toda forma μ-normal es fuertemente estable.*

DEMOSTRACIÓN. Sea $C[\,] = CMR^{\mu}(t)$. Dado que $t \rightarrow^{*} t$, por la Proposición 4.3.5, $C[\widetilde{\Omega}] \leq t$. Por la Proposición 2.5.9, $\omega(C[\widetilde{\Omega}]) \leq \omega(t)$. La Proposición 4.3.5 también asegura que $C[\widetilde{\Omega}]$ es fuertemente estable, es decir, $\omega(C[\widetilde{\Omega}]) \neq \Omega$. Luego $\omega(t) \neq \Omega$ lo cual concluye la demostración. □

Teorema 4.3.8 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y $\mu \in M_{\Sigma}$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Toda forma μ-normal es estable.*

DEMOSTRACIÓN. Inmediata, a partir del Teorema 4.3.7 y la Proposición 2.5.10. □

Al considerar el Ejemplo 4.2.9, se advierte la necesidad de imponer la linealidad por la izquierda para establecer el Teorema 4.3.8. El Ejemplo 4.2.10 también muestra que la condición $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ no ha sido impuesta en vano.

Si un SRTE es confluente, se verifica la propiedad de *unicidad de formas normales*: si un término tiene forma normal, ésta es única.

Ejemplo 4.3.9 *Aunque la presencia de variables extra en las reglas de reescritura hace que sea difícil obtener una relación de reescritura confluente, existen ejemplos de SRTEs confluentes. El SRTE $\mathcal{R}$:*

$$\begin{aligned} \mathtt{f(x)} &\rightarrow \mathtt{a} \\ \mathtt{a} &\rightarrow \mathtt{f(x)} \\ \mathtt{c} &\rightarrow \mathtt{b} \end{aligned}$$

es confluente (puede demostrarse que $\rightarrow$ es fuertemente confluente, lo cual implica la confluencia del sistema). Los únicos términos con forma normal son `c` *y* `b`.

Sin embargo, dada una función de reemplazamiento μ, la confluencia del SRTE no implica la propiedad análoga para las formas μ-normales.

Ejemplo 4.3.10 *Consideremos el SRT ortogonal (por tanto confluente) $\mathcal{R}$:*

$$\begin{aligned} \mathtt{f(x,y)} &\rightarrow \mathtt{c(x)} \\ \mathtt{a} &\rightarrow \mathtt{b} \end{aligned}$$

y la función de reemplazamiento μ tal que $\mu(\mathtt{f}) = \{1\}$ y $\mu(\mathtt{c}) = \emptyset$. Observemos que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Sin embargo, tenemos las siguientes derivaciones μ-normalizantes:

$$\underline{\mathtt{f(a,b)}} \rightarrow \mathtt{c(a)}$$

y

$$\mathtt{f}(\underline{\mathtt{a}},\mathtt{b}) \rightarrow \underline{\mathtt{f(b,b)}} \rightarrow \mathtt{c(b)}$$

que obtienen formas μ-normales `c(a)` *y* `c(b)` *distintas.*

Sin embargo, se garantiza una propiedad de unicidad más débil: la unicidad de contextos reemplazantes máximos.

Proposición 4.3.11 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y confluente. Sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Si $\bar{s}', \bar{s}''$ son formas μ-normales de t, entonces $CMR^\mu(\bar{s}') = CMR^\mu(\bar{s}'')$.*

DEMOSTRACIÓN. Sean $C'[\ ] = CMR^\mu(\bar{s}')$ y $C''[\ ] = CMR^\mu(\bar{s}'')$. Dado que $\mathcal{R}$ es confluente, existe un término s tal que $\bar{s}' \rightarrow^* s$ y $\bar{s}'' \rightarrow^* s$. Por la Proposición 4.3.5, $C'[\widetilde{\Omega}] \leq s$ y $C''[\widetilde{\Omega}] \leq s$. Por tanto, $C'[\widetilde{\Omega}] \uparrow C''[\widetilde{\Omega}]$. Entonces, existe un contexto máximo $C[\ ]$ tal que $C'[\widetilde{\Omega}] = C[t'_1, \ldots, t'_n]$ y $C''[\widetilde{\Omega}] = C[t''_1, \ldots, t''_n]$ y o bien $t'_i \leq t''_i$ (y entonces $t'_i = \Omega$) o bien $t'_i \geq t''_i$ (y entonces $t''_i = \Omega$) para $1 \leq i \leq n$. Sin pérdida de generalidad, podemos escoger un i, $1 \leq i \leq n$ tal que $t'_i = \Omega$. Sea $u \in O(C'[\widetilde{\Omega}])$ tal que $t'_i = C'[\widetilde{\Omega}]|_u$. Puesto que $C'[\ ] = CMR^\mu(\bar{s}')$, dicha Ω-occurrencia no es μ-reemplazante, esto es, $u \in \overline{O^\mu}(C'[\widetilde{\Omega}])$. Puesto que, por la Proposición 3.2.6, toda ocurrencia de un subtérmino distinto de Ω en $C'[\widetilde{\Omega}]$ es μ-reemplazante, y $sprefix_{C'[\widetilde{\Omega}]}(u) = sprefix_{C[\widetilde{\Omega}]}(u) = sprefix_{C''[\widetilde{\Omega}]}(u)$, de acuerdo con la Proposición 3.2.3, debe tenerse también $t''_i = \Omega$. Este razonamiento es completamente simétrico si lo iniciamos exigiendo que $t''_i = \Omega$. Por este motivo, podemos asegurar que para todo i, $1 \leq i \leq n$, $t'_i = t''_i = \Omega$, y, por tanto, $C[\ ] = C'[\ ] = C''[\ ]$, lo cual concluye la demostración. □

La propiedad de formas μ-normales únicas queda asegurada exigiendo la μ-confluencia del SRTE. Esta cuestión se abordará en el Capítulo 6.

Sin embargo, gracias a la Proposición 4.3.11, tal como queda demostrado en capítulos posteriores, no va a ser necesario exigir la μ-confluencia para llevar a cabo, mediante el empleo de la *rsc*, cierto tipo de cómputos útiles en programación.

4.4 Trabajos relacionados

La mayoría de los autores que emplean restricciones de reemplazamiento sintácticas para definir algún tipo de restricción de la reescritura asumen que o bien el programador va a responsabilizarse de establecer las restricciones [KW95, Mar90] o que serán establecidas a partir de algún tipo de análisis de *strictness* [HW87, KW95]. Ya discutimos en el capítulo de introducción y en la Sección 4.2 los inconvenientes derivados de esta forma de proceder. En [FGJM85] se hace hincapié en que la definición de estrategias locales para los símbolos de un programa OBJ2 no tiene porqué dejarse a la discreción del usuario y propone el empleo de 'heurísticos' para acometer esta tarea. El heurístico propuesto como ejemplo para definir la estrategia local asociada a la función + definida de la manera usual (ver el Ejemplo 3.3.3) anticipa la discusión realizada en la Sección 4.1 que ha llevado a la definición de la función de reemplazamiento canónica, $\mu^{com}_{\mathcal{R}}$, para un SRTE $\mathcal{R}$. No obstante, el resultado que se obtendría al generalizar el ejemplo de heurístico propuesto en [FGJM85], empleándolo como los

autores sugieren, no conduce a una definición de estrategias locales equivalente a las definibles a partir de $\mu^{com}_{\mathcal{R}}$.

En este capítulo hemos propuesto un método simple para establecer las restricciones de reemplazamiento adecuadas para un programa, mediante la función de reemplazamiento canónica $\mu^{com}_{\mathcal{R}}$. En secciones posteriores analizaremos qué comportamiento cabe esperar de los programas ejecutados mediante la *rsc* bajo una función de reemplazamiento μ. En ninguno de los trabajos considerados: [FGJM85, KW95, Mar90, OF97] se ofrece un análisis formal de estas cuestiones. Cuando μ es mayor o igual que $\mu^{com}_{\mathcal{R}}$, obtenemos un gran número de resultados acerca de la completitud de la *rsc* en la evaluación de términos a formas estables, valores, formas normales, ...; su capacidad para definir buenas estrategias sensibles al contexto (que incorporan una restricción adicional superpuesta a la inducida por la función de reemplazamiento); la eficiencia de los cómputos respecto a la noción de reducción necesaria, etc. Como hemos visto, el cálculo de la función $\mu^{com}_{\mathcal{R}}$ no requiere conocimiento alguno de las partes derechas de las reglas de reescritura y tampoco involucra el cálculo de puntos fijos.

Capítulo 5

Terminación

Dado que la *rsc* es una restricción de la reescritura, es razonable esperar que la imposición de las restricciones sensibles al contexto pueda evitar la existencia de secuencias de reducción infinitas, incluso en el supuesto de que el SRT no sea terminante. Sin embargo, se hace necesario disponer de métodos que permitan, en cada caso, demostrar esta conjetura razonable de forma concluyente. En este capítulo abordamos este problema.

Existe una vasta bibliografía sobre el tema de la terminación de sistemas de reescritura y encontramos también un amplio abanico de técnicas y procedimientos para demostrar de forma efectiva la terminación de un SRT [Art96, AG97, BL87, Der82, Der87, DH95, DM79, Fer95, GKM83, Gra96, MZ97, Zan94, Zan95]. Una forma de abordar el problema de la definición de métodos para demostrar la terminación de la *rsc* para un SRT (posiblemente con variables extra) y una función de reemplazamiento dada es tratar de adaptar al ámbito de la *rsc* los métodos existentes para la reescritura estándar (que no tienen en cuenta las restricciones de reemplazamiento expresadas por una función de reemplazamiento). Como quiera que dicho procedimiento exigiría la revisión pormenorizada de toda la literatura sobre el tema para adaptarla a la presencia de las restricciones de reemplazamiento sensibles al contexto, hemos adoptado un enfoque que permite emplear directamente los resultados estándar sin necesidad de adaptación: introducimos una trasformación que, dado un SRTE y una función de reemplazamiento, produce un SRTE cuya relación de reescritura asociada preserva las propiedades de terminación de la relación sensible al contexto del SRTE original.

La Sección 5.1 formaliza la metodología de análisis de la terminación de la *rsc* por comparación con la terminación de la relación no restringida asociada a algún tipo de SRT derivado a partir del SRTE original y la función de reemplazamiento correspondiente.

La Sección 5.2 presenta una transformación (llamada *contractiva*) sobre signaturas y términos que toma una función de reemplazamiento como parámetro y que emplearemos para realizar la demostración formal de terminación de la *rsc*. Más adelante, usaremos también esta transformación al estudiar las conexiones entre las restricciones de reemplazamiento sensibles al contexto y la secuencialidad fuerte de un SRTE.

La Sección 5.3 extiende la transformación contractiva a SRTEs. A partir de un SRTE $\mathcal{R}$ obtenemos un SRTE $\mathcal{R}^\mu$. Establecemos condiciones para asegurar que al aplicar la transformación sobre un SRTE se obtiene un SRT (sin variables extra).

La Sección 5.4 formaliza el empleo de la transformación contractiva para demostrar la μ-terminación de un SRTE.

La Sección 5.5 muestra cómo combinar los distintos métodos presentados para demostrar la μ-terminación de un SRTE, poniendo de manifiesto la necesidad de todos ellos pues ninguno de los métodos discutidos incluye totalmente a los demás.

La Sección 5.6 discute el interés de las técnicas presentadas en el contexto de los trabajos relacionados con la *rsc*.

Finalmente, la Sección 5.7 conecta la Ω-reducción empleando un SRTE $\mathcal{R}$ y la Ω-reducción empleando el SRTE transformado $\mathcal{R}^\mu$. Este estudio es necesario para el análisis que se efectuará en la Parte II acerca de la definición de estrategias sensibles al contexto μ-normalizantes.

5.1 Análisis de μ-terminación

El Ejemplo 1.1.2 sugería la posibilidad de eliminar secuencias de reducción infinitas mediante la imposición de restricciones de reemplazamiento. Podemos preguntarnos si es posible demostrar formalmente la inexistencia de secuencias de μ-reducción infinitas para un SRTE y una función de reemplazamiento μ dadas. Por ejemplo, para $\mathcal{R}$ y μ tal como se dieron en el Ejemplo 1.1.2 mencionado. En otras palabras, nos planteamos si es posible demostrar la μ-terminación de un programa.

Definición 5.1.1 (μ-terminación) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Decimos que $\mathcal{R}$ es μ-terminante si $\hookrightarrow_{\mathcal{R}(\mu)}$ es terminante.*

Con el fin de aprovechar las técnicas existentes para demostrar la terminación de los sistemas de reescritura, los métodos de demostración de la μ-terminación que presentamos están basados en establecer homomorfismos entre la relación de reescritura sensible al contexto y relaciones de reescritura asociadas a otros sistemas de reescritura vinculados al que se estudia. Emplearemos el siguiente resultado.

Teorema 5.1.2 *Sean A y B dos conjuntos y $S \subseteq A \times A$, $T \subseteq B \times B$ relaciones binarias sobre éstos. Sea $h : A \to B$ un homomorfismo entre las relaciones S y T^+. Si T es terminante, entonces S es terminante.*

DEMOSTRACIÓN. Por contradicción. Supongamos que S no es terminante. Entonces existe una secuencia infinita $a_1\ S\ a_2\ S\ a_3 \cdots$, donde $a_i \in A$ para $i \geq 1$. Puesto que h es un homomorfismo entre S y T^+, obtenemos una secuencia infinita $h(a_1)\ T^+\ h(a_2)\ T^+\ h(a_3) \cdots$ lo cual contradice la hipótesis de que T es terminante. □

Cualquier homomorfismo $h : A \to B$ entre relaciones S y T también será un homomorfismo entre S y T^+, dado que T^+ incluye T.

Un ejemplo simple de homomorfismo entre relaciones S y T sobre un mismo conjunto A y tales que $S \subseteq T$ es la función identidad id_A para el conjunto A. Empleando el Teorema 5.1.2, tenemos un primer resultado trivial.

Teorema 5.1.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu, \mu' \in M_\Sigma$ tales que $\mu \sqsubseteq \mu'$. Si $\mathcal{R}$ es μ'-terminante, entonces $\mathcal{R}$ es μ-terminante.*

DEMOSTRACIÓN. Tomamos $S = \hookrightarrow_{\mathcal{R}(\mu)}$, $T = \hookrightarrow_{\mathcal{R}(\mu')}$ y $h = id_{\mathcal{T}(\Sigma,V)}$ en el Teorema 5.1.2. La Proposición 3.3.7 asegura que $id_{\mathcal{T}(\Sigma,V)}$ es un homomorfismo entre $\hookrightarrow_{\mathcal{R}(\mu)}$ y $\hookrightarrow_{\mathcal{R}(\mu')}$, luego también entre $\hookrightarrow_{\mathcal{R}(\mu)}$ y $\hookrightarrow^+_{\mathcal{R}(\mu')}$. □

El siguiente resultado formaliza una primera aproximación a la idea de que podemos emplear directamente los métodos estándar para demostrar la terminación de SRTs en la demostración de la μ-terminación de un SRT.

Teorema 5.1.4 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$. Si $\mathcal{R}$ es terminante, entonces $\mathcal{R}$ es μ-terminante.*

DEMOSTRACIÓN. Hacemos $\mu' = \mu_\top$ en el Teorema 5.1.3. □

Observación 5.1.5 *No tiene sentido formular el Teorema 5.1.4 para SRTEs puesto que no existen SRTEs terminantes. Esto es así porque la presencia de una regla $l \to r$ con variables extra implica la existencia de una derivación infinita*

$$l \to r[l]_u \to r[r[l]_u]_u \to \cdots$$

si $r|_u \in V$ es una variable extra *de la regla.*

Aunque la μ-terminación del SRT presentado en el Ejemplo 3.3.3 (que es terminante) puede demostrarse con facilidad empleando el Teorema 5.1.4, dicho teorema no es útil para demostrar la μ-terminación de SRTEs. Tampoco es útil para demostrar la μ-terminación del SRT $\mathcal{R}$ del Ejemplo 1.1.2 con respecto a la función de reemplazamiento

μ en ese ejemplo. Esto se debe a que $\mathcal{R}$ no es terminante. Sin embargo, no es difícil comprobar que la restricción $\mu(::) = \emptyset$ introduce un impedimento esencial para la construcción de secuencias de μ-reducción infinitas. Esto es así porque esta restricción rompe el carácter no terminante que introduce en el SRT la regla de reescritura

$$\texttt{from(x)} \rightarrow \texttt{x::from(s(x))}$$

que define la función `from`.

5.2 La transformación contractiva

Básicamente, la introducción de restricciones de reemplazamiento mejora el comportamiento de terminación de un SRTE porque evita que las secuencias de reducción se perpetúen hasta el infinito al inhibir la capacidad de algunos subtérminos (los no reemplazantes) de iniciar estas secuencias. Por un lado, la introducción de restricciones de reemplazamiento permite hablar de (μ-)terminación de un SRTE, algo que no tiene sentido en reescritura no restringida (recordemos la Observación 5.1.5).

Ejemplo 5.2.1 *Sea $\mathcal{R}$ definido como sigue:*

$$\texttt{f(x)} \rightarrow \texttt{g(y)}$$

Se trata de un SRT con variables extra, un SRTE. Si imponemos $\mu(\texttt{f}) = \{1\}$ y $\mu(\texttt{g}) = \emptyset$, una derivación infinita como

$$\texttt{f(x)} \rightarrow \texttt{g(f(x))} \rightarrow \texttt{g(g(f(x)))} \rightarrow \cdots$$

no puede producirse porque tras el primer paso de μ-reducción se obtiene una forma μ-normal `g(f(x))`.

Volviendo al caso planteado al final de la sección anterior, empleando una función de reemplazamiento μ que satisfaga $\mu(::) = \emptyset$ se corta una secuencia infinita como

$$\underline{\texttt{from(x)}} \rightarrow \texttt{x::}\underline{\texttt{from(s(x))}} \rightarrow \texttt{x::s(x)::}\underline{\texttt{from(s(s(x)))}} \rightarrow \cdots$$

en el segundo paso de reducción. Esto es así porque impedimos la generación de dicha secuencia por el término `x::from(s(x))`, parte derecha de la regla de reescritura

$$\texttt{from(x)} \rightarrow \texttt{x::from(s(x))}$$

Para dar un tratamiento formal a este fenómeno, observemos que, si eliminamos por completo el subtérmino no reemplazante `from(s(x))` en la parte derecha de la regla anterior, sustituyéndolo por una nueva constante $\Box$, esto puede expresar el hecho de que no vamos a necesitar considerar las posibles reducciones iniciadas por el subtérmino sustituido. De esta forma, podemos pensar que la regla de reescritura, en cuanto a la *rsc* se refiere, se comporta como si tuviera la forma

$$\texttt{from(x)} \rightarrow \texttt{x::}\Box$$

Es sencillo ver que dicha regla no introduce secuencias de reducción infinitas.

De hecho, podemos ir más allá y *eliminar por completo* aquellos subtérminos situados sobre ocurrencias no reemplazantes de un término dado. Esto significa que la aridad de los símbolos de función quedará modificada lo cual implica trabajar con símbolos de una signatura diferente: la signatura μ-contracta.

Definición 5.2.2 (Signatura μ-contracta) *Sea $\Sigma = \{f, g, \ldots\}$ una signatura y $\mu \in M_\Sigma$. La signatura μ-contracta $\Sigma^\mu = \{f_\mu, g_\mu, \ldots\}$ se encuentra en correspondencia biunívoca con Σ y sus símbolos verifican: $ar_{\Sigma^\mu}(f_\mu) = |\mu(f)|$.*

Por tanto, un símbolo f_μ de la signatura μ-contracta Σ^μ admite tantos argumentos como argumentos reemplazantes $|\mu(f)|$ tiene el símbolo correspondiente $f \in \Sigma$. Este es el efecto que deseábamos obtener en cuanto a los símbolos de la signatura se refiere. En cuanto a los términos, podemos asociar a cada término $t \in \mathcal{T}(\Sigma, V)$ un término en $\mathcal{T}(\Sigma^\mu, V)$ donde hayan desaparecido por completo los subtérminos no reemplazantes. Esto se formaliza mediante una función de *μ-contracción* de términos.

Definición 5.2.3 (Función de μ-contracción) *Sea Σ una signatura y $\mu \in M_\Sigma$. La* función de μ-contracción $\tau_\mu : \mathcal{T}(\Sigma, V) \to \mathcal{T}(\Sigma^\mu, V)$ *se define como sigue:*

$$\tau_\mu(t) = \begin{cases} x & si\ t = x \in V \\ f_\mu(\tau_\mu(\tilde{t}_{\mu(f)})) & si\ t = f(\tilde{t}) \end{cases}$$

La definición inductiva de τ_μ establece que el cálculo del término μ-contraído a partir de un término $t = f(\tilde{t})$ procede cambiando el símbolo f por f_μ y situando bajo el nuevo símbolo la μ-contracción de los argumentos *μ-reemplazantes* de f conservando la disposición relativa que éstos tenían en el término original. Los argumentos no μ-reemplazantes de f se desechan.

Ejemplo 5.2.4 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 1.1.2 y la función de reemplazamiento μ dada en el ejemplo. Entonces,*

- *El término $t =$* `first(s(x),y::z)` *se contrae a $\tau_\mu(t) =$* `first`$_\mu$`(s`$_\mu$`,::`$_\mu$`)`.
- *El término $t' =$* `y::first(x,z)` *se contrae a $\tau_\mu(t') =$* `::`$_\mu$.

Escribimos $\equiv_{\tau_\mu}$ para denotar la equivalencia inducida por la función τ_μ sobre el conjunto $\mathcal{T}(\Sigma, V)$: $t \equiv_{\tau_\mu} s$ si y sólo si $\tau_\mu(t) = \tau_\mu(s)$. Distintos miembros de una clase de equivalencia comparten el mismo contexto reemplazante máximo,

Proposición 5.2.5 *Para todo $t, s \in \mathcal{T}(\Sigma, V)$ y $\mu \in M_\Sigma$, $t \equiv_{\tau_\mu} s$ implica que $t = C[t_1, \ldots, t_n]$ y $s = C[s_1, \ldots, s_n]$, donde $C[\,] = CMR^\mu(t) = CMR^\mu(s)$.*

También es interesante advertir que, si $t \equiv_{\tau_\mu} s$, y $C[\,] = CMR^\mu(t) = CMR^\mu(s)$, entonces $\tau_\mu(s) = \tau_\mu(t) = \tau_\mu(C[\,])$. Por tanto, cada contexto reemplazante máximo puede verse como el representante de una $\equiv_{\tau_\mu}$-clase.

A continuación demostramos que la función τ_μ es sobreyectiva.

Proposición 5.2.6 *Sea Σ una signatura y $\mu \in M_\Sigma$. La función de μ-contracción $\tau_\mu : \mathcal{T}(\Sigma, V) \to \mathcal{T}(\Sigma^\mu, V)$ es sobreyectiva.*

DEMOSTRACIÓN. Hay que demostrar que, para todo $s \in \mathcal{T}(\Sigma^\mu, V)$, existe un $t \in \mathcal{T}(\Sigma, V)$ tal que $s = \tau_\mu(t)$. Lo haremos por inducción en la estructura de s. Si $s \in \Sigma^\mu \cup V$, distinguimos dos casos: si $s = x \in V$, definimos $t = x$. Si $s = f_\mu \in \Sigma^\mu$, sea $f \in \Sigma$ el símbolo correspondiente a f_μ en Σ^μ. Distinguimos dos casos:

1. $ar_\Sigma(f) = 0$. Hacemos $t = f$.
2. $ar_\Sigma(f) > 0$. Dado que $ar_{\Sigma^\mu}(f_\mu) = 0$, debe tenerse $\mu(f) = \emptyset$. Entonces, podemos escoger un $t \in \mathcal{T}(\Sigma, V)$ cualquiera que verifique $root(t) = f$.

Si $s = f_\mu(\tilde{s})$, con $\tilde{s} \in \mathcal{T}(\Sigma^\mu, V)^{ar_{\Sigma^\mu}(f_\mu)}$, por la hipótesis de inducción existe una tupla $\widetilde{t'} \in \mathcal{T}(\Sigma, V)^{ar_{\Sigma^\mu}(f_\mu)}$ tal que $\tilde{s} = \tau_\mu(\widetilde{t'})$. Definimos $t = f(\tilde{t})$, para alguna tupla $\tilde{t} \in \mathcal{T}(\Sigma, V)^{ar_\Sigma(f)}$ tal que $\tilde{t}_{\mu(f)} = \widetilde{t'}$. □

Esta proposición nos autoriza a referirnos a cualquier término $s \in \mathcal{T}(\Sigma^\mu, V)$ como la imagen por τ_μ de algún término $t \in \mathcal{T}(\Sigma, V)$. Por tanto, a menudo escribiremos $\tau_\mu(t)$ en vez de s asumiendo que $s = \tau_\mu(t)$. Esta proposición también significa que τ_μ es una biyección entre $\mathcal{T}(\Sigma, V)/\equiv_{\tau_\mu}$ y $\mathcal{T}(\Sigma^\mu, V)$.

A partir de la definición de τ_μ, es claro que los subtérminos que ocupan las ocurrencias no reemplazantes se pierden. Es posible relacionar el conjunto de ocurrencias reemplazantes de un término t con el conjunto de ocurrencias del término contracto $\tau_\mu(t)$ mediante una función biyectiva $\nu_{\mu,t} : O^\mu(t) \to O(\tau_\mu(t))$.

Definición 5.2.7 *Sea Σ una signatura y $\mu \in M_\Sigma$. Sea $t \in \mathcal{T}(\Sigma, V)$. La función $\nu_{\mu,t} : O^\mu(t) \to O(\tau_\mu(t))$ se define como sigue:*

$$\nu_{\mu,t}(u) = \begin{cases} \epsilon & \text{si } u = \epsilon \\ |\mathbb{N}_i^+ \cap \mu(f)|.\nu_{\mu,t_i}(u') & \text{si } t = f(\tilde{t}) \text{ y } u = i.u' \end{cases}$$

Escribiremos simplemente ν_t en lugar de $\nu_{\mu,t}$ cuando la función de reemplazamiento μ se sobreentienda en el contexto de lo que se expone.

A continuación, establecemos cómo se pueden emplear las funciones τ_μ y ν_t para determinar la estructura de los términos t contraídos por τ_μ.

Proposición 5.2.8 (τ_μ preserva los subtérminos reemplazantes) *Sea Σ una signatura y $\mu \in M_\Sigma$. Si $t \in \mathcal{T}(\Sigma, V)$ y $u \in O^\mu(t)$, entonces $\tau_\mu(t|_u) = \tau_\mu(t)|_{\nu_t(u)}$.*

DEMOSTRACIÓN. Por inducción en la longitud de u. Si $u = \epsilon$, $\tau_\mu(t|_u) = \tau_\mu(t|_\epsilon) = \tau_\mu(t) = \tau_\mu(t)|_\epsilon = \tau_\mu(t)|_{\nu_t(\epsilon)} = \tau_\mu(t)|_{\nu_t(u)}$. Si $u = i.u'$, sea $t = f(\tilde{t})$. Puesto que $t|_u = t_i|_{u'}$, tenemos que $\tau_\mu(t|_u) = \tau_\mu(t_i|_{u'})$. Por la hipótesis de inducción, $\tau_\mu(t_i|_{u'}) = \tau_\mu(t_i)|_{\nu_{t_i}(u')}$. Por otro lado, $\tau_\mu(t) = \tau_\mu(f(\tilde{t})) = f_\mu(\tau_\mu(\tilde{t}_{\mu(f)}))$. Dado que, de acuerdo

con la definición de τ_μ, el i-ésimo argumento de f en $f(\tilde{t})$ se convierte (tras ser μ-contraído) en el $|\mathbb{N}_i^+ \cap \mu(f)|$-ésimo argumento de $f_\mu(\tau_\mu(\tilde{t}_{\mu(f)}))$, si $\nu_{t_i}(u') \in O(\tau_\mu(t_i))$, también se tiene que $|\mathbb{N}_i^+ \cap \mu(f)|.\nu_{t_i}(u') \in O(\tau_\mu(t))$. De acuerdo con la Definición 5.2.7, podemos escribir $\nu_t(u) \in O(\tau_\mu(t))$ y dado que $\tau_\mu(t_i)|_{\nu_{t_i}(u')} = \tau_\mu(t)|_{\nu_t(u)}$ se sigue la conclusión. □

La siguiente propiedad será utilizada más adelante.

Proposición 5.2.9 *Sean $t, s \in \mathcal{T}(\Sigma, V)$, $\mu \in M_\Sigma$ y $u \in O^\mu(t) \cap O^\mu(s)$. Si $sprefix_t(u) = sprefix_s(u)$, entonces $\nu_t(u) = \nu_s(u)$.*

DEMOSTRACIÓN. Por inducción sobre la estructura de la ocurrencia u. Si $u = \epsilon$, es inmediato. Si $u = i.u'$, entonces, dado que $sprefix_t(u) = sprefix_s(u)$, tenemos que $t = f(\tilde{t})$, $s = f(\tilde{s})$ y $sprefix_{t_i}(u') = sprefix_{s_i}(u')$. Por la Proposición 3.2.1, $u' \in O^\mu(t_i)$ y $u' \in O^\mu(s_i)$, es decir, $u' \in O^\mu(t_i) \cap O^\mu(s_i)$. Por la hipótesis de inducción, $\nu_{t_i}(u') = \nu_{s_i}(u')$. Por definición de ν, $\nu_t(u) = |\mathbb{N}_i^+ \cap \mu(f)|.\nu_{t_i}(u') = |\mathbb{N}_i^+ \cap \mu(f)|.\nu_{s_i}(u') = \nu_s(u)$ lo cual concluye la demostración. □

5.3 Contracción de sistemas de reescritura

La trasformación sobre términos presentada en la sección anterior se extiende fácilmente a una transformación sobre sistemas de reescritura, sin más que aplicarla a los términos que constituyen las reglas del sistema.

Definición 5.3.1 (SRTE μ-contracto) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Definimos la μ-contracción $\mathcal{R}^\mu = (\Sigma^\mu, R^\mu)$ de $\mathcal{R}$ como sigue:*

- *Σ^μ es la signatura μ-contracta (Definición 5.2.2).*
- *$R^\mu = \{\tau_\mu(l) \to \tau_\mu(r) \mid l \to r \in R\}$.*

La función sobreyectiva $\rho_\mu : R \to R^\mu$ relaciona, de la forma obvia, las reglas de reescritura de $\mathcal{R}$ y $\mathcal{R}^\mu$.

Es posible que $\mathcal{R}^\mu$ no sea un SRT, incluso cuando $\mathcal{R}$ es un SRT.

Ejemplo 5.3.2 *Consideremos el SRT $\mathcal{R} = (\Sigma, R)$ del Ejemplo 3.3.3 y la función de reemplazamiento μ dada en el ejemplo. El conjunto R^μ obtenido consta de las siguientes 'reglas':*

$$\begin{array}{llll} \mathtt{if}_\mu(\mathtt{true}_\mu) & \to \mathtt{x} & \mathtt{and}_\mu(\mathtt{true}_\mu) & \to \mathtt{x} \\ \mathtt{if}_\mu(\mathtt{false}_\mu) & \to \mathtt{y} & \mathtt{and}_\mu(\mathtt{false}_\mu) & \to \mathtt{false}_\mu \\ +_\mu(\mathtt{0}_\mu) & \to \mathtt{x} & & \\ +_\mu(\mathtt{s}_\mu) & \to \mathtt{s}_\mu & & \end{array}$$

que no constituyen un SRT, puesto que varias reglas poseen variables extra.

Puesto que sólo tiene sentido preguntarse por la terminación de un SRT (recordemos la Observación 5.1.5), estudiamos las condiciones que aseguran que la contracción de un SRTE es un SRT. Para asegurar que la aplicación de τ_μ a una regla $l \to r$ produce una regla de reescritura $\tau_\mu(l) \to \tau_\mu(r)$ sin variables extra, bastaría con exigir que cada variable reemplazante en r también lo sea en l, es decir, $Var^\mu(r) \subseteq Var^\mu(l)$. De esta manera, puesto que τ_μ preserva los subtérminos reemplazantes (Proposición 5.2.8), cualquier variable reemplazante en r seguirá presente tanto en $\tau_\mu(r)$ como en $\tau_\mu(l)$ y cumplirá la condición $Var(\tau_\mu(r)) \subseteq Var(\tau_\mu(l))$ de ausencia de variables extra en la regla transformada.

Definición 5.3.3 (SRTE μ-conservativo) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Decimos que el SRTE $\mathcal{R}$ es* μ-conservativo *si, para toda $l \to r \in R$, $Var^\mu(r) \subseteq Var^\mu(l)$.*

Observación 5.3.4 *Si un SRTE no es colapsante (*collapsing*, ver [KKSV95]), es decir, no contiene reglas del tipo $l \to x$ donde $x \in Var(l)$ es una variable, siempre es posible definir una función de reemplazamiento μ que haga que el SRTE sea μ-conservativo: Si imponemos $\mu(root(r)) = \emptyset$ para toda regla $l \to r$ del SRTE, forzamos que r no contenga ninguna variable reemplazante. Por tanto, $Var^\mu(r) = \emptyset$ y, trivialmente, $Var^\mu(r) \subseteq Var^\mu(l)$ para toda regla.*

La siguiente proposición establece que, dado un SRTE $\mathcal{R}$ y una función de reemplazamiento μ, el hecho de que $\mathcal{R}$ sea μ-conservativo es una condición suficiente para que la μ-contracción $\mathcal{R}^\mu$ sea un SRT.

Proposición 5.3.5 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Si $\mathcal{R}$ es μ-conservativo, entonces $\mathcal{R}^\mu$ es un SRT.*

DEMOSTRACIÓN. Sea $\mathcal{R}^\mu = (\Sigma^\mu, R^\mu)$. Para cada regla $\tau_\mu(l) \to \tau_\mu(r) \in R^\mu$, verificamos que $\tau_\mu(l) \notin V$ y $Var(\tau_\mu(r)) \subseteq Var(\tau_\mu(l))$. A partir de la Definición 5.2.3, la primera condición es inmediata si asumimos que $\mathcal{R}$ es un SRTE. Para comprobar la segunda condición, consideremos $x \in Var(\tau_\mu(r))$. Puesto que $x \in Var(\tau_\mu(r))$, y dado que ν_r es biyectiva, por la Proposición 5.2.8 debe tenerse $x \in Var^\mu(r)$. Puesto que $\mathcal{R}$ es conservativo, $x \in Var^\mu(l)$ y, por la Proposición 5.2.8, $x \in Var(\tau_\mu(l))$. Por tanto, $Var(\tau_\mu(r)) \subseteq Var(\tau_\mu(l))$. □

A continuación nos ocupamos brevemente de estudiar la preservación, por parte de la transformación de μ-contracción de SRTEs, de determinadas características sintácticas de los SRTEs.

Claramente, la transformación τ_μ siempre preserva la linealidad por la izquierda de los SRTEs.

Teorema 5.3.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y $\mu \in M_\Sigma$. Entonces, $\mathcal{R}^\mu$ es lineal por la izquierda.*

Este resultado no se verifica en sentido contrario.

Ejemplo 5.3.7 *Consideremos el SRT $\mathcal{R}$:*

$\mathtt{f(x,x)} \rightarrow \mathtt{a}$

que no es lineal por la izquierda y la función de reemplazamiento μ dada por $\mu(\mathtt{f}) = \{1\}$. El SRT $\mathcal{R}^\mu$

$\mathtt{f}_\mu\mathtt{(x)} \rightarrow \mathtt{a}_\mu$

sí es lineal por la izquierda.

Demostramos que, bajo ciertas condiciones, τ_μ establece una correspondencia biunívoca entre los pares críticos de un SRTE y su μ-contracción. Para ello necesitamos un resultado auxiliar.

Proposición 5.3.8 *Sea Σ una signatura y $\mu \in M_\Sigma$. Sea $t \in \mathcal{T}(\Sigma, V)$ y $\sigma \in Subst(\Sigma, V)$. Sea $\sigma_\mu \in Subst(\Sigma^\mu, V)$ definida como sigue: $\sigma_\mu(x) = \tau_\mu(\sigma(x))$ para toda $x \in V$. Entonces, $\tau_\mu(\sigma(t)) = \sigma_\mu(\tau_\mu(t))$.*

DEMOSTRACIÓN. Por inducción estructural. Si $t = x \in V$, dado que $\tau_\mu(x) = x$ y por definición de σ_μ, $\sigma_\mu(\tau_\mu(t)) = \sigma_\mu(\tau_\mu(x)) = \sigma_\mu(x) = \tau_\mu(\sigma(x)) = \tau_\mu(\sigma(t))$. Si $t = f(\tilde{t})$, por definición de τ_μ y por la hipótesis de inducción, $\tau_\mu(\sigma(t)) = \tau_\mu(\sigma(f(\tilde{t}))) = \tau_\mu(f(\sigma(\tilde{t}))) = f_\mu(\tau_\mu(\sigma(\tilde{t}_{\mu(f)}))) = f_\mu(\sigma_\mu(\tau_\mu(\tilde{t}_{\mu(f)}))) = \sigma_\mu(f_\mu(\tau_\mu(\tilde{t}_{\mu(f)}))) = \sigma_\mu(\tau_\mu(t))$. Luego, $\tau_\mu(\sigma(t)) = \sigma_\mu(\tau_\mu(t))$. □

Teorema 5.3.9 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Entonces, $\langle \sigma(l)[\sigma(r')]_u, \sigma(r)\rangle$ es un par crítico de $\mathcal{R}$ si y sólo si $\langle \tau_\mu(\sigma(l)[\sigma(r')]_u), \tau_\mu(\sigma(r))\rangle$ es un par crítico de $\mathcal{R}^\mu$.*

DEMOSTRACIÓN. Si $\langle \sigma(l)[\sigma(r')]_u, \sigma(r)\rangle$ es un par crítico de $\mathcal{R}$, entonces $u \in O_\Sigma(l)$ es tal que l' y $l|_u$ unifican con *umg* σ, es decir, $\sigma(l') = \sigma(l|_u)$. Dado que $u \in O_\Sigma(l)$ y $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, tenemos que $u \in O^\mu(l)$ y, por tanto, $\nu_l(u) \in O_{\Sigma^\mu}(\tau_\mu(l))$ (y viceversa). Sea σ_μ definida como sigue: $\sigma_\mu(x) = \tau_\mu(\sigma(x))$ para toda $x \in V$. Por la Proposición 5.3.8, $\tau_\mu(\sigma(l')) = \sigma_\mu(\tau_\mu(l'))$ y $\tau_\mu(\sigma(l|_u)) = \sigma_\mu(\tau_\mu(l|_u))$. Dado que $\sigma(l') = \sigma(l|_u)$, también tenemos que $\sigma_\mu(\tau_\mu(l|_u)) = \sigma_\mu(\tau_\mu(l'))$, es decir, σ_μ es un unificador de las lhs's $\tau_\mu(l')$ y $\tau_\mu(l|_u)$ y, por tanto, $\langle \sigma_\mu(\tau_\mu(l))[\sigma_\mu(\tau_\mu(r'))]_{\nu_l(u)}, \sigma_\mu(\tau_\mu(r))\rangle$ es un par crítico de $\mathcal{R}^\mu$. Viceversa, si $\sigma_\mu(\tau_\mu(l|_u)) = \sigma_\mu(\tau_\mu(l'))$, tenemos $\sigma(l|_u) = \sigma(l')$ para una σ que verifique $\sigma_\mu(x) = \tau_\mu(\sigma(x))$ para toda $x \in V$ puesto que, siendo $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, los únicos subtérminos no reemplazantes en las lhs's de las reglas de reescritura son variables y los enlaces que quedan libres para las variables no reemplazantes de $l|_u$ o l' siempre

pueden hacerse coincidir. Aplicando las Proposiciones 5.2.8 y 5.3.8, concluimos que $\langle \sigma_\mu(\tau_\mu(l))[\sigma_\mu(\tau_\mu(r'))]_{\nu_l(u)}, \sigma_\mu(\tau_\mu(r))\rangle = \langle \tau_\mu(\sigma(l)[\sigma(r')]_u), \tau_\mu(\sigma(r))\rangle$. □

Del Teorema 5.3.6 y el Teorema 5.3.9 extraemos una consecuencia inmediata.

Teorema 5.3.10 *Un SRTE $\mathcal{R} = (\Sigma, R)$ lineal por la izquierda es ortogonal si y sólo si $\mathcal{R}^\mu$ es ortogonal para $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$.*

Además, la transformación τ_μ preserva la ortogonalidad débil (y la cuasi ortogonalidad) de los SRTEs.

Teorema 5.3.11 *Si $\mathcal{R} = (\Sigma, R)$ es un SRTE débilmente (cuasi) ortogonal y $\mu \in M_\Sigma$ es tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, entonces $\mathcal{R}^\mu$ es débilmente (cuasi) ortogonal.*

DEMOSTRACIÓN. El Teorema 5.3.6 asegura que la linealidad por la izquierda de $\mathcal{R}$ queda preservada en $\mathcal{R}^\mu$. Por el Teorema 5.3.9, para cada un par crítico $\langle \sigma(l)[\sigma(r')]_u, \sigma(r)\rangle$ de $\mathcal{R}$, tenemos un par crítico $\langle \tau_\mu(\sigma(l)[\sigma(r')]_u), \tau_\mu(\sigma(r))\rangle$ de $\mathcal{R}^\mu$ y viceversa. Si $\mathcal{R}$ es débilmente ortogonal, entonces $\sigma(l)[\sigma(r')]_u = \sigma(r)$, es decir, $\tau_\mu(\sigma(l)[\sigma(r')]_u) = \tau_\mu(\sigma(r))$, luego $\mathcal{R}^\mu$ también es débilmente ortogonal.

Si, en particular, $u = \epsilon$, es decir, $\mathcal{R}$, es cuasi ortogonal, entonces $\nu_l(u) = \nu_l(\epsilon) = \epsilon$ y $\mathcal{R}^\mu$ también es cuasi ortogonal. □

Debido a que τ_μ no es, en general, una función biyectiva, el Teorema 5.3.11 no se puede establecer como una equivalencia.

5.4 μ-terminación y μ-contracción

Nuestra intención es utilizar $\mathcal{R}^\mu$ para demostrar que $\mathcal{R}$ es μ-terminante. Nos serviremos del Teorema 5.1.2 demostrando previamente que τ_μ es un homomorfismo entre la relación $\hookrightarrow_{\mathcal{R}(\mu)}$ y la relación de reescritura $\rightarrow_{\mathcal{R}^\mu}$ inducida por $\mathcal{R}^\mu$.

La siguiente proposición caracteriza τ_μ como un homomorfismo entre la relación de μ-reescritura en $\mathcal{R}$ y la relación de reescritura en $\mathcal{R}^\mu$.

Proposición 5.4.1 (μ-reescritura en $\mathcal{R}$ y reescritura en $\mathcal{R}^\mu$) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Sean $t, s \in \mathcal{T}(\Sigma, V)$. Si $t \overset{u}{\hookrightarrow} s$, entonces $\tau_\mu(t) \overset{\nu_t(u)}{\rightarrow} \tau_\mu(s)$.*

DEMOSTRACIÓN. Si $t \overset{u}{\hookrightarrow} s$, entonces $u \in O^\mu(t)$ y existen $l \rightarrow r \in R$ y $\sigma \in Subst(\Sigma, V)$ tales que $t|_u = \sigma(l)$ y $s = t[\sigma(r)]_u$. Sea $v = \nu_t(u)$ y sea $\sigma_\mu \in Subst(\Sigma^\mu, V)$ una sustitución tal que $\sigma_\mu(x) = \tau_\mu(\sigma(x))$ para toda $x \in V$. Por la Proposición 5.2.8 y la Proposición 5.3.8 tenemos que $\tau_\mu(t)|_v = \tau_\mu(t|_u) = \tau_\mu(\sigma(l)) = \sigma_\mu(\tau_\mu(l))$. Por la Definición 5.3.1, si $l \rightarrow r \in R$, entonces $\tau_\mu(l) \rightarrow \tau_\mu(r) \in R^\mu$. Por tanto, el término $\tau_\mu(t)$ puede ser reducido a la ocurrencia v empleando la regla $\tau_\mu(l) \rightarrow$

$\tau_\mu(r) \in R^\mu$ y el emparejamiento σ_μ. El término obtenido es $s' = \tau_\mu(t)[\sigma_\mu(\tau_\mu(r))]_v = \tau_\mu(t)[\tau_\mu(\sigma(r))]_v = \tau_\mu(t[\sigma(r)]_u) = \tau_\mu(s)$. Esto concluye la demostración. □

El siguiente ejemplo muestra que $\tau_\mu(t) \to_{\mathcal{R}^\mu} \tau_\mu(s)$ no implica que $t \hookrightarrow_{\mathcal{R}(\mu)} s$.

Ejemplo 5.4.2 *Consideremos el SRT* $\mathcal{R}$

$$\mathtt{f(a,b)} \to \mathtt{a}$$

y la función de reemplazamiento μ *tal que* $\mu(\mathtt{f}) = \{1\}$. *El SRT* $\mathcal{R}$ *es trivialmente conservativo puesto que se trata de un SRT básico, cuyas reglas no contienen variables. Su* μ*-contracción* $\mathcal{R}^\mu$ *es*

$$\mathtt{f}_\mu(\mathtt{a}_\mu) \to \mathtt{a}_\mu$$

Sea $t = \mathtt{f(a,a)}$ *y* $\tau_\mu(\mathtt{f(a,a)}) = \mathtt{f}_\mu(\mathtt{a}_\mu)$. *Obtenemos*

$$\tau_\mu(\mathtt{f(a,a)}) = \mathtt{f}_\mu(\mathtt{a}_\mu) \to_{\mathcal{R}^\mu} \mathtt{a}_\mu = \tau_\mu(\mathtt{a})$$

Sin embargo, $\mathtt{f(a,a)} \not\hookrightarrow_{\mathcal{R}(\mu)} \mathtt{a}$.

Ahora podemos establecer el resultado deseado.

Teorema 5.4.3 (μ-terminación de $\mathcal{R}$ y terminación de $\mathcal{R}^\mu$) *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRTE* μ*-conservativo respecto a* $\mu \in M_\Sigma$. *Si* $\mathcal{R}^\mu$ *es terminante, entonces* $\mathcal{R}$ *es* μ*-terminante.*

DEMOSTRACIÓN. La Proposición 5.4.1 demuestra que τ_μ es un homomorfismo entre $\hookrightarrow_{\mathcal{R}(\mu)}$ y $\to_{\mathcal{R}^\mu}$ (luego también entre $\hookrightarrow_{\mathcal{R}(\mu)}$ y $\to^+_{\mathcal{R}^\mu}$). Por tanto, hacemos $S = \hookrightarrow_{\mathcal{R}(\mu)}$, $T = \to_{\mathcal{R}^\mu}$ y $h = \tau_\mu$ en el Teorema 5.1.2. □

Observación 5.4.4 *Tampoco tiene sentido formular el Teorema 5.4.3 para SRTEs no* μ*-conservativos puesto que, en ese caso,* $\mathcal{R}^\mu$ *sería un SRTE y no existen SRTEs terminantes.*

Ejemplo 5.4.5 *Ahora es posible dar una prueba formal de* μ*-terminación del SRT* $\mathcal{R}$ *del Ejemplo 1.1.2 respecto a la función de reemplazamiento* μ *del ejemplo. El SRT* $\mathcal{R}^\mu$ *obtenido por* μ*-contracción de* $\mathcal{R}$ *es:*

$$\mathtt{from}_\mu \to \mathtt{::}_\mu \qquad \mathtt{first}_\mu(\mathtt{0}_\mu,\mathtt{x}) \to \mathtt{[]}_\mu$$
$$\mathtt{zero}_\mu \to \mathtt{::}_\mu \qquad \mathtt{first}_\mu(\mathtt{s}_\mu,\mathtt{::}_\mu) \to \mathtt{::}_\mu$$

que es claramente terminante. Ello se demuestra mediante un rpo *sin más que escoger la precedencia* $\mathtt{zero}_\mu > \mathtt{0}_\mu$, $\mathtt{from}_\mu > \mathtt{::}_\mu$, $\mathtt{first}_\mu > \mathtt{[]}_\mu$ *y* $\mathtt{first}_\mu > \mathtt{::}_\mu$.

Podemos considerar un ejemplo adicional que será utilizado más adelante.

Ejemplo 5.4.6 *Consideremos el SRT no terminante $\mathcal{R}$:*

$$\begin{array}{lll} \mathtt{f(g(a,x,y),f(b,z))} \to \mathtt{x} & & \mathtt{g(d,d,x)} \to \mathtt{h(x)} \\ \mathtt{f(g(x,a,y),f(c,z))} \to \mathtt{z} & & \mathtt{h(x)} \to \mathtt{e(h(x))} \end{array}$$

Si escogemos $\mu = \mu^{com}_{\mathcal{R}}$, obtenemos el siguiente SRT μ-contracto $\mathcal{R}^\mu$:

$$\begin{array}{lll} \mathtt{f}_\mu\mathtt{(g}_\mu\mathtt{(a}_\mu\mathtt{,x),f}_\mu\mathtt{(b}_\mu\mathtt{,z))} \to \mathtt{x} & & \mathtt{g}_\mu\mathtt{(d}_\mu\mathtt{,d}_\mu\mathtt{)} \to \mathtt{h}_\mu \\ \mathtt{f}_\mu\mathtt{(g}_\mu\mathtt{(x,a}_\mu\mathtt{),f}_\mu\mathtt{(c}_\mu\mathtt{,z))} \to \mathtt{z} & & \mathtt{h}_\mu \to \mathtt{e}_\mu \end{array}$$

cuya terminación es fácil de demostrar, pues era la regla que define la función `h` *que introducía el carácter no terminante a través de una recursión sobre* `h` *que ha quedado rota al imponer las restricciones de reemplazamiento.*

El Teorema 5.4.3 también puede emplearse para demostrar la μ-terminación de los SRTEs.

Ejemplo 5.4.7 *Sea el SRTE $\mathcal{R}$ del Ejemplo 5.2.1 y la función de reemplazamiento μ definida en el ejemplo. La μ-contracción, $\mathcal{R}^\mu$, de $\mathcal{R}$ es:*

$$\mathtt{f}_\mu\mathtt{(x)} \to \mathtt{g}_\mu$$

que es un SRT. La terminación de $\mathcal{R}^\mu$ puede demostrarse empleando los métodos estándar, ya que no contiene variables extra (puede utilizarse un rpo *basado en una precedencia $>$ que imponga $\mathtt{f}_\mu > \mathtt{g}_\mu$). Por tanto, $\mathcal{R}$ es μ-terminante.*

5.5 Demostración de la μ-terminación

Los Teoremas 5.1.3, 5.1.4 y 5.4.3 pueden emplearse para demostrar la μ-terminación de un SRTE. En esta sección presentamos ejemplos de que ninguno de los métodos formalizados mediante dichos teoremas engloba a los demás. Por tanto, pueden utilizarse de forma complementaria para demostrar la μ-terminación de los SRTEs. También veremos algunos casos donde los métodos presentados en esta sección fallan pero todavía es posible demostrar la μ-terminación de un SRT empleando el método transformacional ideado por Zantema [Zan97]. Sin embargo, existen SRTs cuya μ-terminación puede demostrarse empleando nuestros métodos, mientras que el de Zantema no es aplicable. Además, el método de Zantema no es aplicable a la demostración de μ-terminación de SRTEs.

Al considerar el SRT $\mathcal{R}$ y la función de reemplazamiento μ del Ejemplo 3.3.3 observamos que el Teorema 5.4.3 *no* es aplicable para demostrar la μ-terminación de $\mathcal{R}$ que, sin embargo, puede demostrarse empleando el Teorema 5.1.4. Esto se debe a que $\mathcal{R}$ no es μ-conservativo lo cual impide aplicar el Teorema 5.4.3 (ver el Ejemplo 5.3.2).

Por otro lado, el Ejemplo 5.4.5 muestra que el Teorema 5.4.3 puede aplicarse cuando el Teorema 5.1.4 falla.

Finalmente, a veces es útil emplear el Teorema 5.1.3 en combinación con los anteriores cuando ninguno de ellos es suficiente de forma independiente.

Ejemplo 5.5.1 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 4.2.3 y la función de reemplazamiento $\mu = \mu_{\mathcal{R}}^{com}$ dada en el ejemplo:*

$$\mu(\mathtt{s}) = \mu(\mathtt{::}) = \mu(\mathtt{from}) = \emptyset,\ \mu(\mathtt{if}) = \mu(\mathtt{and}) = \mu(\mathtt{+}) = \{1\}, \mu(\mathtt{first}) = \{1,2\}$$

Este SRT $\mathcal{R}$ no es terminante, debido a la regla para la función `from`*. Por tanto, el Teorema 5.1.4 no puede aplicarse. Por otro lado, $\mathcal{R}$ no es μ-conservativo y tampoco podemos aplicar el Teorema 5.4.3. No obstante, podemos relajar las restricciones impuestas por μ para obtener una función de reemplazamiento menos restrictiva μ' (es decir $\mu \sqsubseteq \mu'$) que permita demostrar que $\mathcal{R}$ es μ'-terminante. Entonces, por el Teorema 5.1.3 obtendríamos la demostración de μ-terminación de $\mathcal{R}$. Por ejemplo, tomando*

$$\mu'(\mathtt{s}) = \mu'(\mathtt{::}) = \mu'(\mathtt{from}) = \emptyset,$$

$$\mu'(\mathtt{and}) = \mu'(\mathtt{+}) = \mu'(\mathtt{first}) = \{1,2\} \text{ y } \mu'(\mathtt{if}) = \{1,2,3\}$$

podemos comprobar que $\mathcal{R}$ es μ'-conservativo. Obtenemos el SRT μ'-contracto $\mathcal{R}^{\mu'}$.

$$\begin{array}{llll}
\mathtt{if}_{\mu'}(\mathtt{true}_{\mu'},\mathtt{x},\mathtt{y}) & \to \mathtt{x} & \mathtt{and}_{\mu'}(\mathtt{true}_{\mu'},\mathtt{x}) & \to \mathtt{x} \\
\mathtt{if}_{\mu'}(\mathtt{false}_{\mu'},\mathtt{x},\mathtt{y}) & \to \mathtt{y} & \mathtt{and}_{\mu'}(\mathtt{false}_{\mu'},\mathtt{y}) & \to \mathtt{false}_{\mu'} \\
\mathtt{first}_{\mu'}(\mathtt{0}_{\mu'},\mathtt{x}) & \to []_{\mu'} & \mathtt{0}_{\mu'}\mathtt{+}_{\mu'}\mathtt{x} & \to \mathtt{x} \\
\mathtt{first}_{\mu'}(\mathtt{s}_{\mu'},\mathtt{::}_{\mu'}) & \to \mathtt{::}_{\mu'} & \mathtt{s}_{\mu'}\mathtt{+}_{\mu'}\mathtt{y} & \to \mathtt{s}_{\mu'} \\
\mathtt{from}_{\mu'} & \to \mathtt{::}_{\mu'} & &
\end{array}$$

cuya terminación es posible demostrar. Luego $\mathcal{R}$ es μ'-terminante y, puesto que $\mu \sqsubseteq \mu'$, también μ-terminante.

No obstante, hay casos en los que los resultados presentados aquí se revelan insuficientes para demostrar la μ-terminación de un SRT.

Ejemplo 5.5.2 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 3.4.3 y la función de reemplazamiento utilizada en el ejemplo:*

$$\mu(\mathtt{from}) = \mu(\mathtt{::}) = \mu(\mathtt{s}) = \{1\}, \mu(\mathtt{sel}) = \{1,2\}$$

que se limita a restringir el reemplazamiento del segundo argumento del constructor de listas '`::`*'. El SRT no es terminante. Tampoco es μ-conservativo. De hecho, su μ-contracción $\mathcal{R}^{\mu}$ es*

$$\begin{array}{lll}
\mathtt{sel}_{\mu}(\mathtt{0},\mathtt{::}_{\mu}(\mathtt{x})) & \to \mathtt{x} & \mathtt{from}_{\mu}(\mathtt{x}) \to \mathtt{::}_{\mu}(\mathtt{x}) \\
\mathtt{sel}_{\mu}(\mathtt{s}_{\mu}(\mathtt{x}),\mathtt{::}_{\mu}(\mathtt{y})) & \to \mathtt{sel}_{\mu}(\mathtt{x},\mathtt{z}) &
\end{array}$$

que no es un SRT debido a la presencia de la variable extra `z` *en la segunda regla para la función* `sel`*. Observemos que la única función de reemplazamiento que es estrictamente superior a* μ *es* $\mu_\top$*, lo que significa que no existe, a diferencia del procedimiento seguido en el Ejemplo 5.5.1, la posibilidad de encontrar una función de reemplazamiento* μ' *menos restrictiva (y distinta de* $\mu_\top$*) que haga el SRT* μ'*-conservativo y permita demostrar que* $\mathcal{R}$ *es* μ'*-terminante.*

Hans Zantema ha desarrollado un método para demostrar la μ-terminación de SRTs que también está basado en transformar el SRT original [Zan97]. Con el método de Zantema, en lugar de eliminar los subtérminos no reemplazantes, se realiza un *marcado* de las ocurrencias no reemplazantes[1]. A partir de un SRT $\mathcal{R}$ y una función de reemplazamiento μ, obtenemos un nuevo SRT $\Psi(\mathcal{R},\mu)$ (sin variables extra).

Ejemplo 5.5.3 [Zan97] *(Continuando el Ejemplo 5.5.2) A partir del SRT del Ejemplo 3.4.3 y de la función de reemplazamiento* μ *utilizada en él, obtenemos el siguiente SRT* $\Psi(\mathcal{R},\mu)$*:*

$$\begin{array}{lll}
\mathtt{sel(0,x::y)} & \to \mathtt{x} & \mathtt{from(x)} \to \mathtt{x::}\overline{\mathtt{from}}\mathtt{(s(x))} \\
\mathtt{sel(s(x),y::z)} & \to \mathtt{sel(x,a(z))} & \\
\mathtt{a(}\overline{\mathtt{from}}\mathtt{(x))} & \to \mathtt{from(x)} & \mathtt{from(x)} \to \overline{\mathtt{from}}\mathtt{(x)} \\
\mathtt{a(x)} & \to \mathtt{x} &
\end{array}$$

que, como se indica en [Zan97], puede probarse terminante utilizando un rpo *con la siguiente precedencia:* $\mathtt{sel} > \mathtt{a} > \mathtt{from} > \overline{\mathtt{from}}$, $\mathtt{from} > \mathtt{s}$ *y* $\mathtt{from} > \mathtt{::}$.

Zantema demuestra que la terminación de $\Psi(\mathcal{R},\mu)$ implica la μ-terminación de $\mathcal{R}$.

Teorema 5.5.4 [Zan97] *Un SRT* $\mathcal{R}$ *es* μ*-terminante si* $\Psi(\mathcal{R},\mu)$ *es terminante.*

El método de Zantema puede emplearse, pues, para demostrar la μ-terminación de SRTs no μ-conservativos. No obstante, tal como se hace notar en [Zan97], este método no subsume el método transformacional basado en la función τ_μ.

Ejemplo 5.5.5 *Consideremos el SRT* $\mathcal{R}$*:*

$$\mathtt{f(x)} \to \mathtt{g(h(f(x)))}$$

y la función de reemplazamiento μ *dada por* $\mu(\mathtt{f}) = \mu(\mathtt{h}) = \{1\}$ *y* $\mu(\mathtt{g}) = \emptyset$*.* $\mathcal{R}$ *es* μ*-terminante, puesto que es sencillo demostrar que su* μ*-contracción* $\mathcal{R}^\mu$*:*

$$\mathtt{f}_\mu\mathtt{(x)} \to \mathtt{g}_\mu$$

es terminante. Sin embargo, $\Psi(\mathcal{R},\mu)$*:*

[1] Estrictamente, no todos los símbolos situados sobre ocurrencias no reemplazantes en las reglas de un SRT resultan marcados en los SRT transformados mediante el método de Zantema. Ver el Ejemplo 5.5.3.

$$\begin{aligned}
\texttt{f(x)} &\rightarrow \texttt{g}(\overline{\texttt{h}}(\texttt{f(x)})) \\
\texttt{a}(\overline{\texttt{h}}(\texttt{x})) &\rightarrow \texttt{h(x)} \\
\texttt{h(x)} &\rightarrow \overline{\texttt{h}}(\texttt{x}) \\
\texttt{a(x)} &\rightarrow \texttt{x}
\end{aligned}$$

no es terminante debido a la primera regla.

Finalmente, la μ-terminación del SRTE del Ejemplo 5.2.1 no puede demostrarse empleando el método de Zantema (que sólo se aplica a SRTs), pero sí empleando el Teorema 5.4.3 (ver el Ejemplo 5.4.7).

5.6 Trabajos relacionados

El trabajo sobre terminación de la *rsc* realizado por Zantema no se limita a presentar un método de demostración de la μ-terminación de SRTs. También ha formalizado un tratamiento algebraico de la μ-terminación (de SRTs), identificando las propiedades que debe reunir un orden para ser utilizado en demostraciones de μ-terminación e introduciendo métodos interpretativos para demostrar ésta [Zan97].

Por otro lado, con relación a las propuestas de reescritura restringida descritas en la Sección 3.5 del Capítulo 3 no se contempla ninguna noción de terminación de un SRT respecto a la restricción de la reescritura propuesta en cada caso. Por tanto, tampoco se ha realizado para dichas restricciones un análisis similar al estudio sobre μ-terminación desarrollado en este capítulo. La única referencia lejana a este problema la encontramos en [KW95] donde se menciona el hecho intuitivamente claro de que, siendo una restricción de la reescritura, la reescritura perezosa presentada por los autores preserva la terminación de los SRT.

Empleando los resultados de este capítulo es posible demostrar formalmente si, en la terminología de [OF97], la estrategia E correspondiente a un determinado conjunto de estrategias locales asociadas a los símbolos del programa siempre obtendrá la forma E-normal de un término. Esto es así porque todo paso de reducción elemental efectuado por la estrategia E es un paso de reducción sensible al contexto y las formas E-normales coinciden con las μ-normales cuando la función de reemplazamiento μ considerada se obtiene, a partir de las estrategias locales asociadas a cada símbolo de función del programa, procediendo tal como se indicó en la Sección 3.5. En este caso, si el SRT puede demostrarse μ-terminante, cualquier secuencia de μ-reducción será finita y, en particular, lo serán las secuencias de reducción obtenidas mediante la estrategia E. Puesto que, hasta ahora, no han sido establecidos resultados formales acerca del carácter E-normalizante de la estrategia E, esta es una aportación original de las técnicas de demostración de μ-terminación al análisis de las propiedades

computacionales de dicha estrategia y, por ende, al análisis del comportamiento de los programas ejecutados mediante ésta en los sistemas OBJ y CafeOBJ.

5.7 Ω-reducción con $\mathcal{R}$ y $\mathcal{R}^\mu$

En esta sección estudiamos la conexión existente entre la $\rightarrow_\Omega$-reducción basada en un SRTE $\mathcal{R}$ (que, como siempre, denominaremos Ω-reducción) y las $\rightarrow_{\Omega_\mu}$-reducción basada en $\mathcal{R}^\mu$ (que denominaremos simplemente Ω_μ-reducciones, sobreentendiendo que el SRTE involucrado es $\mathcal{R}^\mu$). Estos resultados, que presentamos aquí por su carácter fundamental, serán utilizados en la Parte II (Sección 8.2) para relacionar la secuencialidad fuerte de un SRTE y su μ-contracción, y para definir buenas estrategias de reducción sensible al contexto.

Observación 5.7.1 *A partir de aquí, al considerar un Ω-término $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$, asumiremos que los símbolos Ω de t, siendo extraños a la signatura Σ, se consideran como* variables *a la hora de calcular la función de reemplazamiento μ_t. Por ejemplo, si $t = f(\Omega, \Omega)$, entonces $\mu_t(f) = \emptyset$ en vez de $\mu_t(f) = \{1, 2\}$, como resultaría al considerar Ω como un símbolo constante perteneciente a la signatura. Esto es equivalente a pensar que $t \in \mathcal{T}(\Sigma, V \cup \{\Omega\})$ en lugar de que $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$.*

El siguiente resultado establece que τ_μ es monótona respecto al orden $\leq$ entre Ω-términos y que, además, preserva los términos compatibles.

Proposición 5.7.2 *Sea Σ una signatura, $t, s \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ y $\mu \in M_\Sigma$.*

1. *Si $t \leq s$, entonces $\tau_\mu(t) \leq \tau_\mu(s)$.*
2. *Si $t \uparrow s$, entonces $\tau_\mu(t) \uparrow \tau_\mu(s)$.*

DEMOSTRACIÓN. $(t \leq s \Rightarrow \tau_\mu(t) \leq \tau_\mu(s))$: Por inducción estructural sobre t. Si $t \in \Sigma \cup \{\Omega\} \cup V$, entonces $t \leq s$ implica que $t = \Omega$ o bien $t = s$. En el primer caso, $\tau_\mu(t) = \tau_\mu(\Omega) = \Omega_\mu \leq \tau_\mu(s)$ y en el segundo, $\tau_\mu(t) = \tau_\mu(s)$, esto es, en ambos casos, $\tau_\mu(t) \leq \tau_\mu(s)$.

Si $t = f(\tilde{t})$, entonces $t \leq s$ implica que $s = f(\tilde{s})$ y $t_i \leq s_i$, para $1 \leq i \leq ar(f)$. Por la hipótesis de inducción, $\tau_\mu(t_i) \leq \tau_\mu(s_i)$ para $1 \leq i \leq ar(f)$. En particular, $\tau_\mu(t_i) \leq \tau_\mu(s_i)$, para todo $i \in \mu(f)$. Por tanto, $\tau_\mu(f(\tilde{t})) = f_\mu(\tau_\mu(\tilde{t}_{\mu(f)})) \leq f_\mu(\tau_\mu(\tilde{s}_{\mu(f)})) = \tau_\mu(f(\tilde{s}))$.

$(t \uparrow s \Rightarrow \tau_\mu(t) \uparrow \tau_\mu(s))$: es inmediato, empleando el resultado anterior. □

Adviértase que, en general, no es posible extender la Proposición 5.7.2 al orden estricto.

Ejemplo 5.7.3 *Sea* $t = f(\Omega, \Omega) < f(\Omega, a) = s$. *Si imponemos* $\mu(f) = \{1\}$, *tenemos que* $\tau_\mu(t) = f_\mu(\Omega_\mu) = \tau_\mu(s)$, *en vez de* $\tau_\mu(t) < \tau_\mu(s)$.

Para obtener el resultado análogo a la Proposición 5.7.2 para el orden estricto entre Ω-términos, necesitamos imponer condiciones adicionales.

Proposición 5.7.4 *Sea* Σ *una signatura,* $t, s \in \mathcal{T}(\Sigma \cup \{\Omega\})$ *y* $\mu \in M_\Sigma$ *tal que* $\mu_s \sqsubseteq \mu$. *Si* $t < s$, *entonces* $\tau_\mu(t) < \tau_\mu(s)$.

DEMOSTRACIÓN. Por inducción estructural sobre t. Si $t = \Omega$ (el único caso posible, pues $t < s$), entonces $\tau_\mu(t) = \tau_\mu(\Omega) = \Omega_\mu < \tau_\mu(s)$, pues debe ser $s \neq \Omega$ y, por tanto, $\tau_\mu(s) \neq \Omega_\mu$.

Si $t = f(\tilde{t})$, entonces $t < s$ implica que $s = f(\tilde{s})$ y existe un i, $1 \leq i \leq ar(f)$ tal que $t_i < s_i$ y $t_j \leq s_j$, para $1 \leq i \leq ar(f)$, $i \neq j$. Por la Proposición 5.7.2, $\tau_\mu(t_j) \leq \tau_\mu(s_j)$ para $i \neq j$ y, por la hipótesis de inducción, $\tau_\mu(t_i) < \tau_\mu(s_i)$. Puesto que $t_i < s_i$, debe tenerse que $s_i \neq \Omega$, es decir, $root(s_i) \in \Sigma$ y, dado que $\mu_s \sqsubseteq \mu$, tenemos que $i \in \mu(f)$. Por tanto, $\tau_\mu(f(\tilde{t})) = f_\mu(\tau_\mu(\tilde{t}_{\mu(f)})) < f_\mu(\tau_\mu(\tilde{s}_{\mu(f)})) = \tau_\mu(f(\tilde{s}))$. □

Las proposiciones anteriores se ven complementadas por la siguiente proposición.

Proposición 5.7.5 *Sea* Σ *una signatura,* $t \in \mathcal{T}(\Sigma \cup \{\Omega\})$, $s \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ *y* $\mu \in M_\Sigma$ *tal que* $\mu_t \sqsubseteq \mu$. *Entonces,*

1. *Si* $\tau_\mu(t) \leq \tau_\mu(s)$, *entonces* $t \leq s$.
2. *Si* $\tau_\mu(t) < \tau_\mu(s)$, *entonces* $t < s$.
3. *Si* $\tau_\mu(t) \uparrow \tau_\mu(s)$, *entonces* $t \uparrow s$.

DEMOSTRACIÓN. ($\tau_\mu(t) \leq \tau_\mu(s) \Rightarrow t \leq s$): Por inducción estructural sobre t. Si $t \in \Sigma \cup \{\Omega\}$, entonces $\tau_\mu(t) \leq \tau_\mu(s)$ implica que $t = \Omega$ o bien $t = s$, es decir, $t \leq s$.

Si $t = f(\tilde{t})$, entonces

$$\tau_\mu(t) = \tau_\mu(f(\tilde{t})) = f_\mu(\tau_\mu(\tilde{t}_{\mu(f)})) \leq f_\mu(\tau_\mu(\tilde{s}_{\mu(f)})) = \tau_\mu(f(\tilde{s})) = \tau_\mu(s)$$

si y sólo si para todo $i \in \mu(f)$, $\tau_\mu(t_i) \leq \tau_\mu(s_i)$. Por la hipótesis de inducción, $t_i \leq s_i$ para todo $i \in \mu(f)$. Si $i \notin \mu(f)$, entonces, puesto que $\mu_t \sqsubseteq \mu$ y t no contiene ocurrencias de variables, debe ser $t_i = \Omega$ y, por tanto, $t = f(\tilde{t}) \leq f(\tilde{s}) = s$.

($\tau_\mu(t) < \tau_\mu(s) \Rightarrow t < s$): Por inducción estructural. Si $t \in \Sigma \cup \{\Omega\}$, entonces debe tenerse $t = \Omega$ y $s \neq \Omega$. Por tanto, $\Omega_\mu = \tau_\mu(t) < \tau_\mu(s) \neq \Omega_\mu$.

Si $t = f(\tilde{t})$ y $s = f(\tilde{s})$ son tales que $\tau_\mu(t) < \tau_\mu(s)$, entonces existe un $i \in \mu(f)$ tal que $\tau_\mu(t_i) < \tau_\mu(s_i)$ y, para todo $j \in \mu(f) \backslash \{i\}$, $\tau_\mu(t_j) \leq \tau_\mu(s_j)$. Aplicando la hipótesis de inducción y siguiendo un razonamiento similar al del apartado 1, concluimos el resultado.

($\tau_\mu(t) \uparrow \tau_\mu(s) \Rightarrow t \uparrow s$): Inmediato al aplicar el resultado establecido en el apartado 1. □

Adviértase que la asunción $t \in \mathcal{T}(\Sigma \cup \{\Omega\})$ es necesaria para asegurar el resultado anterior.

Ejemplo 5.7.6 *Sea* $t = \mathtt{f(\Omega,x)}$, $s = \mathtt{f(a,y)}$ *y* $\mu(\mathtt{f}) = \{1\}$. *Se satisface* $\mu_t \sqsubseteq \mu$ *(recordemos la Observación 5.7.1). Tenemos que* $\tau_\mu(t) = \mathtt{f}_\mu(\Omega_\mu) \leq \mathtt{f}_\mu(\mathtt{a}_\mu) = \tau_\mu(s)$. *Sin embargo,* $t \not\trianglelefteq s$ *puesto que* $\mathtt{x} \not\trianglelefteq \mathtt{y}$.

La condición $\mu_t \sqsubseteq \mu$ tampoco es prescindible.

Ejemplo 5.7.7 *Sean los términos* $s = \mathtt{first(s(x),y::z)}$ *y* $t = \mathtt{first(0,\Omega)}$. *Si* $\mu(\mathtt{first}) = \{2\}$, *entonces* $\mu \sqsubset \mu_t$. *Tenemos que* $\tau_\mu(s) = \mathtt{first}_\mu(\mathtt{y::}_\mu\mathtt{z})$ *y* $\tau_\mu(t) = \mathtt{first}_\mu(\Omega_\mu)$. *Por tanto,* $\tau_\mu(t) \uparrow \tau_\mu(s)$, *pero* t *y* s *no son compatibles.*

Corolario 5.7.8 *Sea* Σ *una signatura,* $t \in \mathcal{T}(\Sigma \cup \{\Omega\})$ *y* $s \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$. *Sea* μ *tal que* $\mu_t \sqsubseteq \mu$. *Entonces,* $t \leq s$ *si y sólo si* $\tau_\mu(t) \leq \tau_\mu(s)$.

DEMOSTRACIÓN. Consecuencia de las Proposiciones 5.7.2 y 5.7.5. □

Este resultado tiene una consecuencia inmediata que también resulta de interés.

Teorema 5.7.9 *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRTE lineal por la izquierda y* $\mu \in M_\Sigma$ *tal que* $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. *Un término* $t \in \mathcal{T}(\Sigma, V)$ *es un redex de* $l \in L(\mathcal{R})$ *si y sólo si* $\tau_\mu(t)$ *es un redex de* $\tau_\mu(l) \in L(\mathcal{R}^\mu)$.

DEMOSTRACIÓN. De acuerdo con el Lema 2.3.3, t es un redex de l si y sólo si $l_\Omega \leq t$. Por la Proposición 5.7.2, tenemos que $\tau_\mu(l_\Omega) \leq \tau_\mu(t)$. Dado que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$, tenemos que $\mu_{l_\Omega} \sqsubseteq \mu$ y, por la Proposición 5.7.5, tenemos que $l_\Omega \leq t$ si y sólo si $\tau_\mu(l_\Omega) \leq \tau_\mu(t)$, lo cual, por el Lema 2.3.3, significa que $\tau_\mu(t)$ es un redex de $\tau_\mu(l)$ si y sólo si t es un redex de l. □

El siguiente teorema establece las condiciones para que sea posible determinar si un término es una forma μ-normal respecto a $\mathcal{R}$ comprobando si es una forma normal respecto a $\mathcal{R}^\mu$.

Teorema 5.7.10 *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRTE lineal por la izquierda y sea* $\mu \in M_\Sigma$ *tal que* $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. *Un término* $t \in \mathcal{T}(\Sigma, V)$ *es una forma* μ*-normal si y sólo si* $\tau_\mu(t)$ *es una forma normal respecto a* $\mathcal{R}^\mu$.

DEMOSTRACIÓN. Por inducción estructural, empleando el Teorema 5.7.9. □

Otro resultado derivado de las proposiciones anteriores es el siguiente corolario que será utilizado más adelante.

Corolario 5.7.11 *Sea Σ una signatura y $t, s \in \mathcal{T}(\Sigma \cup \{\Omega\})$. Sea $\mu \in M_\Sigma$ tal que $\mu_t \sqcup \mu_s \sqsubseteq \mu$. Entonces, $t = s$ si y sólo si $\tau_\mu(t) = \tau_\mu(s)$.*

DEMOSTRACIÓN. Si $t = s$, es evidente que $\tau_\mu(t) = \tau_\mu(s)$. Por otro lado, tenemos que $\tau_\mu(t) = \tau_\mu(s)$ si y sólo si $\tau_\mu(t) \leq \tau_\mu(s)$ y $\tau_\mu(t) \geq \tau_\mu(s)$. Por tanto, puesto que $t, s \in \mathcal{T}(\Sigma \cup \{\Omega\})$ y $\mu_t \sqcup \mu_s \sqsubseteq \mu$, por el Corolario 5.7.8, tenemos que $t \leq s$ y $t \geq s$, es decir, $t = s$. □

5.7.1 Aplicación a la Ω-reducción

En esta sección estudiamos algunas propiedades básicas de la tranformación τ_μ en conexión con la Ω-reducción. La primera de ellas establece la posibilidad de realizar un paso de Ω_μ-reducción siempre que pueda efectuarse un paso de Ω-reducción.

Proposición 5.7.12 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\mu \in M_\Sigma$ y $u \in O^\mu(t)$. Si $t \xrightarrow{u}_\Omega s$, entonces $\tau_\mu(t) \overset{\nu_t(u)}{\rightarrow}_{\Omega_\mu} \tau_\mu(s)$.*

DEMOSTRACIÓN. Por la Proposición 5.7.2, si $t|_u \uparrow l$ entonces $\tau_\mu(t|_u) \uparrow \tau_\mu(l)$. Por la Proposición 5.2.8, $\tau_\mu(t)|_{\nu_t(u)} \uparrow \tau_\mu(l)$. También tenemos que $s = t[\Omega]_u$. Por tanto, por la Proposición 5.2.8, $\tau_\mu(s) = \tau_\mu(t)[\Omega_\mu]_{\nu_t(u)}$. Por tanto, de acuerdo con la definición de $\mathcal{R}^\mu$, concluimos el resultado deseado. □

La siguiente proposición establece el resultado que complementa la proposición anterior: bajo ciertas condiciones, todo paso de Ω_μ-reducción induce un paso de Ω-reducción.

Proposición 5.7.13 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, y $u \in O^\mu(t)$. Si $\tau_\mu(t) \overset{\nu_t(u)}{\rightarrow}_{\Omega_\mu} \tau_\mu(s)$, entonces existe un $s' \equiv_{\tau_\mu} s$ tal que $t \xrightarrow{u}_\Omega s'$.*

DEMOSTRACIÓN. Por definición de $\mu^{com}_{\mathcal{R}}$ (y teniendo en cuenta la Observación 5.7.1), tenemos que, para todo $l \in L_\Omega(\mathcal{R})$, $\mu_l \sqsubseteq \mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Puesto que $\tau_\mu(t)|_{\nu_t(u)} \uparrow \tau_\mu(l)$, por la Proposición 5.2.8, $\tau_\mu(t|_u) \uparrow \tau_\mu(l)$. Dado que $\mu_l \sqsubseteq \mu$ y $l \in \mathcal{T}(\Sigma \cup \{\Omega\})$, por la Proposición 5.7.5, $t|_u \uparrow l$. Puesto que $\tau_\mu(s) = \tau_\mu(t)[\Omega_\mu]_{\nu_t(u)}$, por la Proposición 5.2.8, tenemos que $\tau_\mu(s) = \tau_\mu(t[\Omega]_u)$. Tomando $s' = t[\Omega]_u$, obtenemos que $t \rightarrow_\Omega s'$, lo cual concluye la demostración. □

El Ejemplo 5.7.7 demuestra que no se puede prescindir de la condición $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ en la Proposición 5.7.13. Por otro lado, debemos conformarnos con Ω-reductos *eqTau*-equivalentes (y no idénticos) al que consideramos en la Ω_μ-derivación.

Ejemplo 5.7.14 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 4.3.10 y la función de reemplazamiento μ utilizada en el ejemplo (que verifica $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$). Tenemos que*

$$\tau_\mu(\mathtt{f(a,a)}) = \mathtt{f}_\mu(\underline{\mathtt{a}_\mu}) \rightarrow_{\Omega_\mu} \mathtt{f}_\mu(\Omega_\mu) = \tau_\mu(\mathtt{f}(\Omega,\mathtt{b}))$$

pero

$$\mathtt{f(a,a)} \not\rightarrow_\Omega \mathtt{f}(\Omega,\mathtt{b})$$

Proposición 5.7.15 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Si $t \rightarrow_\Omega^* s$, entonces $\tau_\mu(t) \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$.*

DEMOSTRACIÓN. Por inducción en la longitud n de la derivación $t \rightarrow_\Omega^* s$. Si $n = 0$ es inmediato. Si $n > 0$, sea $t \xrightarrow{u}_\Omega t' \rightarrow_\Omega^* s$. Por la hipótesis de inducción, tenemos que $\tau_\mu(t') \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$. Si $u \in O^\mu(t)$, por la Proposición 5.7.12, $\tau_\mu(t) \xrightarrow{\nu_t(u)}_{\Omega_\mu} \tau_\mu(t') \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$. Si $u \notin O^\mu(t)$, entonces $\tau_\mu(t) = \tau_\mu(t')$. Por tanto, $\tau_\mu(t) \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$. □

Observemos que podemos realizar pasos de Ω-reducción sobre un término t que no corresponden a ningún paso de Ω_μ-reducción (si la Ω-reducción tiene lugar sobre ocurrencias no reemplazantes).

Proposición 5.7.16 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y sea $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$. Si $\tau_\mu(t) \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$, entonces existe un $s' \equiv_{\tau_\mu} s$ tal que $t \rightarrow_\Omega^* s'$.*

DEMOSTRACIÓN. Por inducción sobre la longitud n de la derivación $\tau_\mu(t) \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$. Si $n = 0$ es inmediato. Si $n > 0$, sea $\tau_\mu(t) \xrightarrow{\nu_t(u)}_{\Omega_\mu} \tau_\mu(t') \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$. Tenemos que $u \in O^\mu(t)$ y, por la Proposición 5.7.13, existe un $t'' \equiv_{\tau_\mu} t'$ tal que $t \xrightarrow{u}_\Omega t''$. Dado que $\tau_\mu(t') \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$ y $\tau_\mu(t'') = \tau_\mu(t')$, se tiene que $\tau_\mu(t'') \rightarrow_{\Omega_\mu}^* \tau_\mu(s)$ es una derivación de longitud $n-1$. Por la hipótesis de inducción, existe un $s' \equiv_{\tau_\mu} s$ tal que $t'' \rightarrow_\Omega^* s'$ y, por tanto, $t \rightarrow_\Omega^* s'$. □

El siguiente teorema permite asegurar que siempre que no sea posible realizar pasos de Ω-reducción sobre un término, tampoco será posible Ω_μ-reducir su μ-contracción.

Teorema 5.7.17 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y sea $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ es una forma $\rightarrow_\Omega$-normal, entonces $\tau_\mu(t)$ es una forma $\rightarrow_{\Omega_\mu}$-normal respecto a $\mathcal{R}^\mu$.*

DEMOSTRACIÓN. Por contradicción. Si no fuera así, debería existir una ocurrencia $v \in O(\tau_\mu(t))$ y un s tal que $\tau_\mu(t) \xrightarrow{v}_{\Omega_\mu} \tau_\mu(s)$. Puesto que $\nu_t : O^\mu(t) \rightarrow O(\tau_\mu(t))$ es una biyección, para todo término existe un $u \in O^\mu(t)$ tal que $v = \nu_t(u)$. Puesto que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$, por la Proposición 5.7.13 existe un $s' \equiv_{\tau_\mu} s$ tal que $t \rightarrow_\Omega s'$. Por tanto, t no es una forma $\rightarrow_\Omega$-normal, lo cual es una contradicción. □

No es posible, en general, asegurar la afirmación complementaria tal como revela el siguiente ejemplo.

Ejemplo 5.7.18 *Consideremos el SRT $\mathcal{R}$:*

$$\texttt{f(x,a)} \rightarrow \texttt{b}$$

y el término $\texttt{f(f(x,}\Omega\texttt{),b)}$. *Haciendo* $\mu = \mu^{com}_{\mathcal{R}}$, *tenemos que* $\tau_\mu(\texttt{f(f(x,}\Omega\texttt{),b)}) = \texttt{f}_\mu\texttt{(b}_\mu\texttt{)}$, *que es una forma* $\rightarrow_{\Omega_\mu}$*-normal, puesto que* $\mathcal{R}^\mu$ *es:*

$$\texttt{f}_\mu\texttt{(a}_\mu\texttt{)} \rightarrow \texttt{b}_\mu$$

Sin embargo, $\texttt{f(f(x,}\Omega\texttt{),b)}$ *no es una forma* $\rightarrow_\Omega$*-normal, pues tenemos que*

$$\texttt{f(}\underline{\texttt{f(x,}\Omega\texttt{)}}\texttt{,b)} \rightarrow_\Omega \texttt{f(}\Omega\texttt{,b)}$$

Esto se debe a que, a pesar de que $\tau_\mu(t)$ sea una forma $\rightarrow_{\Omega_\mu}$-normal, es posible que tengan lugar más Ω-reducciones sobre el término t si éste posee subtérminos compatibles con un esquema de redex $l \in L_\Omega(\mathcal{R})$ que estén situados sobre ocurrencias no reemplazantes de t. Sin embargo, todavía podemos asegurar lo siguiente.

Teorema 5.7.19 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ y $C[\,] = CMR^\mu(t)$. Si $\tau_\mu(t)$ es una forma $\rightarrow_{\Omega_\mu}$-normal, entonces $C[\widetilde{\Omega}]$ es rígido.*

Demostración. Por contradicción. Si $C[\widetilde{\Omega}]$ no es rígido, entonces $\omega(C[\widetilde{\Omega}]) \neq C[\widetilde{\Omega}]$ y, por la Proposición 2.5.9, $\omega(C[\widetilde{\Omega}]) < C[\widetilde{\Omega}]$. Luego existe una Ω-derivación

$$C[\widetilde{\Omega}] = s_1 \overset{u_1}{\rightarrow}_\Omega s_2 \overset{u_2}{\rightarrow}_\Omega \cdots \overset{u_{n-1}}{\rightarrow}_\Omega s_n \overset{u_n}{\rightarrow}_\Omega s_{n+1} = \omega(C[\widetilde{\Omega}])$$

Sea u_i la primera ocurrencia reemplazante considerada en dicha derivación. Esta ocurrencia debe existir puesto que, si no fuera así, debe tenerse que $\omega(C[\widetilde{\Omega}]) = C[\widetilde{\Omega}]$, dado que los s_j para $1 \leq j \leq n+1$ sólo diferirían entre sí en ocurrencias no reemplazantes, luego pertenecerían a la misma $\equiv_{\tau_\mu}$-clase con un mismo representante que, por la Proposición 5.2.5, debe ser precisamente $C[\,]$ lo cual, por hipótesis, no es cierto.

Sin embargo, el término s_i pertenecerá a la clase de t, es decir, $\tau_\mu(t) = \tau_\mu(s_i)$ y, puesto que $s_i \overset{u_i}{\rightarrow}_\Omega s_{i+1}$ y $u_i \in O^\mu(s_i)$, por la Proposición 5.7.12, tenemos $\tau_\mu(s_i) \overset{\nu_{s_i}(u_i)}{\rightarrow}_\Omega \tau_\mu(s_{i+1})$. Es decir, $\tau_\mu(t) \overset{\nu_t(u_i)}{\rightarrow}_\Omega \tau_\mu(s_{i+1})$, lo que significa que $\tau_\mu(t)$ no es una forma $\rightarrow_{\Omega_\mu}$-normal, llegándose a una contradicción. □

Capítulo 6

Confluencia

La confluencia de la relación de reescritura asociada a un SRT dado es una propiedad de interés computacional evidente: podemos asegurar que, cuando un término tiene una forma normal, cualquier estrategia normalizante empleada para reducir dicho término obtendrá el mismo resultado.

El problema de asegurar la confluencia de la relación de reescritura asociada a un SRT a partir de las características estructurales del SRT, es decir, la *caracterización sintáctica* de la confluencia, ha sido estudiado exhaustivamente en [BK86, Gra96, Hue80, HL91, Ros73, Wir95], entre otros.

Dos métodos bien conocidos para demostrar la confluencia de un SRT son los derivados de los siguientes resultados:

1. Si el SRT es terminante y todo par crítico $\langle t, t' \rangle$ es convergente, el SRT es confluente [Hue80].

2. Si el SRT es lineal por la izquierda y no tiene pares críticos (es decir, es ortogonal), entonces es confluente [HL91].

En este capítulo, presentamos una extensión de estos resultados que aplicamos a la demostración de la confluencia de la *rsc* para un determinado SRT y una función de reemplazamiento dada. En la Sección 6.1 se identifican algunas de las condiciones sintácticas básicas que son relevantes para demostrar la confluencia de la *rsc* mediante la extensión de los métodos anteriores.

Siguiendo el primero de los métodos anteriores (1), que se basa en demostrar la confluencia local de un SRT terminante y utilizar el Lema de Newman (Lema 2.1.2) para concluir la confluencia del SRT, en la Sección 6.2 se proporciona un criterio para demostrar la μ-confluencia local de un SRT.

Siguiendo el segundo método (2), que consiste en demostrar la confluencia fuerte de la relación de reescritura paralela (expresada por el *parallel moves lemma*, Lema 2.3.9)

y emplear el Lema 2.1.1 para concluir la confluencia, en la Sección 6.3 caracterizamos (sintácticamente) una clase de SRTs cuya relación de reescritura sensible al contexto paralela asociada es fuertemente confluente.

La Sección 6.4 aplica los resultados anteriores a la demostración de la μ-confluencia.

6.1 Identificación de condiciones sintácticas básicas para la μ-confluencia

El concepto de μ-confluencia de un sistema de reescritura es análogo al del caso estándar no restringido.

Definición 6.1.1 (μ-confluencia) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Decimos que $\mathcal{R}$ es μ-confluente si $\hookrightarrow_{\mathcal{R}(\mu)}$ es confluente.*

En primer lugar, mostramos que la confluencia de la relación de reescritura no asegura la confluencia de la *rsc* y que los métodos usuales no son directamente aplicables sin imponer alguna condición adicional. Esto se pone de manifiesto con el siguiente ejemplo.

Ejemplo 6.1.2 *Consideremos el SRT $\mathcal{R}$:*

$$\texttt{f(x)} \rightarrow \texttt{g(x,x)}$$
$$\texttt{h(0)} \rightarrow \texttt{0}$$

Si imponemos $\mu(\texttt{f}) = \{1\}$ y $\mu(\texttt{g}) = \{1\}$, tenemos las siguientes μ-derivaciones que llevan a formas μ-normales distintas para el término inicial considerado:

$$\underline{\texttt{f(h(0))}} \hookrightarrow \texttt{g}(\underline{\texttt{h(0)}},\texttt{h(0)}) \hookrightarrow \texttt{g(0,h(0))}$$

y

$$\texttt{f}(\underline{\texttt{h(0)}}) \hookrightarrow \underline{\texttt{f(0)}} \hookrightarrow \texttt{g(0,0)}$$

Observemos que el SRT del Ejemplo 6.1.2 satisface las condiciones de aplicación de los dos métodos anteriormente mencionados: es lineal por la izquierda, terminante y no tiene pares críticos. Por tanto es confluente, pero no es μ-confluente para la función de reemplazamiento μ considerada en el ejemplo. De hecho, el Ejemplo 4.3.10 ya sugería la no aplicabilidad de los criterios usuales a la *rsc*.

La razón de esta discrepancia se encuentra en la asimetría, introducida por la función de reemplazamiento μ, en la capacidad de realizar ulteriores reducciones sobre los subtérminos que aparecen al instanciar las variables de la parte derecha de la regla de reescritura que se aplica en un determinado paso de reducción. Esto es lo que sucede con la regla $\texttt{f(x)} \rightarrow \texttt{g(x,x)}$ en el ejemplo anterior. A continuación caracterizamos la clase de SRTs donde no se producen este tipo de asimetrías.

Definición 6.1.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$. Decimos que la regla $l \to r$ tiene* variables μ-reemplazantes homogéneas por la izquierda *si, para toda $x \in Var^\mu(l)$, $O_x(l) = O^\mu_x(l)$ y $O_x(r) = O^\mu_x(r)$. El SRT $\mathcal{R}$ tiene variables μ-reemplazantes homogéneas por la izquierda si todas sus reglas tienen variables μ-reemplazantes homogéneas por la izquierda.*

Observación 6.1.4 *Merece la pena reparar en dos clases de SRTs que admiten una formulación más simple de la propiedad descrita por la Definición 6.1.3:*

1. *Si un SRT $\mathcal{R}$ es lineal por la izquierda, tendrá variables μ-reemplazantes homogéneas por la izquierda si para toda regla $l \to r$, la existencia de una variable $x \in Var^\mu(l)$ implica que $O_x(r) = O^\mu_x(r)$.*

2. *Si un SRT $\mathcal{R}$ no tiene variables reemplazantes en ninguna lhs $l \in L(\mathcal{R})$, trivialmente tiene variables μ-reemplazantes homogéneas por la izquierda. Más adelante estudiaremos una clase de SRTs que verifica esta propiedad (ver la Sección 10.2).*

Los SRTs de los Ejemplos 1.1.2, 3.3.3 y 3.4.4 tienen variables μ-reemplazantes homogéneas por la izquierda respecto a la función de reemplazamiento μ dada en el ejemplo correspondiente. Los SRTs de los Ejemplos 3.4.3, 4.3.10 y 6.1.2 carecen de esta propiedad.

Los resultados sobre confluencia que vamos a presentar exigen siempre que los SRTs cuya μ-confluencia deseamos demostrar tengan variables μ-reemplazantes homogéneas por la izquierda. Sin embargo, también tendremos ocasión de ofrecer algún ejemplo de que esta condición no es *necesaria* para demostrar la μ-confluencia del SRT (ver el Ejemplo 11.1.7).

De forma análoga a la confluencia, otro motivo de pérdida de μ-confluencia es la presencia de *pares críticos*. Los pares críticos representan situaciones en las que una misma regla de reescritura (con la posible colaboración de otras) es capaz de originar distintas secuencias de reducción que no siempre será posible hacer converger en un mismo término, lo cual siempre implica la pérdida de la confluencia.

La restricciones de reemplazamiento pueden evitar el problema introducido por los pares críticos limitando la posibilidad efectiva de que aparezcan secuencias de reducción divergentes. Esto se consigue imponiendo que la ocurrencia de una *lhs* de una regla que da lugar a un par crítico sea no reemplazante. Esta observación conduce a una noción de par crítico adecuada para la *rsc*.

Definición 6.1.5 (Par μ-crítico) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$. Un par crítico $\langle \sigma(l)[\sigma(r')]_u, \sigma(r) \rangle$ de $\mathcal{R}$ tal que $u \in O^\mu_\Sigma(l)$ es un* par μ-crítico.

De forma análoga a la reescritura no restringida, decimos que un par μ-crítico $\langle t, t' \rangle$ es μ-convergente si t y t' tienen un μ-reducto común, es decir, si existe un término s tal que $t \hookrightarrow^* s$ y $t' \hookrightarrow^* s$.

El siguiente ejemplo muestra cómo pueden emplearse las restricciones de reemplazamiento para evitar solapamientos que podrían llevar a reducciones divergentes.

Ejemplo 6.1.6 *Consideremos el SRT $\mathcal{R}$:*

$$\begin{array}{ll} \mathtt{g(b,a)} & \rightarrow \mathtt{b} \\ \mathtt{a} & \rightarrow \mathtt{b} \end{array}$$

y una función de reemplazamiento μ tal que $\mu(\mathtt{g}) = \{1\}$. Este SRT tiene un solo par crítico $\langle \mathtt{g(b,a)}[\mathtt{b}]_2, \mathtt{b} \rangle = \langle \mathtt{g(b,b)}, \mathtt{b} \rangle$. No obstante, dado que 2 no es una ocurrencia reemplazante de `g(b,a)`*, no hay pares μ-críticos en el SRT.*

Una vez introducidos los conceptos anteriores, podemos describir cómo se extienden las técnicas usuales para demostrar la confluencia de un SRT a la demostración de la μ-confluencia de un SRT. Los métodos descritos de manera informal al principio del capítulo se basan en demostrar ciertas propiedades elementales de la relación de reescritura o de la relación de reescritura paralela que, bajo ciertas condiciones, implican la confluencia. Dichas propiedades son la *confluencia local* y la *confluencia fuerte*, respectivamente.

De acuerdo con el Lema de Newman (Lema 2.1.2), una relación localmente confluente y terminante es confluente. Este hecho fundamenta el primero de los métodos presentados que extienden a la *rsc* el método usual en reescritura [Hue80]. Bajo la hipótesis de que el SRT es terminante, se demuestra que la relación de reescritura que verifica la propiedad de convergencia de pares críticos es localmente confluente y se emplea el Lema de Newman para concluir la confluencia.

El segundo método explota el hecho de que si es posible demostrar que el cierre transitivo de dos relaciones coincide y una de ellas es fuertemente confluente, entonces la otra es confluente (Lema 2.1.1). El resultado presentado en [HL91] hace uso de esa propiedad. En lugar de estudiar la confluencia fuerte de la relación de reescritura en un paso ($\rightarrow$), se establece la confluencia fuerte de la relación de reescritura *paralela* ($\between$) para SRTs ortogonales (esto lo expresa el *parallel moves lemma*, Lema 2.3.9). Puesto que el cierre transitivo de ambas relaciones coincide ($\rightarrow^+ = \between^+$), la propiedad anterior asegura la confluencia del SRT.

Dado que las propiedades anteriores son válidas para relaciones abstractas definidas sobre un conjunto A cualquiera, vamos a proceder de forma análoga a Huet [Hue80] y a Huet y Lévy [HL91] estableciendo condiciones que aseguren la confluencia local de la relación de reescritura sensible al contexto en un paso y la confluencia fuerte de la relación de reescritura paralela sensible al contexto.

6.2 Confluencia local

La siguiente proposición establece la posibilidad de simular, mediante una secuencia de reducción sensible al contexto, una reducción múltiple realizada sobre instancias de variables que ocupan posiciones reemplazantes en un término dado.

Proposición 6.2.1 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$. Sea $t \in \mathcal{T}(\Sigma, V)$, $x \in V$ y $O_x(t) = \{u_i\}_{i=1}^n$, para $n \geq 0$. Sean $\sigma, \sigma' \in Subst(\Sigma, V)$ tales que $\sigma(x) \hookrightarrow \sigma'(x)$ y, para todo $y \neq x$, $\sigma(y) = \sigma'(y)$. Sea $s_0 = \sigma(t)$ y $s_i = s_{i-1}[\sigma'(x)]_{u_i}$ $1 \leq i \leq n$. Si $O_x(t) = O_x^\mu(t)$, entonces $s_i \hookrightarrow^{n-i} \sigma'(t)$, $0 \leq i \leq n$.*

DEMOSTRACIÓN. Por inducción sobre $n = |O_x(t)|$, el número de ocurrencias de la variable x en $O(t)$. Si $n = 0$ es inmediato. Si $n > 0$, consideremos $u \in O_x(t)$. Por la hipótesis de inducción, asumimos $s_1 \hookrightarrow^{n-1} \sigma'(t)$. Puesto que $u \in O^\mu(t)$, por la Proposición 3.2.9, $u \in O^\mu(\sigma(t))$. Por la Proposición 2.2.3, $\sigma(t)|_u = \sigma(x)$. Puesto que $\sigma(x) \hookrightarrow \sigma'(x)$, por la Proposición 3.3.6, obtenemos $\sigma(t) = s_0 \hookrightarrow s_1 = \sigma(t)[\sigma'(x)]_u$. De este modo, empleando la Proposición 3.2.10 y la hipótesis de inducción se sigue la conclusión deseada. □

La siguiente proposición es un caso particular de la Proposición 3.7 de [Hue80], puesto que un par μ-crítico es siempre un par crítico que involucra las mismas reglas de reescritura y ocurrencia de solapamiento.

Proposición 6.2.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$. Sean $l_1 \to r_1, l_2 \to r_2 \in R$, $u \in O^\mu(l_1)$, $t = l_1|_u \notin V$ y $\sigma_1, \sigma_2 \in Subst(\Sigma, V)$ tales que $\sigma_1(t) = \sigma_2(l_2)$. Existe un par μ-crítico $\langle t_1, t_2 \rangle$ de $\mathcal{R}$ y $\theta \in Subst(\Sigma, V)$ tal que $\sigma_1(l_1)[\sigma_2(r_2)]_u = \theta(t_1)$ y $\sigma_1(r_1) = \theta(t_2)$.*

El siguiente teorema es el análogo al conocido "Lema de los pares críticos" de Huet [Hue80, Klo92]

Teorema 6.2.3 (Confluencia local de la *rsc*) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT con variables μ-reemplazantes homogéneas por la izquierda respecto a $\mu \in M_\Sigma$. El SRT $\mathcal{R}$ es localmente μ-confluente si y sólo si todo par μ-crítico es μ-convergente.*

DEMOSTRACIÓN. La demostración es similar a la del Lema 3.1 en [Hue80]. Introducimos las modificaciones necesarias para integrar la presencia de restricciones de reemplazamiento.

Sea $\langle t_1, t_2 \rangle = \langle \sigma(l_1)[\sigma(r_2)]_u, \sigma(r_1) \rangle$ un par μ-crítico de $\mathcal{R}$.

Para la parte *sólo si*, puesto que $u \in O^\mu(l_1)$, por la Proposición 3.2.9, $\sigma(l_1) \hookrightarrow t_1$. Obviamente, $\sigma(l_1) \hookrightarrow \sigma(r_2) = t_2$. Puesto que $\hookrightarrow$ es localmente confluente, existe un s tal que $t_1 \hookrightarrow^* s$ y $t_2 \hookrightarrow^* s$, es decir, el par μ-crítico $\langle t_1, t_2 \rangle$ es μ-convergente.

($\Leftarrow$) Para la parte *si*, sean $t, t', t'' \in \mathcal{T}(\Sigma, V)$ tales que $t \hookrightarrow t'$ y $t \hookrightarrow t''$, es decir, existen reglas $l_1 \to r_1, l_2 \to r_2 \in R$, ocurrencias $u_1, u_2 \in O^{\mu}(t)$, y sustituciones σ_1, σ_2 tales que $t|_{u_1} = \sigma_1(l_1)$, $t|_{u_2} = \sigma_2(l_2)$, $t' = t[\sigma_1(r_1)]_{u_1}$ y $t'' = t[\sigma_2(r_2)]_{u_2}$. Para probar la confluencia local de $\mathcal{R}$, debemos demostrar que existe un μ-reducto común s de t' y t''. Consideramos dos casos según la posición relativa de los dos redexes.

1. $u_1 \parallel u_2$, es decir, las ocurrencias u_1 y u_2 no son comparables.
   Por la propiedad de persistencia y por la Proposición 3.2.10, tenemos que $t'|_{u_2} = \sigma_2(l_2)$, $u_2 \in O^{\mu}(t')$ y $t''|_{u_1} = \sigma_1(l_1)$, $u_1 \in O^{\mu}(t'')$. Por conmutatividad, $s = t'[\sigma_2(r_2)]_{u_2} = t''[\sigma_1(r_1)]_{u_1}$ y, por tanto, $t' \hookrightarrow s$ y $t'' \hookrightarrow s$.

2. Si u_1 y u_2 son comparables, sin pérdida de generalidad podemos asumir que $u_1 \leq u_2$ (el caso $u_2 \leq u_1$ es totalmente análogo). Sea $u_2 = u_1.v$. Por la propiedad de cancelación: $\sigma_1(l_1)|_v = \sigma_2(l_2)$ y, por distributividad, $t''|_{u_1} = \sigma_1(l_1)[\sigma_2(r_2)]_v$. Puesto que $u_2 = u_1.v$ y $u_2 \in O^{\mu}(t)$, por la Proposición 3.2.1, tenemos que $v \in O^{\mu}(t|_{u_1})$ o, lo que es lo mismo, $v \in O^{\mu}(\sigma_1(l_1))$. Por tanto, $\sigma_1(l_1) \hookrightarrow \sigma_1(l_1)[\sigma_2(r_2)]_v$.

   Ahora debemos demostrar que existe un término s tal que $\sigma_1(r_1) \hookrightarrow^* s$ y $\sigma_1(l_1)[\sigma_2(r_2)]_v \hookrightarrow^* s$. Entonces, la existencia de un μ-reducto común de t' y t'' se sigue empleando la Proposición 3.3.6. De acuerdo con la Proposición 2.2.3 debemos considerar dos casos:

   (a) $v = v_1.v_2$, $l_1|_{v_1} = x \in V$ y $\sigma_2(l_2) = \sigma_1(x)|_{v_2}$.
   Sea la sustitución σ_1' definida como sigue: $\sigma_1'(x) = \sigma_1(x)[\sigma_2(r_2)]_{v_2}$ y $\sigma_1'(y) = \sigma_1(y)$ para toda variable $y \neq x$. Tomamos el término $s = \sigma_1'(r_1)$. Es suficiente mostrar que $\sigma_1(x) \hookrightarrow \sigma_1'(x)$: puesto que $\sigma_1(x)|_{v_2} = \sigma_2(l_2)$, tenemos que $\sigma_1(x)$ se reescribe a la ocurrencia v_2 emplando la regla $l_2 \to r_2$, es decir, $\sigma_1(x) \to \sigma_1'(x)$. Mostremos ahora que $v_2 \in O^{\mu}(\sigma_1(x))$: puesto que, por la Proposición 2.2.3, $\sigma_1(l_1)|_{v_1} = \sigma_1(x)$ y $v \in O^{\mu}(\sigma_1(l_1)) \Leftrightarrow v_1.v_2 \in O^{\mu}(\sigma_1(l_1))$, por la Proposición 3.2.1, $v_2 \in O^{\mu}(\sigma_1(x))$. De esta forma, $\sigma_1(x) \hookrightarrow \sigma_1'(x)$. También, por el Corolario 3.2.4, $v_1 \in O^{\mu}(\sigma_1(l_1))$ y por la Proposición 3.2.9, $v_1 \in O^{\mu}(l_1)$ puesto que $v_1 \in O(l_1)$. Por tanto, $l_1|_{v_1} = x$ es una variable μ-reemplazante, $x \in Var^{\mu}(l_1)$. Puesto que $\mathcal{R}$ tiene variables μ-reemplazantes homogéneas por la izquierda, cualquier otra ocurrencia de x en l_1 y r_1 es μ-reemplazante y, por la Proposición 6.2.1, tenemos que $\sigma_1(r_1) \hookrightarrow^* \sigma_1'(r_1) = s$ y $\sigma_1(l_1)[\sigma_2(r_2)]_v \hookrightarrow^* \sigma_1'(l_1)$. Dado que $\sigma_1'(l_1) \hookrightarrow \sigma_1'(r_1) = s$ tenemos que $\sigma_1(l_1)[\sigma_2(r_2)]_v \hookrightarrow^* s$. Recordemos que $t'|_{u_1} = \sigma_1(r_1)$, $t''|_{u_1} = \sigma_1(l_1)[\sigma_2(r_2)]_v$ y, por la Proposición 3.2.10, $u_1 \in O^{\mu}(t')$ y $u_1 \in O^{\mu}(t'')$. Aplicando la Proposición 3.3.6 al contexto $t[\ ]_{u_1}$, obtenemos que $t \hookrightarrow t' \wedge t \hookrightarrow t''$ implican que t' y t'' tienen un μ-reducto común.

(b) $\exists l \notin V$ tal que $l = l_1|_v$ y $\sigma_2(l_2) = \sigma_1(l)$.
Puesto que $\sigma_1(l_1)$ se μ-reescribe a $\sigma_1(l_1)[\sigma_2(r_2)]_v$ utilizando la regla $l_2 \to r_2$ y la ocurrencia $v \in O^\mu(\sigma_1(l_1))$, por la Proposición 6.2.2, existe un par μ-crítico $\langle t_1, t_2 \rangle$ y una sustitución θ tal que $\sigma_1(l_1)[\sigma_2(r_2)]_v = \theta(t_1)$ y $\sigma_1(r_1) = \theta(t_2)$. Por hipótesis, existe un término t_3 tal que $t_1 \hookrightarrow^* t_3$ y $t_2 \hookrightarrow^* t_3$. Tomamos $s = \theta(t_3)$ y la conclusión se deduce por la Proposición 3.3.6 y por la estabilidad de $\hookrightarrow$ (Proposición 3.3.8).

Con esto completamos la demostración. □

Es sencillo ver que, incluso imponiendo fuertes restricciones en los reemplazamientos, no es posible asegurar la validez del Teorema 6.2.3 para SRTEs.

Ejemplo 6.2.4 *Sea el SRTE ortogonal* $\mathcal{R}$

```
f(a) → a
f(b) → b
c    → x
```

Tomemos $\mu = \mu_\perp$. *Aunque* $\mathcal{R}$ *no tiene pares críticos, no es localmente* μ*-confluente, pues:*

```
c ↪ f(a)
```

y

```
c ↪ f(b)
```

pero `f(a)` *y* `f(b)` *no tienen un* μ*-reducto común.*

6.3 Confluencia fuerte

Si somos capaces de demostrar la confluencia fuerte de una relación, también podemos inferir que ésta es confluente. Esta afirmación puede probarse haciendo $S = R$ en el Lema 2.1.1. De esta manera concluimos que, si R es fuertemente confluente, entonces es confluente. Sin embargo, este lema tiene aplicaciones más interesantes cuando las relaciones R y S no coinciden.

En la teoría de los sistemas de reescritura ortogonales, el *parallel moves lemma* (Lema 2.3.9) es un resultado bien conocido que, entre otras cosas, establece la confluencia fuerte de la reescritura *paralela*.

Para generalizar este resultado a la *rsc*, realizamos un análisis de las condiciones de definición de dichas multiderivaciones elementales canónicas cuando las restricciones de reemplazamiento condicionan su definición efectiva. Pero el *parallel moves lemma* exhibe una propiedad aún más interesante: proporciona un procedimiento *canónico* para definir multiderivaciones elementales $B \backslash A$ y $A \backslash B$ dadas las multiderivaciones

elementales A y B que arrancan de un mismo término t. El concepto fundamental sobre el que se asienta la definición de estas derivaciones canónicas $B\backslash A$ y $A\backslash B$ es el de residuo de un redex respecto a una derivación (Definición 2.3.8). En primer lugar, definimos el concepto de residuo respecto a una derivación sensible al contexto en un paso.

Definición 6.3.1 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT, $\mu \in M_\Sigma$, y $A^\mu : t \overset{[u,\alpha]}{\hookrightarrow} s$ una μ-derivación elemental, donde $\alpha : l \to r \in R$. Sea $v \in O^\mu_\mathcal{R}(t)$. El conjunto $v\backslash A^\mu$ de* residuos *de $t|_v$ respecto a A^μ es un subconjunto de $O(s)$ definido como sigue: $v\backslash A^\mu = v\backslash A$, donde $A : t \overset{[u,\alpha]}{\to} s$.*

Dado que cualquier μ-derivación es también una derivación, la definición anterior es correcta. El hecho de que $v\backslash A \subseteq O_\mathcal{R}(s)$ es esencial para definir el concepto de derivación residual, etc. (ver la Sección 2.3.3). Esto quiere decir que la derivación A (que contrae el redex $t|_u$) deja inalterada la posibilidad de reducir los residuos del redex $t|_v$ en los pasos de reducción posteriores.

Necesitamos asegurar una propiedad análoga para las μ-derivaciones $A^\mu : t \overset{u}{\hookrightarrow} s$, es decir, que siempre que $v \in O^\mu_\mathcal{R}(t)$ obtendremos $v\backslash A^\mu \subseteq O^\mu_\mathcal{R}(s)$. Desafortunadamente, aunque seguirá siendo un redex, un residuo de un redex reemplazante no tiene porqué ser reemplazante.

Ejemplo 6.3.2 *Consideremos el SRT ortogonal $\mathcal{R}$ del Ejemplo 6.1.2. Si definimos $\mu(\mathtt{f}) = \{1\}$ y $\mu(\mathtt{g}) = \{1\}$, podemos realizar la siguiente μ-derivación:*

$$A^\mu : \underline{\mathtt{f(h(0))}} \hookrightarrow \mathtt{g(h(0),h(0))}$$

Las ocurrencias residuales del redex reemplazante `h(0)` *en $t =$* `f(h(0))` *respecto a la μ-derivación A^μ son $\{1,2\}$. Sin embargo, $2 \notin O^\mu($`g(h(0),h(0))`$)$ y, por tanto,* 2 no *es una ocurrencia reemplazante del redex* `h(0)` *en* `g(h(0),h(0))`.

El problema se pone de manifiesto al considerar las ocurrencias de redexes reemplazantes $v \in O^\mu_\mathcal{R}(t)$ que se encuentran *bajo* la ocurrencia $u \in O^\mu_\mathcal{R}(t)$ que contrae la μ-derivación elemental A^μ. De acuerdo con la Definición 2.3.8 (apartado (*iii*)), las ocurrencias residuales $u.w_1.v_1$ que se encuentran en el conjunto $v\backslash A^\mu$ están parcialmente constituidas por ocurrencias $w_1 \in O_V(r)$ donde $\alpha : l \to r$ es la regla involucrada en la μ-derivación elemental A^μ. Para garantizar que $u.w_1.v_1$ es una ocurrencia reemplazante de un redex, debemos asegurar que w_1 también es reemplazante, es decir, $w_1 \in O^\mu(r)$. Si exigimos que $\mathcal{R}$ tenga variables μ-reemplazantes homogéneas por la izquierda, se verifica esta propiedad. Los SRTs que tienen variables μ-reemplazantes homogéneas por la izquierda exhiben una propiedad importante: los subtérminos que instancian (mediante emparejamiento) las variables reemplazantes de las partes izquierdas de las reglas de reescritura son también reemplazantes en la instancia correspondiente de la parte derecha de la regla. Por tanto, cuando consideramos SRTs

con variables μ-reemplazantes homogéneas por la izquierda, los residuos de los redexes reemplazantes son también reemplazantes.

Observemos que, en la Definición 2.3.8, la ortogonalidad es esencial para asegurar que la definición de residuo considera realmente todos los casos. De hecho, el caso $v > u$, tal como ha sido tratado en el apartado (iii) de la definición, quedaría incompleto sin exigir la ortogonalidad. También en el caso (ii), $v < u$, podría suceder que $v \notin O_{\mathcal{R}}(s)$ sin la asunción de ortogonalidad: cuando el SRT es ortogonal, la reducción de un redex $t|_u$ situado bajo de un redex $t|_v$ no impide la reducción ulterior de $s|_v$, donde $s = t[\sigma(r)]_u$; es decir, no se destruye el carácter de redex de $t|_v$.

Podemos relajar el requerimiento de ortogonalidad (manteniendo aún la consistencia de la Definición 6.3.1) prohibiendo el reemplazamiento de redexes solapantes mediante una función de reemplazamiento. La extensión del concepto de SRT ortogonal en presencia de una función de reemplazamiento μ es la siguiente.

Definición 6.3.3 (SRT μ-ortogonal) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT lineal por la izquierda y $\mu \in M_\Sigma$. Decimos que $\mathcal{R}$ es μ-ortogonal si no tiene pares μ-críticos.*

Ejemplo 6.3.4 *Consideremos el SRT $\mathcal{R}$:*

$$\begin{array}{ll} \texttt{f(x,y)} \to \texttt{g(y,y)} & \texttt{a} \to \texttt{c} \\ \texttt{g(x,a)} \to \texttt{f(x,b)} & \texttt{b} \to \texttt{b} \end{array}$$

y la función de reemplazamiento μ definida como sigue: $\mu(\texttt{f}) = \mu(\texttt{g}) = \{1\}$. De acuerdo con la Definición 6.3.3, $\mathcal{R}$ es μ-ortogonal. Observemos que $\mathcal{R}$ también tiene variables μ-reemplazantes homogéneas por la izquierda.

A continuación demostramos que los conceptos introducidos en los párrafos anteriores son suficientes para asegurar la propiedad deseada para los residuos de un redex reemplazante respecto a una μ-derivación elemental.

Proposición 6.3.5 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT μ-ortogonal con variables μ-reemplazantes homogéneas por la izquierda respecto a $\mu \in M_\Sigma$. Sea $A^\mu : t \overset{[u,\alpha]}{\hookrightarrow} s$ una μ-derivación elemental. Si $v \in O^\mu_{\mathcal{R}}(t)$, entonces $v \backslash A^\mu \subseteq O^\mu_{\mathcal{R}}(s)$.*

Demostración. De acuerdo con la Definición 6.3.1, $v \backslash A^\mu = v \backslash A$, por lo que consideramos los casos establecidos en la Definición 2.3.8. Recordemos que $s = t[\sigma(r)]_u$ y $u, v \in O^\mu_{\mathcal{R}}(t)$. Sea $t \overset{[v,\alpha']}{\to} s'$ para alguna regla $\alpha' : l' \to r'$. El caso $u = v$ es inmediato.

1. $v \parallel u$. Entonces $v \backslash A^\mu = \{v\}$. Puesto que $v \in O^\mu(t)$, por la Proposición 3.2.10, $v \in O^\mu(t[\sigma(r)]_u)$. Por tanto, $v \in O^\mu(s)$; más precisamente, $v \in O^\mu_{\mathcal{R}}(s)$.

2. $v < u$. De forma similar al caso anterior, empleando la Proposición 3.2.10, concluimos que $v \in O^\mu(s)$. Comprobamos ahora que $v \in O_{\mathcal{R}}(s)$. Lo hacemos

por contradicción: si $v \notin O_{\mathcal{R}}(s)$, entonces la reducción de la ocurrencia redex u ha destruido el redex anterior. Por tanto, $u = v.w$ y $w \in O_{\Sigma}(l')$ (en otro caso, si $w \in O_V(l')$ o bien $w \notin O(l')$ no hay ningún problema). Sin embargo, puesto que $t|_v = \sigma'(l')$ y $t|_u = \sigma(l) = \sigma'(l')|_w = \sigma'(l'|_w)$, entonces α y α' definen un par crítico. Dado que $u = v.w \in O^{\mu}(t)$, por la Proposición 3.2.1, $w \in O^{\mu}(\sigma(l'))$. Por la Proposición 3.2.9, $w \in O^{\mu}_{\Sigma}(l')$, y esto significa que α y α' definen un par μ-crítico. Pero esto contradice la μ-ortogonalidad del SRT. Por tanto, $v \in O^{\mu}_{\mathcal{R}}(s)$.

3. Si $v > u$, razonando como en el punto anterior la única posibilidad es $v = u.w.v_1$ y $l|_w = x \in V$. Consideremos $v' \in v \backslash A^{\mu}$. Entonces, $v' = u.w_1.v_1$ y $r|_{w_1} = x$. Puesto que $v \in O^{\mu}(t)$, por el Corolario 3.2.4, $u.w \in O^{\mu}(t)$. Puesto que $t|_u = \sigma(l)$, por la Proposición 3.2.1, $w \in O^{\mu}(\sigma(l))$. Por tanto, dado que $w \in O(l)$, por la Proposición 3.2.9, $w \in O^{\mu}(l)$: es decir, $x \in Var^{\mu}(l)$. De esta forma, puesto que $\mathcal{R}$ tiene variables μ-reemplazantes homogéneas por la izquierda, $w_1 \in O^{\mu}(r)$. También, por la Proposición 3.2.1, $v_1 \in O^{\mu}(t|_{u.w})$. Por tanto, puesto que $w_1 \in O^{\mu}(r)$, por la Proposición 3.2.9, $w_1 \in O^{\mu}(\sigma(r))$. Puesto que $\sigma(r)|_{w_1} = t|_{u.w} = s|_{u.w_1}$, por la Proposición 3.2.1, concluimos que $u.w_1.v_1 \in O^{\mu}(s)$. Esto significa que $v' \in O^{\mu}_{\mathcal{R}}(s)$.

□

De esta manera, la extensión sensible al contexto de la noción de *residuo* es correcta. También lo es la noción de derivación residual $B^{\mu} \backslash A^{\mu}$. Ciertamente, si no podemos asegurar que $v \backslash A^{\mu} \subseteq O^{\mu}_{\mathcal{R}}(s)$, tampoco podemos asegurar que $U \backslash A^{\mu} \subseteq O^{\mu}_{\mathcal{R}}(s)$. Dado que la derivación residual $B^{\mu} \backslash A^{\mu}$ se define como la derivación que contrae el conjunto $U \backslash A^{\mu}$, dicha derivación no estaría bien definida para el caso sensible al contexto, si este conjunto pudiera contener ocurrencias de redexes no reemplazantes. Esto significa que $B^{\mu} \backslash A^{\mu} = B \backslash A$ para toda μ-derivación $A^{\mu} : t \overset{U}{\hookrightarrow} t'$, con $U \subseteq O^{\mu}_{\mathcal{R}}(t)$ conteniendo ocurrencias disjuntas dos a dos y μ-derivaciones $B^{\mu} : t \overset{W}{\hookrightarrow} t''$, cuyo conjunto de ocurrencias, $W \subseteq O^{\mu}_{\mathcal{R}}(t)$, consta también de ocurrencias disjuntas dos a dos. Ahora podemos formular la extensión del *parallel moves lemma* a la *rsc*.

Lema 6.3.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT μ-ortogonal con variables μ-reemplazantes homogéneas por la izquierda respecto a $\mu \in M_{\Sigma}$. Sean A^{μ} y B^{μ} dos μ-multiderivaciones elementales que parten del mismo término t. Entonces, $B^{\mu}(A^{\mu} \backslash B^{\mu})$ y $A^{\mu}(B^{\mu} \backslash A^{\mu})$ son μ-multiderivaciones que parten de t y llevan al mismo término s y, para todo $u \in O^{\mu}_{\mathcal{R}}(t)$, tenemos $u \backslash B^{\mu}(A^{\mu} \backslash B^{\mu}) = u \backslash A^{\mu}(B^{\mu} \backslash A^{\mu})$.*

DEMOSTRACIÓN. Puesto que $A^{\mu} = A$, $B^{\mu} = B$, $B^{\mu} \backslash A^{\mu} = B \backslash A$, y $A^{\mu} \backslash B^{\mu} = A \backslash B$, aplicamos el Lema 2.3.9. □

El *parallel moves lemma* para $\Vdash$ (Lema 6.3.6) demuestra que $\Vdash$ es fuertemente confluente para SRTs μ-ortogonales con variables μ-reemplazantes homogéneas por la izquierda.

Teorema 6.3.7 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT μ-ortogonal con variables μ-reemplazantes homogéneas por la izquierda respecto a $\mu \in M_\Sigma$. Entonces, la relación $\Vdash$ es fuertemente confluente.*

6.4 μ-confluencia

El siguiente teorema proporciona un método para demostrar la μ-confluencia de un SRT μ-terminante con variables μ-reemplazantes homogéneas por la izquierda, respecto a la función de reemplazamiento μ. En el enunciado del mismo, $\bar{t}$ denota una forma μ-normal de $t \in \mathcal{T}(\Sigma, V)$.

Teorema 6.4.1 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT μ-terminante con variables μ-reemplazantes homogéneas por la izquierda respecto a $\mu \in M_\Sigma$. Entonces, $\mathcal{R}$ es μ-confluente si y sólo si $\bar{t}_1 = \bar{t}_2$ para todo par μ-crítico $\langle t_1, t_2 \rangle$ de $\mathcal{R}$.*

DEMOSTRACIÓN. Para demostrar la parte *sólo si*, consideramos que para cualquier par μ-crítico $\langle t_1, t_2 \rangle$ de $\mathcal{R}$ existe un $t \in \mathcal{T}(\Sigma, V)$ tal que $t \hookrightarrow t_1$ y $t \hookrightarrow t_2$. Si $\mathcal{R}$ es μ-confluente y μ-terminante, entonces t admite una forma μ-normal única $\bar{t}_1 = \bar{t}_2$. Para la parte *si*, observemos que $\bar{t}_1 = \bar{t}_2$ implica que t_1 y t_2 tienen un μ-reducto común. Por el Teorema 6.2.3, $\mathcal{R}$ es localmente μ-confluente. Dado que $\mathcal{R}$ es μ-terminante y localmente μ-confluente, por el Lema 2.1.2, $\mathcal{R}$ es confluente. □

Dado un SRT $\mathcal{R}$ μ-terminante con variables μ-reemplazantes homogéneas por la izquierda respecto a la función de reemplazamiento μ, el Teorema 6.4.1 sugiere un procedimiento efectivo para demostrar la μ-confluencia de $\mathcal{R}$. Puesto que $\mathcal{R}$ es μ-terminante, para cualquier par μ-crítico $\langle t_1, t_2 \rangle$ podemos calcular, en tiempo finito, tanto $\bar{t}_1$ como $\bar{t}_2$ y, tras esto, comprobar si $\bar{t}_1 = \bar{t}_2$. Si el SRT carece de pares μ-críticos, demostrar la μ-confluencia es más sencillo.

Corolario 6.4.2 *Un SRT $\mathcal{R} = (\Sigma, R)$ μ-terminante, sin pares μ-críticos y con variables μ-reemplazantes homogéneas por la izquierda respecto a $\mu \in M_\Sigma$ es μ-confluente.*

Ejemplo 6.4.3 *Consideremos el SRT $\mathcal{R}$ y la función de reemplazamiento del Ejemplo 6.1.6. Este SRT no es confluente. Por ejemplo:*

$$\texttt{g(b,}\underline{\texttt{a}}\texttt{)} \rightarrow \texttt{g(b,b)}$$

y

$$\underline{\texttt{g(b,a)}} \rightarrow \texttt{b}$$

No obstante, puesto que $\mathcal{R}$ es terminante, por el Teorema 5.1.4, $\mathcal{R}$ es μ-terminante. Puesto que $\mathcal{R}$ tiene variables μ-reemplazantes homogéneas por la izquierda y no tiene pares μ-críticos, por el Corolario 6.4.2, $\mathcal{R}$ es μ-confluente.

Sin embargo, para SRTs lineales por la izquierda sin pares μ-críticos (es decir, SRTs μ-ortogonales), no necesitamos imponer que el SRT sea μ-terminante, puesto que disponemos de un resultado más potente.

Teorema 6.4.4 *Un SRT $\mathcal{R} = (\Sigma, R)$ μ-ortogonal con variables μ-reemplazantes homogéneas por la izquierda respecto a $\mu \in M_\Sigma$ es μ-confluente.*

DEMOSTRACIÓN. Por el Teorema 6.3.7, $\Vdash$ es fuertemente confluente. Por el Lema 3.3.10, $\hookrightarrow^+ = \Vdash^+$. Por el Lema 2.1.1, $\hookrightarrow$ es confluente. □

Por ejemplo, el SRT $\mathcal{R}$ del Ejemplo 6.3.4 respecto a la función de reemplazamiento μ dada en dicho ejemplo no es μ-terminante. No obstante, puesto que $\mathcal{R}$ es μ-ortogonal y tiene variables μ-reemplazantes homogéneas por la izquierda, por el Teorema 6.4.4, $\mathcal{R}$ es μ-confluente.

6.5 Trabajos relacionados

Podemos comparar nuestros resultados sobre confluencia de la *rsc* con el establecido por Maranget para su *reducción condicional* [Mar90]. En la primera sección de su artículo, Maranget sostiene que, *de forma similar a los SRTs estándar, es suficiente comprobar que las partes izquierdas no solapan*[1] para asegurar la confluencia de dicha relación de reducción. Los ejemplos anteriores demuestran que esto no es suficiente en general (Ejemplo 6.1.2). De forma análoga a como se ha procedido en la Sección 6.3, en [Mar90], Maranget demuestra su afirmación indirectamente, estableciendo previamente la confluencia fuerte de la *reducción condicional*. No obstante lo que se ha comentado, la demostración es correcta debido a la imposición de una condición muy fuerte sobre la clase de SRTs condicionales que son considerados. Recordemos que Maranget considera SRTs μ-ortogonales con la siguiente restricción adicional: $V_a(r) \subseteq V_a(l)$ y $V_f(r) \subseteq V_f(l)$ para toda regla $l \to r$ del SRT, donde $V_a(t) = Var^\mu(t)$ y $V_f(t) = \{x \in Var(t) \mid \exists u \in \overline{O^\mu}(t).\ t|_u = x\}$. Dicha restricción implica que los SRTs de la clase bajo consideración tienen variables reemplazantes homogéneas por la izquierda respecto a cualquier función de reemplazamiento (*función de dominio*, en la terminología de Maranget) de las admitidas: dado que los SRTs considerados son lineales por la izquierda, sólo necesitamos demostrar que $x \in Var^\mu(l)$ implica

[1] "as in the case of standard TRSs, it is sufficient to check the non-overlapping condition on the left-hand sides of the reduction rules" [Mar90], pag. 257.

$O_x(r) = O^{\mu}_x(r)$ (recordemos la Observación 6.1.4). Siendo $x \in Var^{\mu}(l)$, supongamos que existe una ocurrencia no reemplazante de x en r. En este caso, $x \in V_f(r)$. Puesto que $V_f(r) \subseteq V_f(l)$, tenemos que $x \in V_f(l)$; es decir, hay una ocurrencia no reemplazante de x en l. Sin embargo, debido a la linealidad por la izquierda, esto no es posible. Así, los SRTs considerados por Maranget son una *subclase* de los SRTs μ-ortogonales con variables μ-reemplazantes homogéneas por la izquierda. Por tanto, con relación a la μ-confluencia, nuestros resultados son más generales. Por ejemplo, la μ-confluencia del SRT del Ejemplo 6.3.4 no podría demostrarse empleando los resultados de Maranget. Además, Maranget realiza ciertas asunciones que se aceptan sin poner de manifiesto cómo condicionan éstas el resultado obtenido sobre confluencia de la *reducción condicional*.

Respecto a la *reescritura perezosa* de Kamperman y Walters [KW95], los autores estudian la noción de ζ-confluencia, que es más débil que la confluencia. La ζ-confluencia surge al considerar la ζ-igualdad en lugar de la igualdad usual, a la hora de definir la convergencia de términos. Dos términos t y s son ζ-iguales si al reemplazar los subtérminos maximales no reemplazantes por una nueva constante ζ obtenemos el mismo término. De hecho, la ζ-igualdad coincide con la $\equiv_{\tau_\mu}$-equivalencia de la Sección 5.2. Kamperman y Walters sólo han establecido que la *reescritura perezosa* preserva la ζ-confluencia. Dado que, a diferencia de la confluencia, cuya definición involucra la igualdad de términos, la ζ-confluencia está basada en la noción de ζ-igualdad, no es posible emplearla para asegurar que distintas secuencias de *reducción perezosa* pueden ser reencauzadas para producir términos *iguales*. Sólo puede asegurarse esto respecto a la ζ-igualdad y, además, ni siquiera empleando la propia restricción considerada (la *reescritura perezosa*) para reunir (salvo ζ-igualdad) los términos discrepantes, sino que sólo se asegura que la reescritura *no restringida* podrá hacerlo. Esto constituye un resultado más débil que el que se ha presentado aquí.

Parte II

Estrategias de reducción sensible al contexto

Capítulo 7

Estrategias sensibles al contexto

Cuando, dada una función de reemplazamiento μ, no es posible asegurar un comportamiento μ-terminante para el SRT considerado, es necesario seleccionar cuidadosamente las reducciones sensibles al contexto que llevan a obtener una forma μ-normal, en caso de que ésta exista. La elección de una *estrategia sensible al contexto* adecuada (es decir, μ-normalizante) cobra especial importancia y este capítulo está dedicado a la definición y análisis de dichas estrategias.

La definición de estrategias sensibles al contexto exige resolver un problema análogo al planteado en la reescritura estándar: el de la definición *efectiva* de una estrategia normalizante, esto es, una estrategia que ante cualquier término normalizable obtiene siempre su forma normal. Este problema es especialmente difícil cuando se desea que la estrategia sea secuencial, es decir, que cada transformación elemental (útil) efectuada sobre el término corresponda a un único paso de reducción.

La teoría de las reducciones necesarias (para la normalización) de Huet y Lévy [HL91, KM91] sirve a este propósito cuando el objetivo es definir estrategias normalizantes. En la Sección 7.1, veremos que dicha teoría no es completamente adecuada para fundamentar la definición de estrategias sensibles al contexto. Afortunadamente, podemos utilizar la teoría de reducciones necesarias para la *estabilización* de términos. En nuestra formalización, haremos uso de la formulación introducida recientemente por Middeldorp [Mid97].

En la Sección 7.2 demostramos que dicha teoría no sólo es adecuada para analizar los cómputos sensibles al contexto sino que sus resultados fundamentales pueden ser refinados introduciendo en el análisis de las reducciones estabilizantes la presencia de las restricciones de reemplazamiento sensibles al contexto.

En la Sección 7.3 demostramos también que es posible emplear los índices fuerte-

mente secuenciales de Huet y Lévy como aproximación decidible a los redexes necesarios para la estabilización de un término.

De acuerdo con este resultado, en la Sección 7.4 estudiamos la capacidad de las restricciones de reemplazamiento sensibles al contexto para capturar ocurrencias que corresponden a índices en determinadas clases de términos.

En la Sección 7.5 demostramos que las estrategias de reducción de índices μ-reemplazantes son μ-normalizantes.

7.1 Concepto de estrategia sensible al contexto

El concepto de estrategia de reescritura sensible al contexto es una extensión natural de la noción estándar de estrategia de reescritura.

Definición 7.1.1 (Estrategia sensible al contexto) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$. Una estrategia (determinista) sensible al contexto (o simplemente una μ-estrategia) es una función $\mathbb{H}_{\mathcal{R}(\mu)} : \mathcal{T}(\Sigma, V) \to \mathcal{T}(\Sigma, V)$ tal que $\mathbb{H}_{\mathcal{R}(\mu)}(t) = t$ si t es una forma μ-normal y $t \hookrightarrow \mathbb{H}_{\mathcal{R}(\mu)}(t)$ en cualquier otro caso.*

La Definición 7.1.1 se refiere a estrategias secuenciales. Su extensión al caso paralelo es inmediata. Es interesante observar que una μ-estrategia sólo permite realizar la μ-normalización de un término, siendo en general imposible su normalización completa. Esto se debe a que, siempre que $O^\mu_\mathcal{R}(t) = \emptyset$, una μ-estrategia ya no está obligada a realizar más reducciones. Por tanto, una μ-estrategia *no* es una estrategia de reescritura.

Observación 7.1.2 *De forma análoga a como se descompone una estrategia $\mathbb{F}_\mathcal{R}$ en sus componentes $(\mathbb{P}_\mathcal{R}, \mathbb{R}_\mathcal{R})$ (Sección 2.4) también podemos descomponer una μ-estrategia $\mathbb{H}_{\mathcal{R}(\mu)}$ en sus componentes $(\mathbb{P}^\mu_\mathcal{R}, \mathbb{R}^\mu_\mathcal{R})$ sin más que sustituir $\to_\mathcal{R}$ por $\hookrightarrow_{\mathcal{R}(\mu)}$ y $O_\mathcal{R}$ por $O^\mu_\mathcal{R}$ en las definiciones correspondientes.*

La teoría de las reducciones necesarias desarrollada por Huet y Lévy para los SRTs ortogonales [HL91, KM91] cubre dos aspectos esenciales de la definición de estrategias de reescritura secuenciales:

1. Formaliza una noción de *eficiencia* de los cómputos por reescritura. Si una derivación normalizante sólo contrae redexes necesarios, puede decirse que es tan eficiente como es posible puesto que no se realizará ningun paso de reducción innecesario[1].

[1] En realidad, esto sólo es cierto cuando disponemos de una implementación basada en reducción de grafos que es capaz de almacenar ocurrencias distintas del mismo redex en una única posición de memoria compartida.

2. Permite definir estrategias normalizantes: las estrategias que sólo llevan a cabo reducciones necesarias son normalizantes (Teoremas 2.5.1 y 2.5.2).

No obstante, la reducción arbitraria de redexes necesarios no es útil como base para definir secuencias de reducción sensibles al contexto que sean capaces de encontrar una forma μ-normal siempre que ésta existe. Esto es debido a la existencia de términos que tienen una forma μ-normal pero *no* tienen forma normal. Para dichos términos, la noción de redex necesario no resulta útil pues, por definición, todo redex de un término que no tiene forma normal es necesario.

Ejemplo 7.1.3 *Sea el siguiente SRT $\mathcal{R}$:*

$$\begin{aligned} \mathtt{f(x)} &\to \mathtt{g(f(x))} \\ \mathtt{a} &\to \mathtt{a} \end{aligned}$$

Si establecemos $\mu(\mathtt{f}) = \{1\}$ *y* $\mu(\mathtt{g}) = \emptyset$*, el término* `g(f(a))` *es una forma μ-normal de* `f(a)` *pues*

$$\underline{\mathtt{f(a)}} \hookrightarrow \mathtt{g(f(a))}$$

Sin embargo, no es posible obtener `g(f(a))` *a partir de* `f(a)` *mediante reducción del redex necesario (y reemplazante)* `a`*:*

$$\mathtt{f(\underline{a})} \hookrightarrow \mathtt{f(\underline{a})} \hookrightarrow \cdots$$

Por tanto, contrariamente a lo que sucede con el cálculo de formas normales, la reducción reiterada de *cualquier* redex necesario no supone la obtención de una forma μ-normal aunque ésta exista.

El Teorema 4.3.8 demuestra que, bajo la asunción de linealidad por la izquierda para el SRT $\mathcal{R}$ considerado, e imponiendo una función de reemplazamiento μ mayor o igual que la función de reemplazamiento canónica, toda forma μ-normal es una forma estable. Esto significa que toda secuencia de reducción de un término t a forma μ-normal es también una secuencia de estabilización del término y que dicha estabilización tiene lugar *antes* (o en el momento) de alcanzar la forma μ-normal. Esta observación sugiere emplear la teoría de *reducciones necesarias para la estabilización* [Mid97] para formalizar la definición de estrategias de μ-normalización como estrategias que realizan reducciones necesarias para la *estabilización* en lugar de reducciones necesarias para la *normalización* como en la teoría de Huet y Lévy.

De hecho, al efectuar este análisis, es posible refinar la teoría desarrollada por Middeldorp mediante la consideración de las restricciones de reemplazamiento sensibles al contexto. En cuanto a la definición efectiva (computable) de μ-estrategias μ-normalizantes, demostramos la posibilidad de emplear las aproximaciones computables desarrolladas para el cálculo de redexes necesarios (los llamados redexes fuertemente necesarios de Huet y Lévy [HL91, KM91]) como aproximaciones computables a

los redexes necesarios para la estabilización, lo cual permite emplear dichas estrategias en tareas de μ-normalización.

7.2 Cómputos necesarios para estabilización y reescritura sensible al contexto

Nuestro primer resultado supone un refinamiento del Teorema 2.5.4 (Teorema 4.3 de [Mid97]) que establece que *"en un SRT ortogonal, todo término no estable contiene un redex necesario para su estabilización"*. Demostramos que esto puede formularse más precisamente: todo término no estable contiene un redex *reemplazante* que es necesario para su estabilización. Para establecer este resultado necesitamos algunos lemas auxiliares.

Lema 7.2.1 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT ortogonal. Sea t un término no estable, que tampoco es un redex y tal que $t \xrightarrow{>\epsilon}{}^{+} \sigma(l)$ para alguna regla $l \to r \in R$. Sea P el conjunto de ocurrencias de subtérminos propios no estables de t. Sea $u \in minimal(P \cap O_\Sigma(l))$ y sea $v \in O_\mathcal{R}(t|_u)$ una ocurrencia de un redex necesario para estabilizar $t|_u$. Entonces $t|_{u.v}$ es necesario para estabilizar t.*

DEMOSTRACIÓN. Implícita en la demostración del Teorema 4.3 de [Mid97]. □

Lema 7.2.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT ortogonal y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_\mathcal{R} \sqsubseteq \mu$. Sea t un término no estable tal que $t \xrightarrow{>\epsilon}{}^{+} \sigma(l)$ para alguna regla $l \to r \in R$. Sea P el conjunto de ocurrencias de subtérminos propios no estables de t. Entonces, $minimal(P \cap O_\Sigma(l)) \subseteq O^\mu(t)$.*

DEMOSTRACIÓN. Sea $u \in minimal(P \cap O_\Sigma(l))$. Dado que $t \xrightarrow{>\epsilon}{}^{+} \sigma(l)$, tenemos que $sprefix_t(u) = sprefix_l(u)$. De no ser así, algún subtérmino $t|_v$ con $v < u$ debería ser reducido para permitir el emparejamiento con la parte $l|_v$ de la regla considerada; en ese caso, $t|_v$ no sería estable y tendríamos que $v \in P$. Aún más, puesto que $u \in O_\Sigma(l)$, tendríamos $v \in P \cap O_\Sigma(l)$, pero, puesto que $v < u$, ello significaría que u no es minimal en $P \cap O_\Sigma(l)$ lo cual nos llevaría a una contradicción. Puesto que $sprefix_t(u) = sprefix_l(u)$ y $\mu^{com}_\mathcal{R} \sqsubseteq \mu$, tenemos que $u \in O_\Sigma(l) \subseteq O^\mu(l)$. Por la Proposición 3.2.3, concluimos que $u \in O^\mu(t)$. □

A continuación establecemos el resultado anunciado anteriormente.

Teorema 7.2.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT ortogonal y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_\mathcal{R} \sqsubseteq \mu$. Todo término no estable contiene un redex μ-reemplazante necesario para su estabilización.*

DEMOSTRACIÓN. La demostración es una adaptación de la realizada por Middeldorp para demostrar el Teorema 4.3 en [Mid97]. Por tanto, simplemente esbozamos la demostración dada por Middeldorp teniendo en cuenta las peculiaridades que surgen al considerar las restricciones sensibles al contexto.

Dado un término t, se distinguen dos casos: que t sea un redex y que no lo sea.

Dado que el SRT es ortogonal, la reducción de un redex a la ocurrencia raíz siempre es necesaria para estabilizar el término. Puesto que siempre se verifica $\epsilon \in O^{\mu}(t)$, se tiene que $\epsilon \in O^{\mu}_{\mathcal{R}}(t)$ es una ocurrencia de un redex reemplazante cuya reducción es necesaria para estabilizar t.

Si el término t no es un redex, cualquier secuencia de reducción que lleve t a una forma estable t' puede descomponerse como sigue: $t \xrightarrow{>\epsilon}{}^{+} \sigma(l) \xrightarrow{\epsilon} t'' \rightarrow^{*} t'$. De acuerdo con el Lema 2.3.6, el patrón l para la reducción $\sigma(l) \xrightarrow{\epsilon} t''$ no depende de la secuencia de reducción concreta $t \xrightarrow{>\epsilon}{}^{+} \sigma(l)$ que se haya seguido inicialmente. Ahora consideremos el conjunto P de ocurrencias de subtérminos propios no estables de t. Los candidatos a redexes necesarios para estabilizar t son los redexes necesarios para estabilizar los subtérminos $t|_u$ con $u \in \mathit{minimal}(P \cap O_{\Sigma}(l))$. En primer lugar, demostramos que u es μ-reemplazante, es decir, $u \in O^{\mu}(t)$. Esto es consecuencia del Lema 7.2.2. Por la hipótesis de inducción, $t|_u$ contiene un redex μ-reemplazante s necesario para estabilizar $t|_u$. Por el Lema 7.2.1, s también es necesario para estabilizar t. Dado que s es μ-reemplazante en $t|_u$ y que $u \in O^{\mu}(t)$, por la Proposición 3.2.1, s es un redex μ-reemplazante necesario para estabilizar t. □

El Teorema 7.2.3 demuestra que el conjunto de redexes reemplazantes de un término no estable siempre contiene redexes necesarios para su estabilización. De esta manera, al buscar redexes necesarios para estabilizar un término, obtenemos una cierta ventaja al poder centrar nuestra atención en un menor número de posibilidades. No obstante, al limitar nuestra atención al conjunto de redexes reemplazantes de un término no estable, por un lado dejamos de considerar redexes necesarios para estabilizar el término y, por otro lado, todavía incluimos algunos redexes innecesarios para ello.

Ejemplo 7.2.4 *Consideremos el SRT ortogonal* $\mathcal{R}$*:*

```
g(a,x,y) → x
g(d,x,a) → a
b        → a
c        → d
```

y el conjunto de todas las secuencias de estabilización para el término $t =$ `g(a,b,c)`*:*

1. `g(a,b,c)` → `b` → `a`
2. `g(a,b,c)` → `g(a,a,c)` → `a`

3. $\mathtt{g(a,\underline{b},c)} \rightarrow \mathtt{g(a,a,\underline{c})} \rightarrow \underline{\mathtt{g(a,a,d)}} \rightarrow \mathtt{a}$

4. $\mathtt{g(a,b,\underline{c})} \rightarrow \mathtt{g(a,\underline{b},d)} \rightarrow \underline{\mathtt{g(a,a,d)}} \rightarrow \mathtt{a}$

5. $\mathtt{g(a,b,\underline{c})} \rightarrow \underline{\mathtt{g(a,b,d)}} \rightarrow \underline{\mathtt{b}} \rightarrow \mathtt{a}$

Como puede observarse, la reducción del redex b *a la ocurrencia* 2 *en* g(a,b,c) *es necesaria para estabilizar el término. Si imponemos* $\mu(\mathtt{g}) = \{1,3\}$ *(observemos que* $\mu = \mu^{com}_{\mathcal{R}}$*), el redex* b *no es* μ*-reemplazante puesto que* $2 \notin O^{\mu}_{\mathcal{R}}(\mathtt{g(a,b,c)}) = \{\epsilon, 3\}$*. Por tanto, existen redexes que, siendo necesarios para estabilizar* t*, no son* μ*-reemplazantes.*

Por otra parte, tal como muestra la segunda derivación, no es necesario reducir el redex μ*-reemplazante* c *de* t *para estabilizarlo. Por tanto, encontramos redexes* μ*-reemplazantes que no son necesarios para estabilizar* t*.*

El Ejemplo 7.2.4 muestra que, para aislar de forma efectiva algún redex necesario para estabilizar un término, es preciso establecer una regla de selección adicional que limite el conjunto de redexes considerados más allá de las restricciones establecidas por una función de reemplazamiento.

Nuestro primer resultado en este sentido ofrece un refinamiento de otro resultado de Middeldorp: el que establece que *los redexes externos de un término no estable constituyen un conjunto necesario de redexes para su estabilización.* De nuevo es posible demostrar que podemos limitar nuestra atención a los redexes reemplazantes.

Teorema 7.2.5 *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRT ortogonal y sea* $\mu \in M_{\Sigma}$ *tal que* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$*. Si* t *es un término no estable, entonces* $minimal(O^{\mu}_{\mathcal{R}}(t))$ *es un conjunto necesario de redexes para estabilizar* t*.*

DEMOSTRACIÓN. Asumiendo la notación introducida en la demostración del Teorema 7.2.3, procedemos por inducción estructural. Si t es un redex, es inmediato. Si no lo es, tal como se razonó en la demostración del Teorema 7.2.3, para cada $u \in minimal(P \cap O_{\Sigma}(l))$, $minimal(O^{\mu}_{\mathcal{R}}(t|_u))$ es un conjunto necesario de redexes para estabilizar $t|_u$. Sea $v \in minimal(O^{\mu}_{\mathcal{R}}(t|_u))$ una ocurrencia de un redex necesario para estabilizar $t|_u$. Por el Lema 7.2.1, $u.v$ también es una ocurrencia de un redex necesario para estabilizar t. Por tanto, para todo $u \in minimal(P \cap O_{\Sigma}(l))$, $u.minimal(O^{\mu}_{\mathcal{R}}(t|_u))$ es un conjunto necesario de redexes para estabilizar t. Puesto que $u \in O^{\mu}(t)$, por la Proposición 3.2.1, $u.minimal(O^{\mu}_{\mathcal{R}}(t|_u)) \subseteq minimal(O^{\mu}_{\mathcal{R}}(t))$, lo cual concluye la demostración. □

Teorema 7.2.6 *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRT confluente y* $\mu \in M_{\Sigma}$ *tal que* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$*. Toda* μ*-estrategia independiente del contexto y estabilizante es* μ*-normalizante.*

DEMOSTRACIÓN. La demostración es análoga a la del Teorema 6.4 de [Mid97] sin más que tener presente la función de reemplazamiento μ. □

7.3 Índices fuertemente secuenciales como aproximaciones a los redexes necesarios para estabilización

En general, no es posible decidir si la reducción de un redex determinado es necesaria para estabilizar el término donde se encuentra dicho redex [Ken94, Mid97]. Tampoco es posible decidir si la reducción de un redex es necesaria para obtener la forma normal del término que lo contiene [HL91, KM91].

Encontrar condiciones para resolver estos problemas de decidibilidad es importante si pretendemos definir estrategias secuenciales de estabilización o normalización basadas en la propiedad (para la estabilización -Teorema 2.5.7- o para la normalización -Teorema 2.5.2-) que establece la imposibilidad de definir secuencias de reducción necesarias infinitas. Si somos capaces de encontrar [DM97]:

1. un método de cálculo efectivo que permita saber qué (ocurrencias de) redexes necesitamos reducir para obtener la forma normal (estable) del término, y

2. una clase de SRTs donde, siempre que un término no se encuentre en forma normal (estable), podamos aplicar el método anterior con la seguridad de que vamos a encontrar al menos un redex necesario para normalizarlo (estabilizarlo),

entonces podremos especificar estrategias de normalización (estabilización) para la clase de SRTs en cuestión. El primer requerimiento exige disponer de un método para discernir si es necesario o no reducir un redex determinado dentro de un término que no está en la forma deseada. Si cumplimos con el segundo requisito seremos capaces de aplicar con éxito el susodicho método siempre que se precise hacerlo.

En el caso de las estrategias secuenciales *normalizantes*, dicho estudio fue iniciado también por Huet y Lévy [HL79, HL91] al menos en la formulación del problema apuntada aquí[2]. Huet y Lévy buscaban definir estrategias secuenciales capaces de establecer el paso de reducción a realizar sin examinar las posibles derivaciones que pueden arrancar del término en cuestión para tomar esa decisión (sin anticipar reducciones). A pesar de que la ortogonalidad garantiza la presencia de un redex necesario en cada término reducible (Teorema 2.5.1), no asegura que éste sea computable. Como resultado de su análisis, Huet y Lévy definieron la noción de *índice secuencial* que se apoya en el concepto de índice de un predicado secuencial introducida por Kahn y Plotkin [KP78, KP93] y que constituía una primera aproximación para localizar los redexes necesarios. Los SRTs (ortogonales) secuenciales son aquéllos que aseguran que todo término que no está en forma normal contiene un redex situado en la posición

[2] Aunque en [O'D77] ya se estudió el problema de la definición de estrategias normalizantes y se propuso una solución, ésta es menos general que en [HL79, HL91].

marcada por un índice secuencial. Por desgracia, los índices secuenciales no son computables y tampoco es decidible si un SRT es secuencial [HL91]. Por ese motivo, los autores introdujeron la noción de *índice fuertemente secuencial* y demostraron que todo redex situado sobre una ocurrencia de un índice fuertemente secuencial es necesario para normalizar el término donde se encuentra. Proporcionaron un método efectivo para localizarlos y definieron la clase de sistemas de reescritura *fuertemente secuenciales* como aquélla que asegura la existencia, en todo término que no se encuentra en forma normal, de un redex situado en una posición índice. También proporcionaron un criterio para elucidar si un SRT satisface dicha condición, es decir, la propiedad de secuencialidad fuerte es decidible [HL91]. Este criterio fue simplificado por Klop y Middeldorp [KM91].

La noción de redex necesario tal como la definieron Huet y Lévy pierde sentido cuando los SRTs involucrados no son ortogonales. Esto se debe a que, en ese caso, no es posible asegurar que todo término que no está en forma normal tenga un redex necesario. No obstante, tanto la noción de índice (cuya definición es independiente del carácter ortogonal de los SRTs) como la de SRT fuertemente secuencial han sido empleadas por Toyama para definir estrategias normalizantes para sistemas lineales por la izquierda con posibles solapamientos [Toy92]. Jouannaud y Sadfi demostraron que es decidible si un SRT lineal por la izquierda es fuertemente secuencial [JS94].

Hasta el momento no había sido analizada en profundidad la posibilidad de emplear índices para aproximar ocurrencias de redexes necesarios para *estabilizar* términos ni el papel que desempeñan los sistemas fuertemente secuenciales como clase de SRTs que permiten la definición de estrategias de estabilización de términos. A continuación caracterizamos y demostramos dicha relación.

Los SRTEs cuasi ortogonales poseen la siguiente propiedad: todo redex situado sobre una ocurrencia índice es externo.

Lema 7.3.1 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE cuasi ortogonal y $t \in \mathcal{T}(\Sigma, V)$. Sean $u, v \in O_{\mathcal{R}}(t)$ tales que $u \neq v$. Si $u \in \mathcal{I}(t[\Omega]_u)$, entonces $v \not\leq u$.*

DEMOSTRACIÓN. Supongamos que $v \leq u$. Puesto que $u \neq v$, tenemos que $v < u$. Puesto que $t|_v = \sigma(l)$ para alguna $l \in L(\mathcal{R})$, por el Lema 2.3.4, $t|_v[\bullet]_u = t[\bullet]_u|_v = \sigma'(l)$ Por tanto, $t[\bullet]_u|_v \uparrow l_\Omega$. Luego $\bullet$ no puede encontrarse en $\omega(t[\bullet]_u)$ y, por tanto, $u \notin \mathcal{I}(t[\Omega]_u)$, contradiciendo la hipótesis inicial. □

Dicha propiedad ya no se cumple cuando consideramos sistemas débilmente ortogonales.

Ejemplo 7.3.2 *Consideremos el SRT débilmente ortogonal $\mathcal{R}$ [AM96, Toy92]:*

```
pred(succ(x)) → x
succ(pred(x)) → x
```

El término $t =$ `pred(succ(pred(x)))` *contiene dos ocurrencias de redexes: $O_{\mathcal{R}}(t) = \{\epsilon, 1\}$. Tanto ϵ como 1 son índices de t, puesto que $\omega(t[\bullet]_\epsilon) = \bullet$ y $\omega(t[\bullet]_1) =$* $\omega($`pred(`$\bullet$`)`$) =$ `pred(`$\bullet$`)`*. Por tanto, existe un redex no externo situado sobre una ocurrencia que corresponde a un índice.*

Estas propiedades son importantes para demostrar nuestro resultado principal: que los redexes de términos no estables que están situados sobre ocurrencias de índices son necesarios para su estabilización. Primero empleamos las propiedades anteriores para particularizar la noción de residuo de una ocurrencia de redex *externo* respecto a un paso de reescritura elemental. Dado un SRT cuasi ortogonal $\mathcal{R} = (\Sigma, R)$, un paso de reducción $A : t \xrightarrow{v} s$ (para alguna regla $l \to r \in R$) y una ocurrencia de redex externa $u \in O_{\mathcal{R}}(t)$, el conjunto $u\backslash A$ de *residuos* del redex $t|_u$ respecto a A es un subconjunto de $O_{\mathcal{R}}(s)$ definido como sigue: $u\backslash A = \emptyset$ si $v = u$ y $u\backslash A = \{u\}$ si $u \parallel v$ o bien $u < v$.

Ahora estamos en condiciones de establecer el resultado anunciado. Puesto que la teoría de cómputos necesarios para la estabilización, tal como ha sido formulada por Middeldorp [Mid97] se refiere a SRTs, formulamos nuestro resultado asumiendo esta limitación.

Teorema 7.3.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT cuasi ortogonal. Sea $t \in \mathcal{T}(\Sigma, V)$ un término no estable y $u \in O_{\mathcal{R}}(t)$. Si $u \in \mathcal{I}(t[\Omega]_u)$, entonces $t|_u$ es un redex necesario para estabilizar t.*

Demostración. Si $u = \epsilon$, es inmediato porque la cuasi ortogonalidad de $\mathcal{R}$ asegura que las reducciones realizadas sobre ocurrencias situadas estrictamente por debajo de la raíz del redex t no pueden estabilizar el término t: esa clase de reductos de t son todavía redexes. Por tanto, si deseamos estabilizar t, deberemos efectuar algún paso de reducción que contraiga el redex a la ocurrencia ϵ. Por tanto, el redex $t|_\epsilon$ es necesario para estabilizar t.

De acuerdo con esto, podemos asumir que $u > \epsilon$. Dado que t no es estable, podemos suponer que tiene un reducto estable s (de no ser así, t no tiene ningún reducto estable y, por definición, todo redex en t es necesario para estabilizarlo). En ese caso, podemos escribir la derivación $t \to^* s$ como $t \xrightarrow{>\epsilon}{}^* \sigma(l) \to^* s$, para alguna regla de reescritura $l \to r \in R$. Ahora razonamos por contradicción. Si $t|_u$ no es necesario para estabilizar t, podemos asumir que no va a ser contraído (ni él ni ninguno de sus residuos) en la derivación anterior. Se puede formalizar este hecho reemplazando el redex $t|_u$ por la constante $\bullet$. Demostramos, por inducción sobre la longitud n de la derivación $t \xrightarrow{>\epsilon}{}^* \sigma(l)$, que existe una derivación $t[\bullet]_u \xrightarrow{>\epsilon}{}^* \sigma'(l)$ para

alguna sustitución σ'. Adviértase que t no es un redex porque, de ser así, por el Lema 7.3.1, la ocurrencia $u > \epsilon$ no sería un índice. Por tanto, el caso base es $n = 1$. Consideremos el paso de reducción $t \xrightarrow{v} \sigma(l)$, donde $v > \epsilon$ es la ocurrencia de un redex $\sigma''(l')$ de alguna regla $l' \to r' \in R$. Entonces tenemos que $t[\sigma''(r')]_v = \sigma(l)$. Por el Lema 2.3.4, $\sigma(l)[\bullet]_u = \sigma'(l)$ para alguna $\sigma' \in \mathit{Subst}(\Sigma \cup \{\bullet\}, V)$. Puesto que u es un índice de $t[\Omega]_u$, no es posible que $v < u$, pues $t|_u$ es un redex externo. Por tanto, sólo debemos considerar dos casos: $u \parallel v$ y $u < v$. En el primer caso, tenemos $t[\bullet]_u = t[\sigma''(l')]_v[\bullet]_u = t[\bullet]_u[\sigma''(l')]_v$ luego $t[\bullet]_u \xrightarrow{v} \sigma'(l)$. En el segundo caso tendríamos $t[\sigma''(r')]_v[\bullet]_u = t[\bullet]_u$ y, ya que $t[\sigma''(r')]_v = \sigma(l)$, $t[\bullet]_u = \sigma'(l)$ tenemos $t[\bullet]_u \xrightarrow{>\epsilon}{}^* \sigma'(l)$, lo que nos lleva a la conclusión deseada.

Si $n > 1$, escribimos $t \xrightarrow{v} t' \xrightarrow{>\epsilon}{}^* \sigma(l)$. Puesto que u es un índice, no es posible que $v < u$, porque $t|_u$ es un redex externo. Por tanto, sólo debemos considerar dos casos: $u \parallel v$ y $u < v$. En ambos casos, el residuo de la ocurrencia redex $u \in O_{\mathcal{R}}(t)$ es la propia ocurrencia, que sigue siendo una ocurrencia redex $u \in O_{\mathcal{R}}(t')$ de t'. De acuerdo con esto, por la hipótesis de inducción, $t'[\bullet]_u \to^* \sigma'(l)$. Si $u \parallel v$, entonces $t' = t[\sigma''(r')]_v$ y $t[\bullet]_u \xrightarrow{v} t'[\bullet]_u$. Si $u < v$, entonces $t[\bullet]_u = t'[\bullet]_u$. En ambos casos, concluimos el resultado deseado.

Entonces, podemos obtener una secuencia de reducción $t[\bullet]_u \xrightarrow{>\epsilon}{}^* \sigma'(l)$ para alguna sustitution σ'. Por la Proposición 2.5.9, $\omega(t[\bullet]_u) \leq \omega(\sigma'(l)) = \Omega$. Por tanto, u no es un índice de $t[\Omega]_u$. Ello supone una contradicción. □

Debemos circunscribir el enunciado del Teorema 7.3.3 a términos no estables.

Ejemplo 7.3.4 *Sea el SRT $\mathcal{R}$ del Ejemplo 7.1.3 y el término estable* `g(f(a))`*. El redex* `f(a)` *está situado sobre un índice puesto que* $\omega($`g(`$\bullet$`)`$) =$ `g(`$\bullet$`)`*. No obstante,* `f(a)` *no es necesario para estabilizar* `g(f(a))` *pues dicho término ya es estable.*

Los índices *no* cubren todas las ocurrencias de redexes necesarios para estabilizar términos no estables, como muestra el siguiente ejemplo.

Ejemplo 7.3.5 *Sea el SRT $\mathcal{R}$*

```
g(a,x) → x
b      → a
```

y el conjunto de todas las derivaciones estabilizantes de `t = g(a,b)`*:*

1. `g(a,`$\underline{\texttt{b}}$`)` $\to$ $\underline{\texttt{g(a,a)}}$ $\to$ `a`*, y*
2. $\underline{\texttt{g(a,b)}}$ $\to$ $\underline{\texttt{b}}$ $\to$ `a`

Por tanto, el redex `b` *a la ocurrencia* 2 *de* `g(a,b)` *es necesario para estabilizar* `t`*. Sin embargo,* 2 *no es un índice:*

$\underline{\texttt{g(a,}\bullet\texttt{)}}$ $\to_\Omega \Omega$

Observación 7.3.6 *El Ejemplo 7.3.5 también muestra que los redexes necesarios para estabilización no siempre son externos (incluso para SRTs ortogonales).*

Finalmente, no es posible extender el Teorema 7.3.3 para aplicarlo a SRTs débilmente ortogonales, tal como evidencia el siguiente ejemplo.

Ejemplo 7.3.7 *Sea el SRT $\mathcal{R}$ del Ejemplo 7.3.2. Tenemos las siguientes derivaciones de estabilización para el término $t =$* `pred(succ(pred(x)))`*:*

`pred(succ(pred(x)))` $\rightarrow$ `pred(x)`

y

`pred(succ(pred(x)))` $\rightarrow$ `pred(x)`

La primera de ellas revela que t no es estable. Tal como se muestra en el Ejemplo 7.3.2, tanto la ocurrencia ϵ como la ocurrencia 1 son índices. Sin embargo, ni $t|_\epsilon$ ni $t|_1$ son necesarios para estabilizar t.

Middeldorp ha demostrado que todo término no estable contiene un redex necesario para su estabilización. Su resultado es válido para SRTs ortogonales. Middeldorp justificó que dicho resultado no es aplicable SRTs cuasi ortogonales mediante el siguiente ejemplo:

Ejemplo 7.3.8 *Sea el SRT cuasi ortogonal $\mathcal{R}$ [Mid97]:*

`T or x` $\rightarrow$ `T`

`x or T` $\rightarrow$ `T`

El término $t =$ `(T or T) or (T or T)` *no es estable. Sin embargo, ni la ocurrencia* 1 *ni la ocurrencia* 2 *del redex* `T or T` *son necesarias para estabilizar el término.*

Advviértase que el SRT del Ejemplo 7.3.8 no es fuertemente secuencial, como demuestra la forma Ω-normal Ω `or` Ω, que no tiene ningún índice.

Empleando el Teorema 7.3.3, podemos refinar un poco más los resultados de Middeldorp (en particular el Teorema 2.5.4).

Teorema 7.3.9 *Sea $\mathcal{R}$ un SRT cuasi ortogonal y fuertemente secuencial. Todo término no estable contiene un redex necesario para su estabilización.*

DEMOSTRACIÓN. Sea s un término no estable y t la forma Ω-normal obtenida al reemplazar cada ocurrencia de un redex en t por símbolos Ω. Debido a la secuencialidad fuerte, t tiene un índice u y, por el Teorema 7.3.3, éste apunta a un redex $s|_u$ necesario para estabilizar s. □

Ejemplo 7.3.10 *(Continuación del Ejemplo 7.3.8) El SRT cuasi ortogonal*

```
T or x → T
T or T → T
F or T → T
```

es fuertemente secuencial (ya que es normal por la izquierda, y los SRT cuasi ortogonales y normales por la izquierda son fuertemente secuenciales, de acuerdo con [Toy92]). La ocurrencia 1 *es un índice de* Ω `or` Ω*:* $\omega(\bullet$ `or` $\Omega) = \bullet$ `or` Ω*. Por tanto, la ocurrencia* 1 *del redex* `T or T` *en el término* `(T or T) or (T or T)` *es necesaria para estabilizar el término. Adviértase que la ocurrencia* 2 *del mismo redex no es necesaria para estabilizar el término t del Ejemplo 7.3.8, porque la derivación*

`(T or T) or (T or T) → T or (T or T) → T`

estabiliza el término inicial sin reducir dicha ocurrencia del redex. Pero observemos que la ocurrencia 2 *no es un índice puesto que* $\omega(\Omega$ `or` $\bullet) = \Omega$*.*

Una consecuencia inmediata del Teorema 7.3.3 es la siguiente:

Teorema 7.3.11 *Sea* $\mathcal{R}$ *un SRT cuasi ortogonal y t un término estabilizable. Sea* $t = t_1 \to t_2 \to \cdots$ *una secuencia de reducción infinita tal que, para todo* $i \geq 1$*, existe un* $j \geq i$ *tal que* $t_j \xrightarrow{u} t_{j+1}$ *y* $u \in \mathcal{I}(t_j\,[\Omega]_u)$*. Existe un* $p \geq 1$ *tal que* t_p *es estable.*

DEMOSTRACIÓN. Si no existe ningún p tal que t_p es estable, por el Teorema 7.3.3, la secuencia de reducción contiene un número infinito de pasos de reducción necesarios para estabilizar t; por tanto, contiene un número infinito de pasos de contracción de conjuntos de redexes necesarios para estabilización, lo cual contradice el Teorema 2.5.7. □

En términos de estrategias de reducción de índices, obtenemos el siguiente corolario.

Teorema 7.3.12 *Toda estrategia de reducción de índices es estabilizante para SRTs cuasi ortogonales y fuertemente secuenciales.*

El Teorema 7.3.12 justifica la selección de redexes en la primera derivación del Ejemplo 7.1.3. Escogimos reducir un redex situado sobre un índice que, debido a que es necesario para estabilizar el término inicial, realmente contribuye a ésta. Por último, nos planteamos la siguiente pregunta:

> *¿Es posible encontrar una clase de SRTs más amplia que la de los SRTs fuertemente secuenciales donde sea posible asegurar que todo término* no estable *contiene un redex situado sobre una ocurrencia índice?*

Esta parece una condición menos exigente que la de que todo término que *no sea una forma normal* contenga un redex situado sobre un índice e invita a confiar en la existencia de dicha clase de SRTs.

Si aceptamos los términos frágiles como aproximaciones suficientes de las formas no estables, la pregunta anterior no tiene una respuesta afirmativa. Esta afirmación es consecuencia del siguiente resultado.

Teorema 7.3.13 *Si $\mathcal{R}$ es un SRT tal que toda forma Ω-normal frágil contiene un índice, entonces $\mathcal{R}$ es fuertemente secuencial.*

DEMOSTRACIÓN. Por contradicción. Supongamos que $\mathcal{R}$ no es fuertemente secuencial. Entonces existe una forma Ω-normal t sin ocurrencias de índices. Sea $t = C[t_1, \ldots, t_n]$ la descomposición única de t en un contexto rígido $C[\,]$ y términos frágiles $t_1, \ldots, t_n$ (Proposición 2.5.12). De entre $t_1, \ldots, t_n$ debe existir uno, t_i, que sea una forma Ω-normal (frágil). Supongamos que $t|_u = t_i$. Por hipótesis, existe un índice $v \in \mathcal{I}(t_i)$. Pero, por la Proposición 2.5.18, se tiene que $u.v \in \mathcal{I}(t)$, lo cual contradice la suposición inicial de inexistencia de índices en t. □

7.4 Condiciones de definición de estrategias sensibles al contexto basadas en reducción de índices

El Teorema 7.3.12 junto con el Teorema 4.3.8 sugieren la posibilidad de utilizar secuencias de reducción de índices para μ-normalizar términos, de forma análoga a como sucede en la reescritura estándar. Sin embargo, si empleamos la *rsc*, los índices considerados deberán ser *reemplazantes*. Los SRTs fuertemente secuenciales nos aseguran que podemos definir secuencias de reducción de índices aptas para normalizar términos porque cualquier término que no está en forma normal contiene un redex situado sobre una ocurrencia índice. Sin embargo, nada garantiza, a priori, que una vez impuestas las restricciones de reemplazamiento se verificará todavía la propiedad que nos permitirá definir estrategias de *reducción de índices reemplazantes*: que todo término que no se encuentra en forma μ-normal tenga un redex situado sobre un índice *reemplazante*. Afortunadamente, tal como probamos a continuación, podemos establecer condiciones para demostrar esta propiedad.

Notación 7.4.1 *Dado que nos interesa distinguir entre índices reemplazantes y no reemplazantes de un término t, denotaremos los primeros escribiendo $\mathcal{I}^\mu(t)$, es decir, $\mathcal{I}^\mu(t) = \mathcal{I}(t) \cap O^\mu(t)$. Escribiremos $\overline{\mathcal{I}^\mu}(t) = \mathcal{I}(t) \cap \overline{O^\mu}(t)$ para referirnos a los no reemplazantes.*

Antes de establecer el resultado mencionado anteriormente, necesitamos demostrar algunos lemas auxiliares.

Lema 7.4.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Sea $t_1 \stackrel{u_1}{\rightarrow}_\Omega t_2 \rightarrow_\Omega \cdots \rightarrow_\Omega t_n \stackrel{u_n}{\rightarrow}_\Omega t_{n+1}$ una Ω-derivación. Para todo i, j, $1 \leq i \leq n$, $j \leq i$, se tiene que $O(t_i) \subseteq O(t_j)$ y $O^\mu(t_i) \subseteq O^\mu(t_j)$.*

DEMOSTRACIÓN. Por inducción sobre la longitud n de la derivación. Si $n = 1$, entonces $t_1 = C[s]_{u_1}$, $t_2 = C[\Omega]_{u_1}$ y $s \uparrow l$ para algún $l \in L_\Omega(\mathcal{R})$. Claramente, $O(t_2) \subseteq O(t_1)$ y, dado que $O^\mu(t_1)$ es un cono en $(O(t_1), \leq)$, $O^\mu(t_2) \subseteq O^\mu(t_1)$. Si $n > 1$, empleando la hipótesis de inducción obtenemos el resultado deseado. □

El siguiente resultado establece que, al considerar un término frágil, podemos definir una secuencia de Ω-reducción para él que sólo realiza pasos de Ω-reducción sobre subtérminos compatibles situados sobre ocurrencias reemplazantes.

Lema 7.4.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ es un término frágil, entonces existe una derivación*

$$t = t_1 \stackrel{u_1}{\rightarrow}_\Omega t_2 \rightarrow_\Omega \cdots \rightarrow_\Omega t_n \stackrel{u_n}{\rightarrow}_\Omega \Omega$$

tal que $u_i \in O^\mu(t_i)$ para todo i, $1 \leq i \leq n$.

DEMOSTRACIÓN. Siendo t frágil, tenemos que $\omega(t) = \Omega$. La demostración se realiza por inducción en la longitud de una derivación cualquiera

$$t = t_1 \stackrel{u_1}{\rightarrow}_\Omega t_2 \rightarrow_\Omega \cdots \rightarrow_\Omega t_n \stackrel{u_n}{\rightarrow}_\Omega \Omega$$

que Ω-reduce t a Ω.

Si $n = 1$, se tiene $t \stackrel{\epsilon}{\rightarrow}_\Omega \Omega$. Puesto que $\epsilon \in O^\mu(t)$, la conclusión es inmediata.

Si $n > 1$, consideremos el primer paso $t_i \stackrel{u_i}{\rightarrow}_\Omega t_{i+1}$ tal que $u_i \in O^\mu(t_i)$. Puesto que $\epsilon \in \{u_1, \ldots, u_n\}$, dicho paso debe tener lugar en algún momento. Sea $U_i = \{u_j \mid u_i < u_j.\ 1 \leq j < i\}$. Este conjunto está bien definido porque, de acuerdo con el Lema 7.4.2, $u_i \in O(t_j)$ para todo $j < i$. Consideramos dos casos para U_i: que esté vacío y que no lo esté.

Si $U_i = \emptyset$, puesto que $u_j \in \overline{O^\mu}(t_j)$ y $u_i \in O(t_j)$ para $1 \leq j < i$, por la Proposición 3.2.8, $u_i \parallel u_j$ para todo $1 \leq j < i$. Entonces podemos realizar un paso de Ω-reducción directamente sobre $t_1|_{u_i}$. Por tanto, construimos la derivación

$$t = t_1 \stackrel{u_i}{\rightarrow}_\Omega t'_1 \stackrel{u_1}{\rightarrow}_\Omega t'_2 \rightarrow_\Omega \cdots \rightarrow_\Omega t'_{i-1} \stackrel{u_{i-1}}{\rightarrow}_\Omega t_{i+1} \stackrel{u_{i+1}}{\rightarrow}_\Omega \cdots \rightarrow_\Omega t_n \stackrel{u_n}{\rightarrow}_\Omega \Omega,$$

donde $t'_j = t_j[\Omega]_{u_i}$ para todo j, $1 \leq j \leq i-1$. Claramente, t'_1 es frágil. Aplicamos la hipótesis de inducción a la derivación $t'_1 \rightarrow^*_\Omega \Omega$ que consta de $n-1$ pasos y de ahí concluimos el resultado deseado.

Si $U_i = \{u_i.v_1, \ldots, u_i.v_m\}$ y $m > 0$, dado que el Lema 7.4.2 asegura que $u_i \in O(t)$, consideremos $s = t|_{u_i}$ y el esquema de redex $l \in L_\Omega(\mathcal{R})$ tal que $t_i|_{u_i} \uparrow l$. Demostramos que $s \uparrow l$ por contradicción. Las reducciones realizadas sobre ocurrencias pertenecientes al conjunto $\{u_1, \ldots, u_n\} \backslash U_i$ son disjuntas de las que consideramos (puesto que asumimos que $u_j \in \overline{O^\mu}(t_j)$ para todo $j < i$, no puede darse el caso $u_j < u_i$), y no tienen interés aquí. Dado que $t_i|_{u_i} \uparrow l$ y $u_i \in O^\mu(t_i)$, las reducciones anteriores (realizadas sobre ocurrencias no reemplazantes contenidas en U_i) no pueden modificar

el símbolo que etiqueta la raíz de $t|_{u_i}$. Por este motivo tenemos $s = C[s_1, \ldots, s_p]$ y $l = C[l_1, \ldots, l_p]$ para algún contexto maximal no vacío $C[\,]$. Si s y l no son compatibles, existe un k, $1 \leq k \leq p$ tal que s_k y l_k no son compatibles. Por tanto debemos tener $root(l_k) \neq \Omega$. De acuerdo con esto, puesto que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$, tenemos que $w_k \in O^{\mu}(l)$ si $l|_{w_k} = l_k$. No obstante, ninguna Ω-reducción realizada por debajo de la raíz del subtérmino s_k sobre posiciones indicadas por ocurrencias pertenecientes a U_i puede hacer que s_k sea compatible con l_k, porque de acuerdo con la Proposición 3.2.3, tenemos $w_k \in O^{\mu}(s)$ y toda Ω-reducción tiene lugar sobre ocurrencias no reemplazantes (como son las de U_i) y nuestra elección de s_k (y por tanto de w_k) es arbitraria. Sin embargo, esto significa que $t_i|_{u_i}$ no será compatible con l, lo cual nos lleva a una contradicción. Luego podemos eliminar los pasos de Ω-reducción $t_j \xrightarrow{u_j}_{\Omega} t_{j+1}$ para $u_j \in U_i$ porque son subsumidos por el paso de reducción $t_i \xrightarrow{u_i}_{\Omega} t_{i+1}$ y por tanto resultan inútiles. Entonces obtenemos una derivación más corta (dado que asumíamos $m > 0$) y aplicando la hipótesis de inducción concluimos la demostración. □

La siguiente proposición establece que, al considerar un término frágil, el conjunto de índices no reemplazantes de dicho término está vacío.

Proposición 7.4.4 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y sea $\mu \in M_{\Sigma}$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ es un término frágil, entonces $\overline{\mathcal{I}^{\mu}}(t) = \emptyset$.*

DEMOSTRACIÓN. Si $O_{\Omega}(t) = \emptyset$, es inmediato porque $\mathcal{I}(t) = \emptyset$ y en ese caso $\overline{\mathcal{I}^{\mu}}(t) = \mathcal{I}(t) \cap \overline{O^{\mu}}(t) = \emptyset$. Supongamos que $O_{\Omega}(t) \neq \emptyset$ y que existe un índice no reemplazante $v \in \overline{\mathcal{I}^{\mu}}(t)$. Puesto que $\omega(t) = \Omega$, por el Lema 7.4.3, existe una derivación $t = t_1 \xrightarrow{u_1}_{\Omega} t_2 \to_{\Omega} \cdots \to_{\Omega} t_n \xrightarrow{u_n}_{\Omega} \Omega$, tal que $u_i \in O^{\mu}(t_i)$ para todo $1 \leq i \leq n$. Demostramos que dicha derivación también puede eliminar $\bullet$ de $t[\bullet]_v$ contradiciendo de esta manera la asunción de que v es un índice de t. Sea u_i, para algún i, $1 \leq i \leq n$ la primera ocurrencia comparable con v, es decir, $\forall j < i$, $u_j \parallel v$ y $u_i < v$ (dado que $t|_v = \Omega$, no es posible que se dé el caso $u_i > v$). Dicha ocurrencia u_i debe existir puesto que, en otro caso, la derivación anterior no podría convertir t en Ω. Puesto que $u_j \parallel v$ para $1 \leq j < i$, la presencia de $\bullet$ en $t_1[\bullet]_v, \ldots, t_{i-1}[\bullet]_v$ no interfiere en los pasos de Ω-reducción correspondientes. Puesto que $t|_v = \Omega$, tenemos que $t[\bullet]_v = t_1[\bullet]_v \xrightarrow{u_1}_{\Omega} t_2[\bullet]_v \to_{\Omega} \cdots \to_{\Omega} t_{i-1}[\bullet]_v \xrightarrow{u_{i-1}}_{\Omega} t_i[\bullet]_v$. Sabemos que $s = t_i|_{u_i}$ verifica $s \uparrow l$ para algún $l \in L_{\Omega}(\mathcal{R})$. Consideremos que $v = u_i.w$. Si $w \notin O(l)$, existe un $w' < w$ tal que $l|_{w'} = \Omega$. Si $w \in O(l)$, puesto que $s \uparrow l$ se tiene $sprefix_s(w) = sprefix_l(w)$. Puesto que $u_i \in O^{\mu}(t_i)$ y $v \in \overline{O^{\mu}}(t_i)$, por la Proposición 3.2.1 tenemos que $w \in \overline{O^{\mu}}(s)$. Por tanto, puesto que $sprefix_s(w) = sprefix_l(w)$, por la Proposición 3.2.3 tenemos $w \in \overline{O^{\mu}}(l)$. Puesto que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$, tenemos $l|_w = \Omega$.

De acuerdo con esto, siempre tenemos una $w' \leq w$ tal que $l|_{w'} = \Omega$. Luego $s[\bullet]_w \uparrow l$, es decir, $t_i[\bullet]_v|_{u_i} \uparrow l$, y por tanto v no es un índice de t. □

Por desgracia, este resultado no puede extenderse a términos que no sean frágiles (es decir, a términos fuertemente estables).

Ejemplo 7.4.5 *Consideremos el SRT del Ejemplo 1.1.2 y la función de reemplazamiento μ definida allí. El término $t = \Omega::\Omega$ es fuertemente estable. Sin embargo, tanto la ocurrencia 1 como la ocurrencia 2 son índices de t, pero $1,2 \notin O^{\mu}(t)$. De esta manera tenemos $\overline{\mathcal{I}^{\mu}}(t) \neq \emptyset$.*

La Proposición 7.4.4 tiene una consecuencia inmediata.

Teorema 7.4.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y sea $\mu \in M_{\Sigma}$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ es frágil, entonces $\mathcal{I}^{\mu}(t) = \mathcal{I}(t)$.*

DEMOSTRACIÓN. Si existe un índice no reemplazante en t, entonces $\overline{\mathcal{I}^{\mu}}(t) \neq \emptyset$, lo cual contradice la Proposición 7.4.4. □

Observación 7.4.7 *Es interesante observar la siguiente consecuencia simple de los resultados presentados: la Proposición 2.5.10, nos permite afirmar que si un término t no es estable, tampoco es fuertemente estable. Por tanto, t es frágil y el Teorema 7.4.6 implica que en derivaciones de estabilización (fuerte) que partan de t,* sólo *se reducen (en su caso) redexes situados sobre índices $\mu_{\mathcal{R}}^{com}$-reemplazantes (incluso sin considerar de forma explícita las restricciones de reemplazamiento), porque, de hecho, no existen índices ocupando posiciones que no son $\mu_{\mathcal{R}}^{com}$-reemplazantes.*

Esto muestra que las restricciones sensibles al contexto pueden emplearse para acotar de forma más precisa los redexes que deben ser contraídos para estabilizar un término: sólo necesitamos comprobar si los redexes reemplazantes *están situados sobre índices.*

El siguiente resultado establece que las restricciones de reemplazamiento siempre permiten capturar algún índice de un término que no es una forma μ-normal. En otras palabras, los SRTEs fuertemente secuenciales tienen la propiedad de que *todo término que no es una forma μ-normal tiene un índice reemplazante.*

Teorema 7.4.8 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y fuertemente secuencial. Sea $\mu \in M_{\Sigma}$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ es una forma Ω-normal tal que $O_{\Omega}^{\mu}(t) \neq \emptyset$, entonces $\mathcal{I}^{\mu}(t) \neq \emptyset$.*

DEMOSTRACIÓN. Apoyándonos en la Proposición 2.5.12, realizamos la demostración por inducción sobre la estructura del (único) contexto rígido $C[\,]$ tal que $t = C[t_1, \ldots, t_n]$ y que $t_1, \ldots, t_n$ son términos frágiles. Adviértase que, debido a la secuencialidad fuerte de $\mathcal{R}$, $\mathcal{I}(t) \neq \emptyset$.

(1) El caso base es $C[\,] = \Box$. Entonces t es frágil y, por la Proposición 7.4.4, todos sus índices son reemplazantes. Por tanto, llegamos a la conclusión deseada.

(2) En el paso de inducción, supondremos que $C[\,]$ no es el contexto vacío. Por tanto, t no es frágil. Puesto que $O^{\mu}_{\Omega}(t) \neq \emptyset$ y $C[\,]$ no contiene ocurrencias de Ω, debe existir algún i, $1 \leq i \leq n$ tal que $t_i = t|_u$ es una forma Ω-normal que cumple $u \in O^{\mu}(t)$, y $O^{\mu}_{\Omega}(t_i) \neq \emptyset$. Por la hipótesis de inducción, existe un índice reemplazante $v \in \mathcal{I}^{\mu}(t_i)$. Por la Proposición 2.5.18, $u.v \in \mathcal{I}(t)$. Por la Proposición 3.2.1, $u.v \in O^{\mu}(t)$. Con esto concluimos la demostración. □

Necesitamos imponer la condición $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ para asegurar el resultado anterior.

Ejemplo 7.4.9 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 1.1.2, la función de reemplazamiento canónica $\mu = \mu^{com}_{\mathcal{R}}$ y el Ω-término $t =$* `first(Ω,Ω)`*. Si μ(*`first`*) $= \{2\}$ entonces $O^{\mu}_{\Omega}(t) = \{2\}$, pero $2 \notin \mathcal{I}(t)$, por tanto $\mathcal{I}^{\mu}(t) = \emptyset$.*

Este ejemplo también muestra que, en general, las restricciones de reemplazamiento no caracterizan completamente a los índices: si tomamos $\mu = \mu^{com}_{\mathcal{R}}$, obtenemos que $2 \in O^{\mu}_{\Omega}(t)$, pero $2 \notin \mathcal{I}(t)$ (aunque ahora $\mathcal{I}^{\mu}(t) = \{1\}$).

El Teorema 7.4.8 permite refinar el Teorema 7.3.9, que también complementa el Teorema 7.2.3.

Teorema 7.4.10 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT cuasi ortogonal y fuertemente secuencial. Sea $\mu \in M_{\Sigma}$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Todo término no estable contiene un redex μ-reemplazante necesario para su estabilización.*

DEMOSTRACIÓN. Sea s un término no estable. Por el Teorema 4.3.8, s no es una forma μ-normal. Por tanto, contiene, al menos, un redex reemplazante. Si cambiamos todos los redexes externos de s por ocurrencias de Ω, obtenemos una forma Ω-normal t y $O^{\mu}_{\Omega}(t) \neq \emptyset$. Por el Teorema 7.4.8, existe un índice reemplazante $u \in \mathcal{I}^{\mu}(t)$. Por el Teorema 7.3.3, dicho índice reemplazante (en t) apunta a un redex necesario para estabilizar s. Puesto que $sprefix_s(u) = sprefix_t(u)$, por la Proposición 3.2.3, $s|_u$ es un redex reemplazante y necesario para estabilizar s. □

7.5 μ-estrategias de reducción de índices

Diremos que una μ-estrategia $\mathbb{H}_{\mathcal{R}(\mu)}$ para un SRT $\mathcal{R}$ lineal por la izquierda y fuertemente secuencial es una μ-estrategia de *reducción de índices* si siempre reduce redexes (μ-reemplazantes) situados sobre índices.

Observación 7.5.1 *De acuerdo con los resultados presentados hasta aquí, sólo podemos asegurar que una μ-estrategia de reducción de índices para $\mathcal{R}$ está bien definida si $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. En este caso, el Teorema 7.4.8 garantiza que todo término que no es una forma μ-normal contiene un redex μ-reemplazante situado sobre un índice.*

A continuación, establecemos las propiedades fundamentales de las μ-estrategias de reducción de índices.

Teorema 7.5.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT cuasi ortogonal y fuertemente secuencial. Sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Toda μ-estrategia de reducción de índices para $\mathcal{R}$ es estabilizante.*

DEMOSTRACIÓN. Sea t un término estabilizable. Si al aplicar la estrategia sobre t, la computación termina obteniendo una forma μ-normal $\bar{s}$, por el Teorema 4.3.8, $\bar{s}$ es estable. Si la estrategia genera una secuencia de μ-reducción infinita $t = t_1 \hookrightarrow t_2 \hookrightarrow \cdots$, por el Teorema 7.3.11, existe un $p \geq 1$ tal que t_p es estable. □

Tal como se discute en [Mid97], las estrategias de estabilización no tienen porqué ser normalizantes.

Ejemplo 7.5.3 *Consideremos el SRT $\mathcal{R}$ [Mid97]:*

```
f(x) → g(a)
b    → b
```

y la estrategia $\mathbb{IF}_{\mathcal{R}}$ que siempre selecciona el (único) redex externo del término t al cual se aplica, salvo en el caso del término `g(f(b))`*, donde resulta seleccionado el redex* `b`*. Está claro que $\mathbb{IF}_{\mathcal{R}}$ es estabilizante. No obstante, no es normalizante puesto que del término* `g(f(b))` *parte la siguiente secuencia de $\mathbb{IF}_{\mathcal{R}}$-reducción infinita:*

$$\texttt{g(f(}\underline{\texttt{b}}\texttt{))} \rightarrow_{\mathbb{IF}_{\mathcal{R}}} \texttt{g(f(}\underline{\texttt{b}}\texttt{))} \rightarrow_{\mathbb{IF}_{\mathcal{R}}} \cdots$$

que no obtiene la forma normal `g(g(a))` *de* `g(f(b))`*, que puede obtenerse, en cambio, mediante la siguiente derivación:*

$$\texttt{g(}\underline{\texttt{f(b)}}\texttt{)} \rightarrow \texttt{g(g(a))}$$

De hecho, el Ejemplo 7.5.3 también muestra que el carácter estabilizante de una μ-estrategia no asegura que sea μ-normalizante. Aún más, a diferencia de la estabilización (Teorema 7.3.12), las secuencias de reducción de índices arbitrarias no tienen porqué ser μ-normalizantes.

Ejemplo 7.5.4 *Consideremos el SRT $\mathcal{R}$:*

```
f(x,a) → c(x,a)
b      → b
```

y la función de reemplazamiento μ tal que $\mu(\texttt{f}) = \mu(\texttt{c}) = \{2\}$. Observemos que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. La secuencia de reducción de índices

$$\texttt{c(}\underline{\texttt{b}}\texttt{,f(x,a))} \rightarrow \texttt{c(}\underline{\texttt{b}}\texttt{,f(x,a))} \rightarrow \cdots$$

no obtiene la forma μ-normal `c(b,c(x,a))` *que, en cambio, puede alcanzarse mediante la siguiente secuencia de reducción de índices:*

`c(b,`$\underline{\texttt{f(x,a)}}$`)` $\rightarrow$ `c(b,c(x,a))`

Además, a pesar de la relación que establece la propiedad de independencia de contexto entre el carácter estabilizante y el μ-normalizante de las μ-estrategias (Teorema 7.2.6), podemos probar que no toda estrategia (determinista) de reducción de índices es independiente del contexto.

Ejemplo 7.5.5 *Sea la estrategia de reducción de índices $\mathbb{IF}_{\mathcal{R}}$ para un SRT $\mathcal{R}$ que, dado el conjunto $I(t) = \{u_1, \ldots, u_n\}$ de ocurrencias de redexes situados sobre índices de un término t, selecciona el del extremo izquierdo (según el orden $\leq_L$) si n es impar y el del extremo derecho si n es par.*

Esta estrategia no es independiente del contexto. Por ejemplo, consideremos el SRT $\mathcal{R}$:

`f(b,c)` $\rightarrow$ `d`
`a` $\rightarrow$ `d`

y el término $t =$ `f(f(a,a),a)`*. El término t es estable pero, debido a que*

`f(f(`$\underline{\texttt{a}}$`,a),a)` $\rightarrow_{\mathbb{IF}_{\mathcal{R}}}$ `f(f(d,a),a)`

mientras que

`f(a,`$\underline{\texttt{a}}$`)` $\rightarrow_{\mathbb{IF}_{\mathcal{R}}}$ `f(a,d)`

concluimos que $\mathbb{IF}_{\mathcal{R}}$ no es independiente del contexto.

Por tanto, no es posible utilizar los Teoremas 7.5.2 y 7.2.6 para concluir que toda μ-estrategia de reducción de índices es μ-normalizante.

No obstante, podemos establecer dicho resultado sin necesidad del Teorema 7.2.6.

Teorema 7.5.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT cuasi ortogonal y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Sea t un término μ-normalizable. Si $A^\mu : t = t_1 \hookrightarrow t_2 \hookrightarrow \cdots$ es una secuencia de μ-reducción tal que, para todo $i \geq 1$, si t_i no es una forma μ-normal, se verifica que $\exists j \geq i$ tal que $t_j \overset{u}{\hookrightarrow} t_{j+1}$ y $u \in \mathcal{I}^\mu(t_j[\Omega]_u)$, entonces existe un $p \geq 1$ tal que t_p es una forma μ-normal de t.*

DEMOSTRACIÓN. Puesto que t es μ-normalizable, sea $\bar{s}$ una forma μ-normal de t. Realizamos la demostración por inducción sobre la estructura de $C[\,] = CMR^\mu(\bar{s})$, el contexto reemplazante máximo de $\bar{s}$. Adviértase que, puesto que los SRTs cuasi ortogonales son confluentes, por la Proposición 4.3.11, $C[\,]$ no depende de la forma μ-normal concreta $\bar{s}$ que consideremos.

Si $C[\,]$ es una constante o una variable, debe ser $C[\,] = \bar{s}$ y $\bar{s}$ es una forma normal. Por confluencia, $\bar{s}$ es la única forma normal. Dado que la confluencia asegura que $C[\,]$ es único para toda forma μ-normal de t y $C[\,]$ es una constante o una variable, la forma μ-normal es única también. Si la secuencia A^μ no la obtiene, de acuerdo con

las premisas del teorema, A^μ es una secuencia de reducción infinita. Por el Teorema 7.3.11, la secuencia de reducción es estabilizante y existe un q tal que $s = t_q$, es una forma estable de t. Por confluencia, existe un término s' tal que $\bar{s} \rightarrow^* s'$ y $s \rightarrow^* s'$. Puesto que $\bar{s}$ es una forma normal, debe tenerse que $\bar{s} = s'$. Dado que s es estable, el símbolo que etiqueta su raíz no puede cambiarse. Como $\bar{s}$ es una constante o una variable, tenemos que $\bar{s} = s' = s$, contradiciendo la asunción de que A^μ no contiene una forma μ-normal.

Sea $C[\,] = f(C_1[\,], \ldots, C_k[\,]) = CMR^\mu(\bar{s})$. Por la Proposición 4.3.11, el símbolo f es un prefijo común a toda forma μ-normal de t. Dado que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$, $\bar{s}$ es una forma estable de t. Luego t es estabilizable. Si suponemos, de nuevo, que A^μ no contiene una forma μ-normal, la secuencia A^μ es infinita y, por el Teorema 7.3.11, existe un $q \geq 1$ tal que $t_q = s = f(s_1, \ldots, s_k)$ es una forma estable de t. Por confluencia, s también tiene una forma μ-normal que, por la Proposición 4.3.11, tiene el mismo contexto reemplazante máximo $C[\,]$. Puesto que s es estable, sabemos que cada s_i tiene una forma μ-normal, para cada $i \in \mu(f)$. Por otro lado, cada s_i, para $i \in \mu(f)$ tiene asociada una secuencia de μ-reducción A_i^μ obtenida a partir de la subsecuencia de μ-reducción original A^μ que parte de s de la forma obvia. Por la Proposición 2.5.17, los pasos de reducción de índices (reemplazantes) realizados sobre un término cualquiera de la subsecuencia de reducción de índices que arranca en s son pasos de reducción de índices sobre cualquier subtérmino inmediato para cualquier $i \in \mu(f)$. Por la Proposición 3.2.1 se trata de índices reemplazantes. Es fácil ver que, si A^μ no obtiene una forma normal, es una secuencia infinita y entonces alguna de las subsecuencias A_i^μ también es infinita y verifica las condiciones del enunciado del teorema. Por tanto, aplicamos la hipótesis de inducción para concluir que cada secuencia de reducción que parte de s_i obtiene una forma μ-normal $\bar{s}_i'$ de s_i para cada $i \in \mu(f)$ lo cual contradice la hipótesis de que la secuencia A^μ no obtiene una forma μ-normal. Puesto que s es estable y los pasos de reducción de índices que arrancan de s_i para un $i \in \mu(f)$ son independientes de los realizados sobre cualquier otro $j \in \mu(f)$, $i \neq j$, concluimos que la secuencia de reducción de índices completa μ-normaliza t, obteniendo una forma μ-normal $\bar{s} = f(\bar{s}_1, \ldots, \bar{s}_k)$, donde $\bar{s}_i = t_i'$ si $i \notin \mu(f)$ y $\bar{s}_i = \bar{s}_i'$ si $i \in \mu(f)$. □

Este teorema tiene una consecuencia inmediata.

Teorema 7.5.7 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT cuasi ortogonal y fuertemente secuencial y sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Toda μ-estrategia de reducción de índices es μ-normalizante.*

También podemos establecer un resultado menos potente pero aplicable a una clase más amplia de SRTs.

Teorema 7.5.8 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT lineal por la izquierda, fuertemente secuencial y uniformemente convergente a la raíz. Sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$ y t*

un término normalizable. Cualquier μ-estrategia de reducción de índices obtiene una forma μ-normal de t.

DEMOSTRACIÓN. Puesto que t es normalizante, el Teorema 2.5.21 asegura que no existen secuencias de reducción infinitas que, partiendo de t, sólo contraen redexes situados sobre índices. Dado que la μ-estrategia de reducción de índices sólo reduce redexes situados sobre índices, la computación se detendrá obteniendo una forma μ-normal de t. □

7.6 Trabajos relacionados

El Teorema 7.2.5 supone un avance respecto al uso de las restricciones de reemplazamiento sintácticas que realiza Maranget [Mar90] en un marco de reducción de grafos para definir derivaciones óptimas mediante secuencias de reducción paralela. Debido a su circunscripción a dicho marco, Maranget limita su atención a una clase muy pequeña de SRTs ortogonales, y a funciones de reemplazamiento tales que se verifiquen las siguientes condiciónes: $V_a(r) \subseteq V_a(l)$ y $V_f(r) \subseteq V_f(l)$ para toda regla $l \to r$ del SRT (ver la Sección 3.5). Por ejemplo, el programa

```
if(true,x,y)  → x
if(false,x,y) → y
```

sólo admite trabajar con una función de reemplazamiento que satisfaga $\mu(\texttt{if}) = \{1,2,3\}$, es decir, que no impone ninguna restricción en los reemplazamientos. Puesto que el Teorema 7.2.5 permite emplear cualquier función de reemplazamiento μ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ y $\mu^{com}_{\mathcal{R}}(\texttt{if}) = \{1\}$, nuestros resultados en este capítulo sobre la necesidad de las reducciones restringidas son más generales (aunque se refieren únicamente a términos no estables). Respecto al trabajo presentado en [OF97], donde se plantea una implementación mediante reescritura de la estrategia E de OBJ y CafeOBJ, y que guarda una estrecha relación con la *rsc*, tampoco se proporciona ningún resultado que permita asegurar el carácter terminante de los cómputos realizados mediante esa estrategia, ni se da indicación alguna acerca de cómo definir la *estrategia local* asociada a cada símbolo para conseguir un comportamiento terminante. Puesto que las estrategias locales pueden verse como una forma de especificar estrategias sensibles al contexto, los resultados presentados en este capítulo y en los posteriores pueden utilizarse para guiar la definición de estrategias locales y para establecer propiedades computacionales de la estrategia E.

Aunque la noción de estrategia sensible al contexto (o de estrategia que incorpore restricciones sintácticas) y las propiedades relevantes que conciernen a su utilización no han sido estudiadas formalmente con anterioridad, la estrategia E, basada en las

llamadas estrategias locales, introducida por los diseñadores de las distintas versiones y derivados del lenguaje OBJ [FGJM85], puede considerarse como un precedente de utilización práctica de estrategias sensibles al contexto.

En el próximo capítulo continuamos el análisis de estas estrategias con el fin de estudiar con más profundidad la relación existente entre las estrategias de reescritura habituales y las estrategias sensibles al contexto. Los resultados que ofrecemos (en cuanto a optimalidad, carácter normalizante, etc.) son, por tanto, aplicables al análisis de los cómputos realizados mediante la estrategia E, por ejemplo.

Capítulo 8

Estrategias genéricas

La definición formal de una estrategia de reescritura $\mathbb{F}_{\mathcal{R}}$ se realiza respecto a un SRT $\mathcal{R}$ concreto. También es interesante disponer de una noción de estrategia *genérica* que pueda aplicarse a *clases* de SRTs. De hecho, éstas han sido empleadas profusamente en reescritura (cuando se habla de estrategias *leftmost-outermost*, *parallel-innermost*, *leftmost-innermost*, etc., en realidad nos estamos refiriendo implícitamente a estrategias genéricas), aunque la noción no parece haberse formalizado explícitamente ni estudiado con mayor profundidad. Dicha noción puede extenderse a la *rsc* de forma inmediata y el problema central que nos planteamos en este capítulo es la definición de estrategias genéricas sensibles al contexto (estrategias genéricas *sc*) a partir de estrategias genéricas estándar y el análisis de las propiedades que aquéllas exhiben en función de las propiedades de que disfrutan éstas.

En la Sección 8.1 definimos y motivamos el concepto de estrategia genérica que desarrollamos en esta tesis. Estudiamos algunas formas de definir estrategias genéricas. En particular, definimos las estrategias genéricas *estructurales* y damos ejemplos bien conocidos de tales estrategias (Sección 8.1.1). En la Sección 8.1.2 definimos el concepto de estrategia genérica *sc*.

En la Sección 8.2 demostramos que la operación de contracción τ_μ introducida en el Capítulo 5 preserva los índices reemplazantes (cuando se aplica sobre términos) y la secuencialidad fuerte de los SRTEs (cuando se aplica sobre éstos). Partiendo de estas observaciones, en las Secciones 8.2.2 y 8.2.3 establecemos una correspondencia entre estrategias genéricas de reducción de índices y estrategias genéricas *sc* de reducción de índices, que denominaremos estrategias genéricas *sc inducidas*.

8.1 Concepto de estrategia genérica

A continuación damos la definición formal de estrategia genérica que consideramos en esta tesis.

Definición 8.1.1 (Estrategia genérica) *Una* estrategia genérica *para una clase CL de SRTs es una función $\mathbb{F}_{CL}$ (o simplemente $\mathbb{F}$) que asocia a cada $\mathcal{R} \in CL$ una estrategia de reescritura $\mathbb{F}_{\mathcal{R}}$.*

Dada una estrategia genérica $\mathbb{F}_{CL}$ para una clase CL de SRTs, decimos que $\mathbb{F}_{\mathcal{R}}$ es una estrategia *concreta* obtenida a partir de $\mathbb{F}$ (o que es una *concreción* de $\mathbb{F}$). Si consideramos $\mathcal{R}$ como un parámetro, la descomposición de una estrategia $\mathbb{F}_{\mathcal{R}}$ para el SRT $\mathcal{R}$ en sus componentes $(\mathbb{P}_{\mathcal{R}}, \mathbb{R}_{\mathcal{R}})$ también puede realizarse con las estrategias genéricas de la forma obvia, derivando en una descomposición $(\mathbb{P}, \mathbb{R})$ de cualquier estrategia genérica $\mathbb{F}$. Decimos que una estrategia genérica $\mathbb{F}_{CL}$ es normalizante, estabilizante, etc., si para todo $\mathcal{R} \in CL$, la concreción $\mathbb{F}_{\mathcal{R}}$ es normalizante, estabilizante, etc.

El interés del concepto de estrategia genérica puede motivarse argumentando tanto sobre los aspectos teóricos como sobre los prácticos:

1. Desde el punto de vista teórico, todas las concreciones de una estrategia genérica para una clase CL comparten ciertas propiedades. Típicamente, serán normalizantes, óptimas, etc., para todo SRT en CL.

2. Desde el punto de vista práctico, la implementación de las concreciones de dichas estrategias emplearán algoritmos y estructuras de datos similares (o idénticas). Por tanto, la complejidad de dichos algoritmos, los recursos requeridos, etc., serán también similares.

De hecho, las estrategias genéricas son utilizadas con frecuencia en reescritura.

Ejemplo 8.1.2 *Consideremos las siguientes "estrategias de reescritura" [Klo92] que, de hecho, son estrategias genéricas para la clase de los SRTs* débilmente ortogonales.

1. *La estrategia (secuencial)* leftmost-innermost *($\mathbb{F}_{li}$) reduce, de entre los redexes más internos, el del extremo izquierdo.*

2. *La estrategia (paralela)* parallel-innermost *($\mathbb{F}_{pi}$) reduce todos los redexes más internos de forma simultánea.*

3. *La estrategia (secuencial)* leftmost-outermost *($\mathbb{F}_{lo}$) reduce, de entre los redexes externos, el del extremo izquierdo.*

4. *La estrategia (paralela)* parallel-outermost *($\mathbb{F}_{po}$) reduce todos los redexes externos de forma simultánea.*

5. *La regla de* sustitución completa *(*full substitution rule, $\mathbb{F}_{fs}$*) reduce simultáneamente todos los redexes (sólo está bien definida para SRTs ortogonales).*

En las estrategias enumeradas, se ha obviado la cuestión de qué regla de reescritura se empleará para reducir el redex seleccionado. Esto no es un problema para SRTs débilmente o cuasi ortogonales, puesto que el reemplazamiento que debe realizarse para contraer dicho redex está definido de forma inequívoca [AM96]. Por tanto, las estrategias concretas anteriores están bien definidas sea cual sea el SRT considerado.

Aunque es posible asociar a cada SRT fuertemente secuencial una estrategia de reducción de índices, una implementación de esta estrategia genérica requeriría, en principio, implementar también la Ω-reducción. Esto no resulta totalmente adecuado, desde el punto de vista práctico, por ser bastante costoso (ver el Capítulo 9 para una discusión más detallada). Para obtener implementaciones más eficientes, normalmente debemos limitarnos a una subclase de SRTs aún a costa de perder completitud (o, al menos, un comportamiento óptimo) cuando (con poca frecuencia, por lo general) nos vemos abocados a ejecutar un programa para el cual dicha implementación no funciona adecuadamente.

Esta situación no es tan anormal en el uso cotidiano de los lenguajes de programación: las implementaciones del lenguaje funcional perezoso *Haskell* no soportan programas fuertemente secuenciales en su máxima generalidad porque la implementación está concebida para funcionar correctamente y de forma más eficiente para una clase más reducida de programas[1].

Por tanto, para una utilización efectiva de un SRT de una clase CL de SRTs (fuertemente secuenciales), proporcionaremos una estrategia genérica especializada $\mathbb{F}$ cuya implementación pueda realizarse de modo más eficaz.

Ejemplo 8.1.3 *Las estrategias de reducción de índices se caracterizan por el hecho de que siempre reducen redexes que están situados sobre índices. Por tanto, pueden formalizarse como estrategias genéricas para la clase de los SRTs lineales por la izquierda y fuertemente secuenciales. Sabemos que estas estrategias son normalizantes, por ejemplo, para la subclase de los SRTs lineales por la izquierda, fuertemente secuenciales y convergentes a la raíz (Teorema 2.5.21). Pero, en particular, sabemos que la estrategia* leftmost-outermost, $\mathbb{F}_{lo}$*, es una estrategia de reducción de índices para la subclase de los SRTs normales por la izquierda. Parece evidente (y será discutido*

[1] En el contexto de la *programación lógica* sucede algo parecido: en el algoritmo de unificación incorporado en muchas implementaciones del lenguaje *Prolog* no se comprueba la ocurrencia de variables anidadas en los enlaces de los unificadores computados (*occur-check*) debido al coste que supone dicha comprobación [Roy94]. Como resultado, estas implementaciones son incorrectas desde el punto de vista lógico. En el caso de los *lenguajes imperativos* tenemos un problema similar: el modo de evaluación de las llamadas a función es siempre *voraz* lo que provoca la entrada de los programas en secuencias de ejecución infinitas (lo cual es innecesario desde el punto de vista teórico).

con más profundidad en el capítulo 9) que va a ser mucho más sencillo implementar $\mathbb{F}_{lo}$ *que implementar la reducción de índices para los SRTs fuertemente secuenciales generales.*

Existe una diferencia interesante entre las estrategias genéricas enumeradas en el Ejemplo 8.1.2 y las estrategias de reducción de índices de carácter general aludidas en el Ejemplo 8.1.3. Esta diferencia se discute en la sección siguiente.

8.1.1 Definición de estrategias genéricas

Para un SRT débilmente ortogonal $\mathcal{R}$, una estrategia $\mathbb{F}_{\mathcal{R}}$ viene completamente determinada por el componente $\mathbb{P}_{\mathcal{R}}$ de la estrategia [AM96]. En el caso de las estrategias genéricas, se puede definir una de éstas sin más que proporcionar una función que llamamos función de *selección*.

Definición 8.1.4 (Función de selección) *Una función* $\widehat{\mathbb{P}} : \mathcal{P}(Occ) \to \mathcal{P}(Occ)$ *es una* función de selección *si satisface las siguientes propiedades:*

1. $\widehat{\mathbb{P}}(U) \neq \emptyset$ *si* $U \neq \emptyset$
2. $\widehat{\mathbb{P}}(U) \subseteq U$ *para todo* $U \in \mathcal{P}(Occ)$

Observemos que la condición 2 en la Definición 8.1.4 implica que la función de selección $\widehat{\mathbb{P}}$ es estricta (respecto al orden usual sobre conjuntos), es decir $\widehat{\mathbb{P}}(\emptyset) = \emptyset$. Esto se debe a que, si $U = \emptyset$, por la condición 2 tenemos que $\widehat{\mathbb{P}}(U) = \widehat{\mathbb{P}}(\emptyset) \subseteq \emptyset$, lo cual sólo es posible si $\widehat{\mathbb{P}}(\emptyset) = \emptyset$.

En la próxima definición, $O_{\mathcal{R}}$ es una función que suministra las ocurrencias $O_{\mathcal{R}}(t)$ de los redexes de un término t de acuerdo con el SRT $\mathcal{R}$.

Definición 8.1.5 (Estrategia genérica estructural) *Una estrategia genérica* $\mathbb{F} = (\mathbb{P}, \mathbb{R})$ *para una clase CL de SRTs débilmente ortogonales es* estructural *si existe una función de selección* $\widehat{\mathbb{P}}$ *tal que para todo* $\mathcal{R} \in CL$*, se verifica* $\mathbb{P}_{\mathcal{R}} = \widehat{\mathbb{P}} \circ O_{\mathcal{R}}$*. Entonces decimos que* $\mathbb{F}$ *está* basada *en* $\widehat{\mathbb{P}}$*.*

Observación 8.1.6 *A partir de la Definición 8.1.5, resulta evidente cómo definir una estrategia estructural a partir de una función de selección* $\widehat{\mathbb{P}}$ *cualquiera. Simplemente, dada una clase CL (cualquiera) de SRTs débilmente ortogonales, imponemos que* $\mathbb{P}_{\mathcal{R}} = \widehat{\mathbb{P}} \circ O_{\mathcal{R}}$ *para todo SRT* $\mathcal{R} \in CL$*. El componente* $\mathbb{R}_{\mathcal{R}}$ *viene determinado por* $\mathbb{P}_{\mathcal{R}}$ *de forma segura.*

Con relación a la Observación 8.1.6, puede comprobarse que el objeto $\mathbb{F}$ definido mediante el procedimiento sugerido allí especifica una estrategia genérica o, lo que es lo mismo, que cada concreción $\mathbb{F}_{\mathcal{R}}$ de ese objeto es una estrategia de reescritura para

un SRT $\mathcal{R}$ dado. La condición $\widehat{\mathbb{P}}(U) \neq \emptyset$ si $U \neq \emptyset$ de la Definición 8.1.4 asegura que la estrategia seleccionará un redex siempre que haya alguno disponible para reducir. Por otro lado, la condición $\widehat{\mathbb{P}}(U) \subseteq U$ para todo $U \in \mathcal{P}(\mathbb{N}^*_{>0})$ asegura que $\widehat{\mathbb{P}}$ no introduce ocurrencias extrañas que no corresponden a un redex (y que, por tanto, no podrían reducirse). Luego el componente $\mathbb{P}_{\mathcal{R}}$ de la estrategia está bien definido. La ortogonalidad débil asegura que el componente $\mathbb{R}_{\mathcal{R}}$ viene determinado de forma segura por el componente $\mathbb{P}_{\mathcal{R}}$.

Observación 8.1.7 *Claramente, si una estrategia genérica* $\mathbb{F}$ *para una clase CL de SRTs débilmente ortogonales es estructural, el componente* $\mathbb{P}_{\mathcal{R}}$ *de toda concreción* $\mathbb{F}_{\mathcal{R}}$ *para cada* $\mathcal{R} \in CL$ *también podrá descomponerse como* $\mathbb{P}_{\mathcal{R}} = \widehat{\mathbb{P}} \circ O_{\mathcal{R}}$. *Por tanto, también diremos que la estrategia* $\mathbb{F}_{\mathcal{R}}$ *es* estructural.

Puesto que $\widehat{\mathbb{P}}$ no depende de ningún SRT, la función $\widehat{\mathbb{P}}$ suele representar la regla empleada para definir una estrategia genérica estructural $\mathbb{F}$ y, a veces, es tomada como tal en lugar de como vehículo de definición de la misma. Sólo $O_{\mathcal{R}}$ hace patente el SRT concreto empleado en la reducción. Las estrategias genéricas enumeradas en el Ejemplo 8.1.2 son, de hecho, estructurales.

Ejemplo 8.1.8 (Ejemplos de estrategias estructurales) *Consideremos la función* $\{_\} : Occ \to \mathcal{P}(Occ)$ *que convierte una cadena* $u \in Occ$ *en un conjunto unitario* $\{u\}$ *y sea* $Id_{\mathcal{P}(Occ)}$ *la identidad para* $\mathcal{P}(Occ)$. *A continuación presentamos las funciones de selección de las estrategias genéricas del Ejemplo 8.1.2 (sólo presentamos la restricción a* $\mathcal{P}(Occ)\backslash\emptyset$; *extenderíamos las definiciones a* $\mathcal{P}(Occ)$ *forzando que sean estrictas):*

1. *La estrategia* leftmost-innermost *(*$\mathbb{F}_{li}$*):*

$$\widehat{\mathbb{P}}_{li} = \{min_{\leq_L} \circ maximal_{\leq}\}$$

2. *La estrategia* parallel-innermost *(*$\mathbb{F}_{pi}$*):*

$$\widehat{\mathbb{P}}_{pi} = maximal_{\leq}$$

3. *La estrategia* leftmost-outermost *(*$\mathbb{F}_{lo}$*):*

$$\widehat{\mathbb{P}}_{lo} = \{min_{\leq_L} \circ minimal_{\leq}\} = \{min_{\leq_L}\}$$

4. *La estrategia* parallel-outermost *(*$\mathbb{F}_{po}$*):*

$$\widehat{\mathbb{P}}_{po} = minimal_{\leq}$$

5. *La regla de* sustitución completa ($\mathbb{F}_{fs}$)*:*

$$\widehat{\mathbb{P}}_{fs} = Id_{\mathcal{P}(Occ)}$$

Las estrategias genéricas estructurales (y sus concreciones) son 'independientes' de las *rhs* de las reglas de los SRTs a que se aplican, debido a que éstas *no* son tenidas en cuenta para seleccionar el redex que va a ser contraído. Esto es así porque el conjunto de ocurrencias de redexes $O_{\mathcal{R}}(t)$, que codifica toda la información acerca del SRT $\mathcal{R}$ que se le permite emplear a una estrategia estructural para decidir el redex a reducir, sólo depende del conjunto $L(\mathcal{R})$ de *lhs*'s de las reglas.

Observación 8.1.9 *Las estrategias estructurales son estrategias sin anticipación (ver la Sección 2.4). Las estrategias estructurales son independientes también de la signatura involucrada y de los símbolos concretos que etiquetan la estructura arbórea asociada al término que va a ser reducido. Esto es así porque la selección del redex está basada en la información* puramente estructural *expresada mediante* ocurrencias *de los términos. Dado que las ocurrencias son cadenas numéricas, hemos perdido la información referente a los símbolos del término de entrada. Esto justifica la denominación* estructural *empleada para esta clase de estrategias.*

Observación 8.1.10 *Si trabajamos con estrategias estructurales, el coste de determinar un paso de reducción está acotado por el máximo entre el coste de localización de un redex en un término t (lo cual puede hacerse con un coste $\mathcal{O}(termsize \times nmatches)$, donde termsize es el tamaño de t y nmatches es el número de redexes en t [HO82]) y el coste del cálculo realizado por la función de selección (que puede ser constante si la selección realizada por $\widehat{\mathbb{P}}$ es indeterminista).*

El siguiente ejemplo muestra que las estrategias de reducción de índices no tienen porqué ser, en general, estructurales.

Ejemplo 8.1.11 *Sea $\mathbb{F}$ la estrategia genérica de reducción de índices para SRTs ortogonales y fuertemente secuenciales que, dado un término t, reduce el redex localizado sobre el índice que se encuentra en el extremo izquierdo de s (el Ω-término obtenido al reemplazar los redexes externos en t por Ω). Esta estrategia genérica de reducción de índices no es estructural. Para comprobarlo, consideremos el SRT ortogonal y fuertemente secuencial $\mathcal{R}$ del Ejemplo 7.5.4. Sean s_1 y s_2 redexes y consideremos los términos $t = \mathtt{c}(s_1, s_2)$ y $t' = \mathtt{f}(s_1, s_2)$. Puesto que 1 es un índice de $\mathtt{c}(\Omega,\Omega)$ pero no de $\mathtt{f}(\Omega,\Omega)$ (porque $\omega(\mathtt{f}(\bullet,\Omega)) = \Omega$), tenemos que $\mathbb{P}_{\mathcal{R}}(t) = 1$ y $\mathbb{P}_{\mathcal{R}}(t') = 2$.*

Sin embargo, $O_{\mathcal{R}}(t) = O_{\mathcal{R}}(t') = \{1, 2\}$. Por tanto, no hay ninguna función de selección $\widehat{\mathbb{P}}$ tal que $\mathbb{P}_{\mathcal{R}} = \widehat{\mathbb{P}} \circ O_{\mathcal{R}}$ (pues $\widehat{\mathbb{P}}(O_{\mathcal{R}}(t)) = \widehat{\mathbb{P}}(O_{\mathcal{R}}(t'))$ pero, por definición de $\mathbb{F}$, $\mathbb{P}_{\mathcal{R}}(t) \neq \mathbb{P}_{\mathcal{R}}(t')$) y ello significa que $\mathbb{F}$ no *es estructural.*

8.1.2 Estrategias genéricas sensibles al contexto

La siguiente definición sirve para extender el concepto de estrategia genérica a la *rsc*.

Definición 8.1.12 (Par válido) *Un par* $(\mathcal{R}, \mu)$ *donde* $\mathcal{R} = (\Sigma, R)$ *es un SRT y* $\mu \in M_\Sigma$ *se denomina* par válido.

Utilizaremos las denominaciones habituales para clasificar SRTs (lineales por la izquierda, ortogonales, etc.) aplicándolas a clases de pares válidos: diremos que un par válido $(\mathcal{R}, \mu)$ es lineal por la izquierda, ortogonal, etc. si lo es el componente $\mathcal{R}$ de dicho par.

Definición 8.1.13 (Estrategia genérica sc) *Una* estrategia genérica *sc para una clase VP de pares válidos es una función* $\mathbb{H}_{VP}$ *(o simplemente* $\mathbb{H}$*) que asocia a cada par* $(\mathcal{R}, \mu) \in VP$ *una* μ*-estrategia* $\mathbb{H}_{\mathcal{R}(\mu)}$ *para* $\mathcal{R}$.

Diremos que una estrategia genérica *sc* para un conjunto de pares válidos *VP* es *sc*-normalizante si para todo par válido $(\mathcal{R}, \mu)$, la concreción $\mathbb{H}_{\mathcal{R}(\mu)}$ es μ-normalizante para $\mathcal{R}$.

La definición de estrategia genérica estructural es fácilmente adaptable a estrategias genéricas *sc* utilizando $O^\mu_\mathcal{R}$ en vez de $O_\mathcal{R}$.

Definición 8.1.14 (Estrategia genérica sc estructural) *Una estrategia genérica* sc $\mathbb{H} = (\mathbb{P}, \mathbb{R})$ *para una clase VP de pares válidos débilmente ortogonales es* estructural *si existe una función de selección* $\widehat{\mathbb{P}}$ *tal que para todo* $(\mathcal{R}, \mu) \in VP$*, se verifica* $\mathbb{P}^\mu_\mathcal{R} = \widehat{\mathbb{P}} \circ O^\mu_\mathcal{R}$*. Entonces decimos que* $\mathbb{H}$ *está* basada *en* $\widehat{\mathbb{P}}$.

Dada una estrategia genérica estructural $\mathbb{F}$ (basada en una función de selección $\widehat{\mathbb{P}}$) para una clase *CL* de SRTs, es posible asociar a ésta varias estrategias genéricas *sc* estructurales (también basadas en $\widehat{\mathbb{P}}$) mediante la definición de distintos conjuntos de pares válidos constituidos por SRTs pertenecientes a *CL*.

Definición 8.1.15 *Sea CL una clase de SRTs. Definimos las clases* $VP^{sc}(CL)$ *y* $VP^{can}(CL)$ *de pares válidos como sigue:*

$$VP^{sc}(CL) = \{(\mathcal{R}, \mu) \mid \mathcal{R} = (\Sigma, R) \in CL \wedge \mu \in M_\Sigma\}$$

$$VP^{can}(CL) = \{(\mathcal{R}, \mu) \mid \mathcal{R} = (\Sigma, R) \in CL \wedge \mu \in M_\Sigma \wedge \mu^{com}_\mathcal{R} \sqsubseteq \mu\}$$

Ahora podemos dar dos nociones de *restricción sensible al contexto* de una estrategia genérica estructural.

Definición 8.1.16 (Restricción sc de una estrategia genérica estructural) *Sea* $\mathbb{F}$ *una estrategia genérica estructural para una clase CL de SRTs débilmente ortogonales, basada en una función de selección* $\widehat{\mathbb{P}}$*. Definimos:*

1. *La* restricción *sc* $\mathbb{F}^{sc}$ *de* $\mathbb{F}$ *es la estrategia genérica* sc *estructural basada en* $\widehat{\mathbb{P}}$ *para el conjunto* $VP^{sc}(CL)$ *de pares válidos.*

2. *La* restricción *sc* canónica $\mathbb{F}^{can}$ *de* $\mathbb{F}$ *es la estrategia genérica* sc *estructural basada en* $\widehat{\mathbb{P}}$ *para el conjunto* $VP^{can}(CL)$ *de pares válidos.*

Ahora podemos establecer un primer resultado: las restricciones canónicas de las estrategias genéricas *parallel-outermost* y de sustitución completa son estabilizantes y *sc*-normalizantes para SRTs ortogonales.

Teorema 8.1.17 *Las estrategias genéricas* sc $\mathbb{F}^{can}_{po}$ *y* $\mathbb{F}^{can}_{fs}$ *son estabilizantes y* sc*-normalizantes para* $VP^{can}(ORT)$.

DEMOSTRACIÓN. Inmediata a partir de los Teoremas 7.2.5, 2.5.7 y 7.2.6. □

Puesto que la *rsc* preserva la terminación de los SRTs (Teorema 5.1.4), es interesante plantearse si sucede lo mismo con el carácter normalizante de las estrategias, es decir:

> *Dada una estrategia genérica estructural* $\mathbb{F}$ *normalizante para una clase CL de SRTs ¿es su restricción* sc $\mathbb{F}^{sc}$ *(o* $\mathbb{F}^{can}$*)* sc*-normalizante para* $VP^{sc}(CL)$ *(*$VP^{can}(CL)$*)?*

Un análisis sistemático del carácter normalizante *heredado* por este tipo de estrategias restringidas es difícil, debido a la necesidad de identificar la *causa* por la que la estrategia original es normalizante, y de estudiar si dicha 'causa' permanece operante tras realizar la restricción *sc* de la estrategia original.

Si consideramos estrategias genéricas de *reducción de índices*, la *causa* del carácter normalizante de las mismas es precisamente el tipo de redexes seleccionados en cada paso de reducción (redexes situados sobre índices) y la propiedad que tienen las secuencias de reducción de índices de ser normalizantes (Teorema 2.5.21). Por desgracia, tal como se puso de manifiesto mediante el Ejemplo 8.1.11, las estrategias de reducción de índices no son, en general, estructurales. Por tanto, ni siquiera disponemos de una noción de estrategia restringida *sc* que se pueda emplear para derivar una estrategia genérica *sc* a partir de una estrategia genérica de reducción de índices cualquiera puesto que la Definición 8.1.16 exige que la estrategia de partida sea estructural.

Aún más, incluso si sabemos que la estrategia genérica de reducción de índices es estructural (como en el caso de $\mathbb{F}_{lo}$ para los SRTs lineales por la izquierda y normales por la izquierda), no está claro si la restricción *sc* de ésta será una estrategia *sc* de reducción de índices: dado un término t y una estrategia de reducción de índices

$\mathbb{F}_{\mathcal{R}}$ para $\mathcal{R}$, el Teorema 7.4.6 asegura que la $\mathbb{F}_{\mathcal{R}}$-subsecuencia inicial de reducción de índices

$$t = t_1 \overset{u_1}{\rightarrow}_{\mathbb{F}_{\mathcal{R}}} t_2 \overset{u_2}{\rightarrow}_{\mathbb{F}_{\mathcal{R}}} \cdots \overset{u_{n-1}}{\rightarrow}_{\mathbb{F}_{\mathcal{R}}} t_n \overset{u_n}{\rightarrow}_{\mathbb{F}_{\mathcal{R}}} t_{n+1} = s$$

donde $t_1, \ldots, t_n$ son términos frágiles, $u_i \in \mathcal{I}(t_i)$ para $1 \leq i \leq n$ y $t_{n+1} = s$ no es frágil (es decir, s es fuertemente estable), verifica que $u_i \in \mathcal{I}^{\mu}(t_i)$ para $1 \leq i \leq n$. Por tanto, imponer

$$\mathbb{H}_{\mathcal{R}(\mu)}(t) = \mathbb{F}_{\mathcal{R}}(t)$$

para todo término frágil t es correcto y se obtiene una μ-estrategia que reduce índices cuando se aplica sobre términos frágiles. Sin embargo, respecto a la $\mathbb{F}_{\mathcal{R}}$-subsecuencia de reducción que parte de s y lleva hasta el final de la $\mathbb{F}_{\mathcal{R}}$-secuencia, no hay ninguna forma clara de definir cómo continuará la $\mathbb{H}_{\mathcal{R}(\mu)}$-secuencia de reducción de índices para hacer que *corresponda* a $\mathbb{F}_{\mathcal{R}}$ en algún sentido y conserve sus buenas propiedades.

En la sección siguiente exploramos una nueva posibilidad para el establecimiento de correspondencias entre estrategias de reducción de índices y estrategias *sc* de reducción de índices. En este caso, será la función de μ-contracción estudiada en el Capítulo 5 la que brinde esta nueva vía de definición y análisis de estrategias *sc*.

8.2 Secuencialidad fuerte y contracción de SRTs

En esta sección veremos que la función de μ-contracción de términos τ_{μ} introducida en el Capítulo 5 preserva las ocurrencias de índices reemplazantes existentes en un término. Esto permite establecer una correspondencia interesante entre la secuencialidad fuerte de un SRTE $\mathcal{R}$ y la secuencialidad fuerte de su μ-contracción $\mathcal{R}^{\mu}$. Al combinar ambos hechos llegamos a la noción de estrategia genérica *sc inducida* por una estrategia genérica. Esta nueva correspondencia entre estrategias y estrategias *sc* sí que preserva el carácter de estrategia de reducción de índices cuando la estrategia base es una estrategia de reducción de índices *arbitraria*. Además, obtenemos la ventaja añadida de que las estrategias *sc* inducidas son aplicables, por lo general, a un mayor número de SRTs que las estrategias de partida (ver el Capítulo 10).

Notación 8.2.1 *Utilizaremos $\omega_{\mu}(t)$ para denotar la forma $\rightarrow_{\Omega_{\mu}}$-normal de t empleando las lhs's de $\mathcal{R}^{\mu}$ y $\mathcal{J}(t)$ para referirnos al conjunto de índices respecto a $\rightarrow_{\Omega_{\mu}}$.*

El siguiente resultado establece que la transformación contractiva preserva los índices reemplazantes del SRTE sobre el cual se aplica.

Proposición 8.2.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\mu \in M_{\Sigma}$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ y $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$. Si $u \in \mathcal{I}^{\mu}(t)$, entonces $\nu_t(u) \in \mathcal{J}(\tau_{\mu}(t))$.*

DEMOSTRACIÓN. Sea $s[\bullet]_u = \omega(t[\bullet]_u)$. Dado que $t[\bullet]_u \rightarrow_\Omega^* s[\bullet]_u$, por la Proposición 5.7.15 tenemos que $\tau_\mu(t[\bullet]_u) \rightarrow_{\Omega_\mu}^* \tau_\mu(s[\bullet]_u)$ y, dado que $u \in O^\mu(t[\bullet]_u)$ y $u \in O^\mu(s[\bullet]_u)$ (y esto a lo largo de la derivación completa, en virtud del Lema 7.4.2), la Proposición 5.2.8 permite obtener el resultado: $\tau_\mu(t)[\bullet]_{\nu_t(u)} \rightarrow_{\Omega_\mu}^* \tau_\mu(s)[\bullet]_{\nu_s(u)}$. Por el Teorema 5.7.17, $\tau_\mu(s)[\bullet]_{\nu_s(u)}$ es una forma $\rightarrow_{\Omega_\mu}$-normal y, de hecho, $\tau_\mu(s)[\bullet]_{\nu_s(u)} = \omega_\mu(\tau_\mu(t[\bullet]_u))$. Puesto que $s[\bullet]_u$ se obtuvo por Ω-reducción a partir de $t[\bullet]_u$, podemos suponer que $sprefix_t(u) = sprefix_s(u)$. Puesto que $u \in O^\mu(t)$ y $u \in O(t) \cap O(s)$, por la Proposición 3.2.3, $u \in O^\mu(t) \cap O^\mu(s)$. Entonces, por la Proposición 5.2.9, tenemos que $\nu_t(u) = \nu_s(u)$. Por tanto, $\tau_\mu(s)[\bullet]_{\nu_s(u)} = \tau_\mu(s)[\bullet]_{\nu_t(u)} = \omega_\mu(\tau_\mu(t[\bullet]_u))$ y por definición de índice (Definición 2.5.15), tenemos que $\nu_t(u) \in \mathcal{J}(\tau_\mu(t))$. □

En general, no es posible prescindir de la condición $\mu_\mathcal{R}^{com} \sqsubseteq \mu$ si queremos asegurar el resultado en las condiciones de la proposición.

Ejemplo 8.2.3 *Consideremos el SRT $\mathcal{R}$:*

```
f(x,b) → a
```

La ocurrencia 1 *es un índice de* $t =$ `f(Ω,a)`*, porque* `f(•,a)` *es una forma* $\rightarrow_\Omega$*-normal. Si escogemos* μ(`f`) $= \{1\}$*, entonces* $\nu_t(1) = 1$ *no es un índice de* $\tau_\mu(t) =$ $\mathtt{f}_\mu(\Omega_\mu)$ *puesto que*

$\mathtt{f}_\mu(\bullet) \rightarrow_{\Omega_\mu} \Omega_\mu$

Por tanto, $1 \notin \mathcal{J}(\tau_\mu(t))$.

Proposición 8.2.4 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$ y $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$. Si $\nu_t(u) \in \mathcal{J}(\tau_\mu(t))$, entonces $u \in \mathcal{I}^\mu(t)$.*

DEMOSTRACIÓN. Por la Definición 2.5.15, tenemos que $\nu_t(u) \in \mathcal{J}(\tau_\mu(t))$ si y sólo si $\tau_\mu(t)[\bullet]_{\nu_t(u)} \rightarrow_{\Omega_\mu}^* \tau_\mu(s)[\bullet]_{\nu_s(u)} = \omega_\mu(\tau_\mu(t)[\bullet]_{\nu_t(u)})$. Esto significa que $\nu_t(u) = \nu_s(u)$. Por la Proposición 5.2.8, $\tau_\mu(t[\bullet]_u) \rightarrow_{\Omega_\mu}^* \tau_\mu(s[\bullet]_u)$. Puesto que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$, por la Proposición 5.7.16, existe un $s'[\bullet]_u \equiv_{\tau_\mu} s[\bullet]_u$ tal que $t[\bullet]_u \rightarrow_\Omega^* s'[\bullet]_u$. Por el Teorema 5.7.19, $CMR^\mu(s'[\bullet]_u) = CMR^\mu(s[\bullet]_u) = C[\,]$ es rígido y como $u \in O^\mu(C[\tilde{\Omega}])$, tenemos que $\bullet$ está presente en $\omega(t[\bullet]_u)$, lo cual completa la demostración. □

Una consecuencia inmediata de las Proposiciones 8.2.2 y 8.2.4 es el siguiente teorema, que expresa la estrecha correspondencia existente entre los índices reemplazantes respecto a un SRT $\mathcal{R}$ y una función de reemplazamiento μ, y los índices computados respecto al SRT μ-contracto $\mathcal{R}^\mu$.

Teorema 8.2.5 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$, entonces $\nu_t(\mathcal{I}^\mu(t)) = \mathcal{J}(\tau_\mu(t))$*

8.2.1 μ-contracción de SRTEs y decidibilidad de la secuencialidad fuerte

El problema de decidir si un SRTE cualquiera es fuertemente secuencial ha sido demostrado co-NP-completo por Durand [Dur95]. Algunas subclases admiten procedimientos de decisión menos costosos. Particularmente, las clases de SRTEs caracterizables mediante algún criterio sintáctico simple, tales como los SRTEs normales por la izquierda, etc.

Los resultados obtenidos en la sección anterior nos permiten establecer conexiones interesantes entre el carácter fuertemente secuencial de un SRTE y el de su μ-contracción. Esto permite, a veces, reducir el coste de reconocer la secuencialidad fuerte de un SRTE examinando la del SRTE μ-contracto.

Teorema 8.2.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Si $\mathcal{R}^\mu$ es fuertemente secuencial, entonces $\mathcal{R}$ es fuertemente secuencial.*

DEMOSTRACIÓN. De acuerdo con la Definición 2.5.19, debemos demostrar que toda forma Ω-normal t tiene un índice. Lo haremos por inducción estructural. Si $t = \Omega$ (el único símbolo constante que es una forma Ω-normal) es inmediato. Para términos más complejos, distinguimos dos casos.

(1) $O^\mu_\Omega(t) = \emptyset$. En ese caso, puesto que cualquier forma normal es una forma μ-normal, por la Proposición 4.3.1 podemos expresar t como $t = C[t_1, \ldots, t_n]$ con $C[\,] = CMR^\mu(t)$ siendo $C[\widetilde{\Omega}]$ un término rígido, de forma que las Ω-ocurrencias no son reemplazantes. Recordemos que $C[\,]$ no es un contexto vacío. Por la Proposición 2.5.12, podemos escribir t como sigue: $t = C'[t'_1, \ldots, t'_m]$, donde C' es un contexto rígido máximo y $t'_1, \ldots, t'_m$ son términos frágiles. Puesto que $C[\widetilde{\Omega}] \leq C'[\widetilde{\Omega}]$ y dado que $C[\,]$ no es vacío, podemos aplicar la hipótesis de inducción a un término t'_i que sea una forma Ω-normal (y este t'_i existe porque $O^\mu_\Omega(t) = \emptyset$ pero $O_\Omega(t) \neq \emptyset$) para concluir que posee un índice. Por la Proposición 2.5.18, dicho índice también es un índice de t.

(2) $O^\mu_\Omega(t) \neq \emptyset$. En este caso, dado que toda forma normal es μ-normal, por el Teorema 5.7.10 $\tau_\mu(t)$ es una forma normal respecto a $\mathcal{R}^\mu$ y, por la Proposición 5.2.8, dado que t contiene una ocurrencia reemplazante de Ω, $\tau_\mu(t)$ contiene una ocurrencia de Ω_μ, luego es una forma Ω_μ-normal. Por la secuencialidad fuerte del SRTE $\mathcal{R}^\mu$, existe un índice $\nu_t(u) \in \mathcal{J}(\tau_\mu(t))$. Por la Proposición 8.2.4, $u \in \mathcal{I}^\mu(t)$ lo cual concluye la demostración. □

Con el fin de ilustrar la aplicación del Teorema 8.2.6, consideremos el SRT $\mathcal{R}$ del Ejemplo 1.1.2. Dicho SRT *no* es normal por la izquierda (hay símbolos de función en las reglas de reescritura que aparecen a la derecha de algunas variables). Por tanto, no hay una caracterización sintáctica simple para enmarcarlo en una familia conocida

de SRTs fuertemente secuenciales. Sin embargo, el SRT $\mathcal{R}^\mu$, donde $\mu = \mu_{\mathcal{R}}^{com}$, del Ejemplo 5.4.5 es normal por la izquierda, luego fuertemente secuencial. Ahora puede utilizarse el Teorema 8.2.6 para establecer que el propio $\mathcal{R}$ es fuertemente secuencial. Presentemos un ejemplo más.

Ejemplo 8.2.7 *Sea el SRT $\mathcal{R}$ del Ejemplo 3.4.4. El SRT $\mathcal{R}$* no *es normal por la izquierda. Si consideramos la función de reemplazamiento μ dada en el Ejemplo 3.4.4, entonces $\mathcal{R}^\mu$ es:*

$$
\begin{array}{lll}
\mathtt{from}_\mu & \to \mathtt{::}_\mu & \\
\mathtt{primes}_\mu & \to \mathtt{sieve}_\mu(\mathtt{from}_\mu) & \\
\mathtt{sieve}_\mu(\mathtt{::}_\mu) & \to \mathtt{::}_\mu & \\
\mathtt{filter}_\mu(\mathtt{::}_\mu) & \to \mathtt{if}_\mu(\mathtt{|}_\mu) & \\
\mathtt{sel}_\mu(\mathtt{0}_\mu,\mathtt{::}_\mu) & \to \mathtt{x} & \mathtt{if}_\mu(\mathtt{true}_\mu) \to \mathtt{x} \\
\mathtt{sel}_\mu(\mathtt{s}_\mu,\mathtt{::}_\mu) & \to \mathtt{sel}_\mu(\mathtt{x},\mathtt{z}) & \mathtt{if}_\mu(\mathtt{false}_\mu) \to \mathtt{y}
\end{array}
$$

El SRTE $\mathcal{R}^\mu$ es normal por la izquierda y, por tanto, fuertemente secuencial. Luego $\mathcal{R}$ también lo es.

A continuación presentamos un resultado que demuestra que la relación entre la secuencialidad fuerte de un SRTE y su μ-contracción es todavía más estrecha.

Teorema 8.2.8 *Si $\mathcal{R} = (\Sigma, R)$ es un SRTE lineal por la izquierda y fuertemente secuencial y $\mu \in M_\Sigma$ es tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$, entonces $\mathcal{R}^\mu$ es fuertemente secuencial.*

DEMOSTRACIÓN. De acuerdo con la Definición 2.5.19, demostraremos que toda forma Ω_μ-normal (respecto a $\mathcal{R}^\mu$) tiene un índice. Sea $\tau_\mu(t) \in \mathcal{T}(\Sigma^\mu \cup \{\Omega_\mu\}, V)$ una forma Ω_μ-normal, es decir, $O_{\mathcal{R}^\mu}(\tau_\mu(t)) = \emptyset$ y $O_{\Omega_\mu}(\tau_\mu(t)) \neq \emptyset$. Sea $C[\,] = CMR^\mu(t)$ el contexto reemplazante máximo de la $\equiv_{\tau_\mu}$-clase $[t]_{\equiv_{\tau_\mu}}$. Adviértase que, por la Proposición 5.2.8, $O_\Omega^\mu(C[\widetilde{\Omega}]) = \nu_t^{-1}(O_{\Omega_\mu}(\tau_\mu(t)))$. Por el Lema 2.3.3 y la Proposición 5.7.5, $C[\widetilde{\Omega}]$ es una forma μ-normal y, puesto que $O_{\Omega_\mu}(\tau_\mu(t)) = O_{\Omega_\mu}(\tau_\mu(C[\widetilde{\Omega}])) \neq \emptyset$, tenemos que $O_\Omega^\mu(C[\widetilde{\Omega}]) \neq \emptyset$. Dado que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$, por el Teorema 7.4.8, $C[\widetilde{\Omega}]$ tiene un índice reemplazante $u \in \mathcal{I}^\mu(C[\widetilde{\Omega}])$. Por la Proposición 8.2.2, y dado que $\tau_\mu(t) = \tau_\mu(C[\widetilde{\Omega}])$, obtenemos la conclusión deseada. □

Podemos utilizar el Teorema 8.2.8 para decidir acerca de la secuencialidad fuerte de los SRTEs.

Ejemplo 8.2.9 *Puede demostrarse que el SRT $\mathcal{R}$ del Ejemplo 5.4.6* no *es fuertemente secuencial. Esto es así porque el SRT $\mathcal{R}^\mu$:*

$$
\begin{array}{lll}
\mathtt{f}_\mu(\mathtt{g}_\mu(\mathtt{a}_\mu,\mathtt{x}),\mathtt{f}_\mu(\mathtt{b}_\mu,\mathtt{z})) & \to \mathtt{x} & \mathtt{g}_\mu(\mathtt{d}_\mu,\mathtt{d}_\mu) \to \mathtt{h}_\mu \\
\mathtt{f}_\mu(\mathtt{g}_\mu(\mathtt{x},\mathtt{a}_\mu),\mathtt{f}_\mu(\mathtt{c}_\mu,\mathtt{z})) & \to \mathtt{z} & \mathtt{h}_\mu \to \mathtt{e}_\mu
\end{array}
$$

no *es fuertemente secuencial (de hecho, las primeras tres lhs's están tomadas del ejemplo de SRT que no es fuertemente secuencial dado en la página 420 de [HL91]). Por tanto, por el Teorema 8.2.8, $\mathcal{R}$ no es fuertemente secuencial.*

Los Teoremas 8.2.6 y 8.2.8 establecen que demostrar la secuencialidad fuerte de un SRTE $\mathcal{R}$ es equivalente a demostrar la de su μ-contracción (siempre que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$).

Observación 8.2.10 *Es conocido que el problema de decidir si un SRT es fuertemente secuencial y el de proporcionar una implementación de la reducción de índices para el SRT examinado se encuentran estrechamente relacionados. De hecho, el primer criterio para decidir la secuencialidad fuerte de los SRTs (introducido por Huet y Lévy en [HL79, HL91]) consistía en un algoritmo para (intentar) construir un* matching dag*, una estructura de datos que sirve para implementar la búsqueda de índices en formas Ω-normales. Por tanto, los Teoremas 8.2.6 y 8.2.8 también sugieren una relación entre las implementaciones de la reducción de índices en $\mathcal{R}$ y en $\mathcal{R}^{\mu}$. Esto es realmente así, tal como veremos en el Capítulo 9.*

En las próximas secciones discutimos el uso de la transformación τ_{μ} para definir estrategias *sc* de reducción de índices a partir de estrategias de reducción de índices. Dado que nuestra discusión sobre estrategias de reescritura se centra en los SRTs, y puesto que la transformación τ_{μ} puede hacer que aparezcan variables extra en las rhs's del SRTE μ-contracto $\mathcal{R}^{\mu}$, empleamos una relación de equivalencia $\sim_l$ entre SRTEs que permita asociar a un SRTE un SRT 'equivalente'. Decimos que (Σ, R) y (Σ', R') son *equivalentes por la izquierda* (y escribimos $(\Sigma, R) \sim_l (\Sigma', R')$) si tienen las mismas lhs's, es decir, tenemos que $R = \{l_i \to r_i \mid 1 \leq i \leq n\}$ y $R' = \{l_i \to r'_i \mid 1 \leq i \leq n\}$ para términos $l_1, \ldots, l_n, r_1, \ldots, r_n, r'_1, \ldots, r'_n$ [KM91]. Claramente, dado un SRTE $\mathcal{R}$, siempre existe un SRT $\mathcal{S}$ que es equivalente por la izquierda a $\mathcal{R}$.

La secuencialidad fuerte es una propiedad independiente de la forma concreta de las partes derechas de las reglas presentes en los SRTEs considerados. Por este motivo, en cuanto a la equivalencia $\sim_l$ se refiere, si un SRTE $\mathcal{R}$ es fuertermente secuencial, todo SRTE $\mathcal{S}$ que sea equivalente por la izquierda a $\mathcal{R}$ también lo es. Además, el conjunto de índices de un Ω-término obtenidos a partir de $\mathcal{R}$ o de $\mathcal{S}$ son exactamente los mismos.

8.2.2 Extensión de clases de SRTs fuertemente secuenciales

El Teorema 8.2.6 hace posible obtener nuevas clases de SRTs lineales por la izquierda y fuertemente secuenciales mediante un operador $\mathsf{T} : \mathcal{P}(\overline{SS}) \to \mathcal{P}(\overline{SS})$.

Definición 8.2.11 (Extensión *sc* de una clase de SRTs) *Sea $CL \subseteq \overline{SS}$ una clase de SRTs lineales por la izquierda y fuertemente secuenciales. La* extensión *sc* $\mathsf{T}(CL)$

de CL viene dada mediante la función $\mathsf{T} : \mathcal{P}(\overline{SS}) \to \mathcal{P}(\overline{SS})$ *definida como sigue:*

$$\mathsf{T}(CL) = \{\mathcal{R} \in \overline{LL} \mid \exists \mu \sqsupseteq \mu^{com}_{\mathcal{R}}.\exists \mathcal{S} \sim_l \mathcal{R}^{\mu}.\ \mathcal{S} \in CL\}$$

Notación 8.2.12 *Dado un SRT* $\mathcal{R} \in \mathsf{T}(CL)$*, denotaremos como* $m_{CL}(\mathcal{R})$ *(o simplemente* $m(\mathcal{R})$*) al conjunto de funciones de reemplazamiento* $\mu \sqsupseteq \mu^{com}_{\mathcal{R}}$ *que satisfacen la condición requerida para incluir un SRT* $\mathcal{R}$ *en* $\mathsf{T}(CL)$*, es decir,* $\mu \in m_{CL}(\mathcal{R})$ *si y sólo si* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ *y* $\exists \mathcal{S} \in CL.\ \mathcal{S} \sim_l \mathcal{R}^{\mu}$*. Observemos que, para todo* $\mu \in m(\mathcal{R})$*, tenemos que* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$*.*

El Teorema 8.2.6 asegura que $\mathsf{T}(CL) \subseteq \overline{SS}$ para todo $CL \subseteq \overline{SS}$. Por tanto, el perfil $\mathsf{T} : \mathcal{P}(\overline{SS}) \to \mathcal{P}(\overline{SS})$ propuesto para la función es correcto. Además, puesto que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu_{\mathsf{T}}$ y $\mathcal{R}^{\mu_{\mathsf{T}}} = \mathcal{R}$ (salvo el posible renombramiento de los símbolos de la signatura), $CL \subseteq \mathsf{T}(CL)$. Por tanto, T es *extensiva* y esto justifica que llamemos a la clase $\mathsf{T}(CL)$ la extensión *sc* de CL. También es sencillo comprobar que $CL \subseteq CL'$ implica que $\mathsf{T}(CL) \subseteq \mathsf{T}(CL')$, es decir, T es monótona. A continuación demostraremos que T también es idempotente.

Proposición 8.2.13 *Si* $CL \subseteq \overline{SS}$*, entonces* $\mathsf{T}(CL) = \mathsf{T}(\mathsf{T}(CL))$*.*

DEMOSTRACIÓN. Siendo T extensiva y monótona, sabemos que $\mathsf{T}(CL) \subseteq \mathsf{T}(\mathsf{T}(CL))$. Sólo necesitamos demostrar que $\mathsf{T}(\mathsf{T}(CL)) \subseteq \mathsf{T}(CL)$. Si $\mathcal{R} \in \mathsf{T}(\mathsf{T}(CL))$, entonces existe un $\mu \sqsupseteq \mu^{com}_{\mathcal{R}}$ y $\mathcal{S} \sim_l \mathcal{R}^{\mu}$ tal que $\mathcal{S} \in \mathsf{T}(CL)$. Si empleamos φ para referirnos a funciones de reemplazamiento $\varphi \in M_{\Sigma^{\mu}}$ de la signatura μ-contracta, esto significa que existe $\varphi \in M_{\Sigma^{\mu}}$, con $\varphi_{\mathsf{T}} \sqsupseteq \varphi \sqsupseteq \varphi^{com}_{\mathcal{S}}$, tal que existe $\mathcal{S}' \sim_l \mathcal{S}^{\varphi}$ y $\mathcal{S}' \in CL$. Si $\varphi = \varphi_{\mathsf{T}}$, es inmediato que $\mathcal{R} \in \mathsf{T}(CL)$, porque $\mathcal{S}^{\varphi_{\mathsf{T}}} = \mathcal{S} \sim_l \mathcal{S}' \in CL$ y, por tanto, $\mathcal{R} \in \mathsf{T}(CL)$, luego asumimos que $\varphi_{\mathsf{T}} \sqsupset \varphi \sqsupseteq \varphi^{com}_{\mathcal{S}}$. Entonces existe un f_{μ} tal que $\varphi_{\mathsf{T}}(f_{\mu}) \supset \varphi(f_{\mu}) \supseteq \varphi^{com}_{\mathcal{S}}(f_{\mu})$. Esto significa que todo argumento j-ésimo para $j \in \varphi_{\mathsf{T}}(f_{\mu}) \backslash \varphi(f_{\mu})$ es una variable en cualquier subtérmino $f_{\mu}(\tilde{s})$ de toda *lhs* l_{μ} de $\mathcal{S}$ (puesto que $j \notin \varphi^{com}_{\mathcal{S}}(f_{\mu})$ y utilizando el Lema 4.1.8). Por tanto, el argumento i-ésimo correspondiente (donde $i \in \mu(f)$ y $j = |\mathbb{N}^{+}_{i} \cap \mu(f)|$) en un subtérmino $f(\tilde{t})$ de la *lhs* $l \in L(\mathcal{R})$ de la cual obtuvimos l_{μ} también es una variable. Por tanto, podríamos utilizar la función de reemplazamiento $\mu' \in M_{\Sigma}$ definida extrayendo de $\mu(f)$ aquellos índices i correspondientes a índices j en $\varphi_{\mathsf{T}}(f_{\mu}) \backslash \varphi(f_{\mu})$ para cada f_{μ}. Puesto que eliminamos índices correspondientes a variables, se mantiene que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu'$. De esta forma obtendríamos directamente $\mathcal{S}'$ a partir de $\mathcal{R}$, puesto que $\mathcal{S}' \sim_l \mathcal{S}^{\mu''} = \mathcal{R}^{\mu'}$ y por tanto $\mathcal{R} \in \mathsf{T}(CL)$. □

De acuerdo con los resultados anteriores, podemos establecer la siguiente propiedad para la función T.

Teorema 8.2.14 T *es un operador de cierre.*

De hecho, el operador T también preserva la unión e intersección de clases de SRTs: $\mathsf{T}(CL \cup CL') = \mathsf{T}(CL) \cup \mathsf{T}(CL')$ y $\mathsf{T}(CL \cap CL') = \mathsf{T}(CL) \cap \mathsf{T}(CL')$.

Observación 8.2.15 (Decidibilidad de la clase extendida *sc***)** *Si la clase CL es decidible (es decir, existe un algoritmo que permite saber si un SRT pertenece o no a CL), $\mathsf{T}(CL)$ puede no ser decidible aunque Σ sea finita. Esto es así porque, en general, el reconocimiento de que un SRT $\mathcal{R}$ pertenece a $\mathsf{T}(CL)$ involucra un proceso de búsqueda de un SRT en la clase CL que sea equivalente por la izquierda a la μ-contracción de $\mathcal{R}$ para algún μ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Puesto que la caracterización de la clase CL puede depender de la forma de las rhs's de los SRTs, no es posible garantizar que dicho proceso de búsqueda termine siempre encontrando el SRT que permite asegurar que $\mathcal{R}$ pertenece a $\mathsf{T}(CL)$.*

No obstante, si la caracterización de CL sólo depende de las lhs's de los SRTs, entonces la clase $\mathsf{T}(CL)$ también será decidible.

Otro caso de interés se plantea cuando la clase CL verifica que, para todo $\mathcal{R} \in CL$ y μ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$, existe un $\mathcal{S}$ tal que $\mathcal{S} \sim_l \mathcal{R}^{\mu}$. Entonces, el problema de decidir si un SRT $\mathcal{R}$ pertenece a $\mathsf{T}(CL)$ es equivalente a decidir si existe un SRT $\mathcal{S} \sim_l \mathcal{R}^{\mu_{\mathcal{R}}^{com}}$ tal que $\mathcal{S} \in CL$.

En la próxima sección necesitaremos asegurar que la función ρ_μ que relaciona las reglas de un SRT $\mathcal{R}$ y su μ-contracción $\mathcal{R}^\mu$ es una biyección (es decir, que la μ-contracción de un SRT no *confunde* sus reglas de reescritura).

Definición 8.2.16 (SRT no μ-confuso) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT, $\mu \in M_\Sigma$ y $\mathcal{R}^\mu = (\Sigma^\mu, R^\mu)$. Decimos que $\mathcal{R}$ es* no μ-confuso *si $\rho_\mu : R \to R^\mu$ es una biyección.*

Ejemplo 8.2.17 *Consideremos el siguiente SRT $\mathcal{R}$:*

$$\mathtt{f(x,y)} \to \mathtt{g(a)}$$
$$\mathtt{f(x,y)} \to \mathtt{g(b)}$$

Si escogemos $\mu = \mu_{\mathcal{R}}^{com}$, obtenemos el siguiente SRT μ-contracto $\mathcal{R}^\mu$ de $\mathcal{R}$:

$$\mathtt{f}_\mu \to \mathtt{g}_\mu$$

Como podemos observar, la función ρ_μ no es inyectiva y, por tanto, no es una biyección y 'confunde' las reglas de $\mathcal{R}$ al μ-contraerlas. No obstante, si escogemos $\mu(\mathtt{g}) = \{1\}$, ρ_μ es ahora biyectiva.

Podemos definir una condición sintáctica sobre los SRTs que asegura esta propiedad (para toda función de reemplazamiento mayor que la canónica).

Definición 8.2.18 (SRT no confuso) *Un SRT $\mathcal{R} = (\Sigma, R)$ es* no confuso *si, para todo par de reglas distintas $l \to r, l' \to r' \in R$, no existe ningún renombramiento σ tal que $\sigma(l) = l'$.*

Observemos que, en particular, los SRTs ortogonales son no confusos. Para los SRTs no confusos, la función ρ_μ siempre es una biyección, tal como demostramos a continuación.

Proposición 8.2.19 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT lineal por la izquierda y no confuso. Entonces, para todo $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$, $\rho_\mu : R \to R^\mu$ es biyectiva.*

DEMOSTRACIÓN. Sólo necesitamos demostrar que ρ_μ es inyectiva puesto que, por definición, ya sabemos que es sobreyectiva. Por tanto, demostramos que, para cualquier par de reglas $\alpha, \alpha' \in R$, tenemos que, si $\rho_\mu(\alpha) = \rho_\mu(\alpha')$, entonces $\alpha = \alpha'$. Lo hacemos por contradicción: supongamos que existen dos reglas distintas $\alpha : l \to r$ y $\alpha' : l' \to r'$ tales que $\tau_\mu(l \to r) = \tau_\mu(l' \to r')$. Entonces, $\tau_\mu(l) = \tau_\mu(l')$, lo cual significa que podemos escribir $l = C[l_1, \ldots, l_n]$ y $l' = C[l'_1, \ldots, l'_n]$, donde $C[\,]$ es el contexto reemplazante máximo que corresponde a la $\equiv_{\tau_\mu}$-clase común a l y l'. Sin embargo, puesto que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$, se tiene que $l_1, \ldots, l_n, l'_1, \ldots, l'_n$, que son los subtérminos situados sobre ocurrencias no reemplazantes, son todos símbolos de variables. Dado que $\mathcal{R}$ es lineal por la izquierda, podemos asumir que $\{l_1, \ldots, l_n\} \cap \{l'_1, \ldots, l'_n\} = \emptyset$. Definiendo el renombramiento σ como sigue: $\sigma(l_i) = l'_i$ y $\sigma(l'_i) = l_i$ para $1 \leq i \leq n$, obtenemos $\sigma(l) = l'$. Por tanto, $\mathcal{R}$ no satisface la Definición 8.2.18. □

Por tanto, la condición sintáctica de no confusión es suficiente para garantizar, para cualquier función de reemplazamiento μ, que el SRT es no μ-confuso.

Finalmente, demostramos que el operador T preserva la ortogonalidad de los SRTs.

Teorema 8.2.20 *Si CL es una clase de SRTs ortogonales, $\mathsf{T}(CL)$ es una clase de SRTs ortogonales.*

DEMOSTRACIÓN. Inmediata, por definición de T y por el Teorema 5.3.10. □

Desafortunadamente, no podemos asegurar un resultado análogo para la clase de SRTs cuasi ortogonales.

El operador T preserva el caracter fuertemente secuencial de los SRTs.

Teorema 8.2.21 *Si CL es una clase de SRTs lineales por la izquierda y fuertemente secuenciales, para todo $\mathcal{R} \in \mathsf{T}(CL)$, $\mathcal{R}$ es fuertemente secuencial.*

DEMOSTRACIÓN. Inmediata, por definición de T y por el Teorema 8.2.6. □

8.2.3 Estrategias *sc* inducidas

En esta sección definimos el concepto de estrategia genérica *sc inducida* por una estrategia genérica. Empleamos los resultados anteriores para demostrar que dichas estrategias son *sc*-normalizantes si las estrategias originales son estrategias de reducción de índices.

Definición 8.2.22 (Estrategia genérica *sc* inducida) *Dada una estrategia genérica* $\mathbb{F}$ *(llamada estrategia* inductora*) para una clase CL de SRTs lineales por la izquierda, definimos su estrategia genérica* sc inducida, $\mathbb{F}^{ind}$ *para la clase*

$$VP^{ind}(CL) = \{(\mathcal{R}, \mu) \mid \mathcal{R} \in \mathsf{T}(CL) \wedge \mu \in m(\mathcal{R}) \wedge \rho_\mu \text{ es biyectiva}\}$$

como la estrategia genérica sc *que, a cada par válido* $(\mathcal{R}, \mu) \in VP^{ind}(CL)$*, le asocia la* μ*-estrategia (llamada también* inducida*)* $\mathbb{F}^\mu_\mathcal{R} = (\mathbb{P}^\mu_\mathcal{R}, \mathbb{R}^\mu_\mathcal{R})$*, donde, para todo* $t \in \mathcal{T}(\Sigma, V)$*:*

$$\begin{aligned} \mathbb{P}^\mu_\mathcal{R}(t) &= \nu_t^{-1}(\mathbb{P}_\mathcal{S}(\tau_\mu(t))) \\ \mathbb{R}^\mu_\mathcal{R}(t) &= \rho_\mu^{-1}(\mathbb{R}_\mathcal{S}(\tau_\mu(t))) \end{aligned}$$

siendo $\mathcal{S} \sim_l \mathcal{R}^\mu$*.*

Cada concreción del objeto especificado por la Definición 8.2.22 para un SRT $\mathcal{R}$ es una μ-estrategia para $\mathcal{R}$ porque, de acuerdo con el Teorema 5.7.10, cuando $\mu^{com}_\mathcal{R} \sqsubseteq \mu$, un término t es una forma μ-normal, si y sólo si $\tau_\mu(t)$ es una forma normal respecto a $\mathcal{R}^\mu$ y $\mathbb{F}_\mathcal{S}$ es una estrategia de reescritura para $\mathcal{S} \sim_l \mathcal{R}^\mu$.

Observación 8.2.23 *A diferencia de las clases* $VP^{sc}(CL)$ *y* $VP^{can}(CL)$*, la clase* $VP^{ind}(CL)$ *se refiere, en general, a más SRTs de los contenidos en la propia clase CL. Esto es así porque los SRTs implicados en la constitución de los pares válidos en* $VP^{ind}(CL)$ *proceden de la extensión* sc $\mathsf{T}(CL)$ *de CL (que puede verificar* $CL \subset \mathsf{T}(CL)$*) mientras que los SRTs involucrados en los pares válidos de* $VP^{sc}(CL)$ *o* $VP^{can}(CL)$ *proceden* directamente *de CL.*

El cálculo del componente $\mathbb{P}^\mu_\mathcal{R}(t)$ de una estrategia inducida (Definición 8.2.22) para un término t dado se realiza empleando τ_μ para pasar al ámbito del SRT $\mathcal{S}$ equivalente por la izquierda a $\mathcal{R}^\mu$, utilizando a continuación el componente $\mathbb{P}_\mathcal{S}$ de la estrategia original y volviendo al ámbito de $\mathcal{R}$ mediante ν_t^{-1}.

Observación 8.2.24 *Si podemos demostrar que la estrategia genérica* sc *inducida es estructural y está basada en la misma función de selección que la estrategia inductora correspondiente, no es necesario emplear la función de* μ*-contracción para calcular la posición del redex a reducir; basta con aplicar la función de selección sobre el conjunto de redexes reemplazantes del SRT* $\mathcal{R}$*. Por tanto, la estrategia inducida coincide con la restringida.*

Si la estrategia genérica $\mathbb{F}$ es estructural, entonces, bajo ciertas condiciones, su estrategia *sc inducida* también lo es.

Proposición 8.2.25 *Sea* $\mathbb{F}$ *una estrategia genérica estructural, para una clase CL de SRTs lineales por la izquierda, que está basada en la función de selección* $\widehat{\mathbb{P}}$*. Si,*

para todo $(\mathcal{R},\mu) \in VP^{ind}(CL)$ *tal que* $\mathcal{R} = (\Sigma, R)$, *y cada* $t \in \mathcal{T}(\Sigma, V)$ *y* $U \subseteq O^{\mu}(t)$ *tal que* $U \neq \emptyset$, *se tiene que* $\nu_t(\widehat{\mathbb{P}}(U)) = \widehat{\mathbb{P}}(\nu_t(U))$, *entonces* $\mathbb{F}^{ind}$ *es una estrategia* sc *estructural (basada en* $\widehat{\mathbb{P}}$*).*

DEMOSTRACIÓN. Por la Definición 8.2.22 y por hipótesis, tenemos que $\mathbb{P}^{\mu}_{\mathcal{R}}(t) = \nu_t^{-1}(\mathbb{P}_{\mathcal{S}}(\tau_\mu(t))) = \nu_t^{-1}(\widehat{\mathbb{P}}(O_{\mathcal{R}^\mu}(\tau_\mu(t))))$. Por el Lema 2.3.3 y el Corolario 5.7.8, los μ-redexes en un término t y los redexes en $\tau_\mu(t)$ están en correspondencia biunívoca, es decir, a cada $u \in O_{\mathcal{R}}(t)$ le corresponde una ocurrencia distinta $\nu_t(u) \in O_{\mathcal{R}^\mu}(\tau_\mu(t)) = O_{\mathcal{S}}(\tau_\mu(t))$ y sólo ésta. Por tanto, puesto que $\nu_t(O^{\mu}(t)) = O(\tau_\mu(t))$, tenemos que $\mathbb{P}^{\mu}_{\mathcal{R}}(t) = \nu_t^{-1}(\widehat{\mathbb{P}}(\nu_t(O^{\mu}_{\mathcal{R}}(t))))$. Por hipótesis, $\mathbb{P}^{\mu}_{\mathcal{R}}(t) = \nu_t^{-1}(\nu_t(\widehat{\mathbb{P}}(O^{\mu}_{\mathcal{R}}(t)))) = \widehat{\mathbb{P}}(O^{\mu}_{\mathcal{R}}(t))$. □

En general, a partir de una estrategia genérica estructural $\mathbb{F}$ no es posible expresar la estrategia genérica *sc* inducida (Definición 8.2.22) como una estrategia generica *sc* estructural (como en la Definición 8.1.14) si la función de selección $\widehat{\mathbb{P}}$ de la estrategia genérica estructural no conmuta (tal como se expresa en la Proposición 8.2.25) con cada ν_t para cada término t.

Ejemplo 8.2.26 *Consideremos la siguiente función de selección* $\widehat{\mathbb{P}}$*:*

$$\widehat{\mathbb{P}}(U) = \begin{cases} \{min_{\leq_L}(U)\} & \text{si } i.u = min_{\leq_L}(U) \text{ e } i \text{ es par} \\ \{max_{\leq_L}(U)\} & \text{en otro caso} \end{cases}$$

Si consideramos el término $t = f(x, s_1, s_2)$, *donde* s_1 *y* s_2 *son redexes y* $\mu(f) = \{2,3\}$, *entonces, de acuerdo con la Definición 8.2.22, para un SRT* $\mathcal{R}$ *tal que* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ *calculamos*

$$\begin{aligned} \mathbb{P}^{\mu}_{\mathcal{R}}(t) &= \nu_t^{-1}(\mathbb{P}_{\mathcal{S}}(\tau_\mu(t))) = \nu_t^{-1}(\widehat{\mathbb{P}}(O_{\mathcal{R}^\mu}(\tau_\mu(t)))) \\ &= \nu_t^{-1}(\widehat{\mathbb{P}}(\{1,2\})) = \nu_t^{-1}(\{2\}) \\ &= \{3\} \end{aligned}$$

Sin embargo, al utilizar la Proposición 8.2.25, obtenemos

$$\begin{aligned} \mathbb{P}^{\mu}_{\mathcal{R}}(t) &= \widehat{\mathbb{P}}(O^{\mu}_{\mathcal{R}}(t)) = \widehat{\mathbb{P}}(\{2,3\}) \\ &= \{2\} \end{aligned}$$

Luego existe una discordancia entre los redexes seleccionados por ambos métodos.

A continuación presentamos el resultado principal de este capítulo: dada una estrategia genérica de reducción de índices $\mathbb{F}$, la estrategia genérica *sc* inducida $\mathbb{F}^{ind}$ es una estrategia genérica *sc* de reducción de índices.

Teorema 8.2.27 *Si* $\mathbb{F}$ *es una estrategia genérica de reducción de índices para una clase CL de SRTs lineales por la izquierda y fuertemente secuenciales, entonces* $\mathbb{F}^{ind}$ *es una estrategia genérica* sc *de reducción de índices para* $VP^{ind}(CL)$.

DEMOSTRACIÓN. Sea $(\mathcal{R},\mu) \in VP^{ind}(CL)$ y $t \in \mathcal{T}(\Sigma,V)$ un término μ-reducible. Por el Teorema 5.7.10, $\tau_\mu(t)$ es reducible. Puesto que $\mathbb{F}_\mathcal{S}$ es una estrategia de reducción de índices para $\mathcal{S} \sim_l \mathcal{R}^\mu$, tenemos que $\mathbb{P}_\mathcal{S}(\tau_\mu(t)) = v \in \mathcal{J}(\tau_\mu(t)[\Omega_\mu]_v) = \mathcal{J}(\tau_\mu(t[\Omega]_u))$ si $v = \nu_t(u)$. De acuerdo con la Definición 8.2.22,

$$\mathbb{P}^\mu_\mathcal{R}(t) = \nu_t^{-1}(\mathbb{P}_\mathcal{S}(\tau_\mu(t))) = \nu_t^{-1}(v) = u \in \mathcal{I}^\mu(t[\Omega]_u).$$

Por tanto, dado que, por el Teorema 8.2.21, para todo $(\mathcal{R},\mu) \in VP^{ind}(CL)$, $\mathcal{R}$ es fuertemente secuencial, la concreción $\mathbb{F}^\mu_\mathcal{R}$ de $\mathbb{F}^{ind}$ es una μ-estrategia de reducción de índices para $\mathcal{R}$. □

Las estrategias *sc* inducidas definidas para clases de SRTs ortogonales disfrutan de las buenas propiedades de las estrategias de reducción de índices reemplazantes.

Teorema 8.2.28 *Si $\mathbb{F}$ es una estrategia genérica de reducción de índices para una clase CL de SRTs ortogonales y fuertemente secuenciales, entonces $\mathbb{F}^{ind}$ es una estrategia genérica* sc *estabilizante y* sc*-normalizante para $VP^{ind}(CL)$.*

DEMOSTRACIÓN. Por el Teorema 8.2.20, todo SRT de la clase $VP^{ind}(CL)$ es ortogonal. Por el Teorema 8.2.27, la estrategia $\mathbb{F}^{ind}$ es una estrategia genérica *sc* de reducción de índices para $VP^{ind}(CL)$. Por el Teorema 7.5.2 $\mathbb{F}^{ind}$ es estabilizante y por el Teorema 7.5.7 es *sc*-normalizante. □

Aunque los resultados sobre el carácter estabilizante y *sc*-normalizante de una estrategia *sc* de reducción de índices están dados para SRTs cuasi ortogonales (Teoremas 7.5.2 y 7.5.7), no podemos garantizar que la extensión *sc* de una clase de SRTs cuasi ortogonales sólo contenga SRTs cuasi ortogonales, debido a que el operador T sólo preserva la ortogonalidad. Por este motivo el Teorema 8.2.28 se limita a SRTs ortogonales.

Podemos dar un resultado aplicable a una clase más amplia de SRTs si nos limitamos a la reducción de términos normalizables.

Teorema 8.2.29 *Sea $\mathcal{R} = (\Sigma,R)$ un SRT lineal por la izquierda, no μ-confuso, fuertemente secuencial y uniformemente convergente a la raíz respecto a $\mu \in M_\Sigma$ tal que $\mu^{com}_\mathcal{R} \sqsubseteq \mu$. Sea $\mathbb{F}_\mathcal{S}$ una estrategia de reducción de índices para un SRT $\mathcal{S} \sim_l \mathcal{R}^\mu$ y sea t un término normalizante. Entonces, $\mathbb{F}^\mu$ obtiene una forma μ-normal de t.*

DEMOSTRACIÓN. Inmediata, empleando el Teorema 8.2.27 y el Teorema 7.5.8. □

8.3 Trabajos relacionados

El Teorema 8.1.17 incluye el resultado de Middeldorp "*parallel outermost* es estabilizante" puesto que, para todo SRT ortogonal $\mathcal{R} \in CL$, tenemos que $(\mathcal{R},\mu_\mathsf{T}) \in$

$VP^{ind}(CL)$ y $\rightarrow_{\mathcal{R}}=\hookrightarrow_{\mathcal{R}(\mu_{\mathsf{T}})}$. No obstante, el Teorema 8.1.17 mejora el resultado de Middeldorp puesto que, en general, evitaremos reducciones inútiles mediante las restricciones de reemplazamiento adicionales.

Capítulo 9

Implementación

En este capítulo abordamos el problema de la implementación de las estrategias de reescritura sensible al contexto basadas en la reducción de índices.

En la Sección 9.1 esbozamos la problemática general planteada por la implementación del proceso de reescritura lo que, en la práctica, lleva a concretar ésta en la implementación de una estrategia u otra.

En la Sección 9.2 revisamos la implementación de las estrategias de reducción de índices mediante *index trees*, una estructura de datos que puede asociarse a cualquier SRT ortogonal fuertemente secuencial.

En la Sección 9.3, demostramos que es posible aprovechar la implementación de una estrategia de reducción de índices para obtener una implementación de la estrategia *sc* inducida correspondiente.

9.1 Implementación de estrategias de reescritura

A pesar de su definición más o menos abstracta, cualquier estrategia de reescritura $\mathbb{F}_{\mathcal{R}}$ para un SRT $\mathcal{R}$ debe ser implementada (normalmente sobre arquitecturas convencionales, no especialmente diseñadas para realizar tareas de búsqueda de redexes y reemplazamientos) mediante determinadas estructuras de datos y algoritmos. Existen algoritmos de propósito general diseñados para implementar las distintas tareas involucradas en la realización efectiva del proceso de reescritura [HO82, O'D85]:

1. Localización de los redexes existentes en un término (*pattern matching*),

2. Selección del siguiente reemplazamiento que debe realizarse,

3. Realización del reemplazamiento.

En muchos casos, estas tareas se intercalan. Se han desarrollado algoritmos eficientes para implementar la fase 1, es decir, para localizar los redexes existentes en un término. Dichos algoritmos son aplicables a programas lineales por la izquierda [HO82, HO82b, O'D85, NWE97, SRR95].

Si deseamos obtener una implementación capaz de computar la forma normal de un término siempre que ésta exista, la definición de algoritmos para seleccionar el siguiente reemplazamiento que debe realizarse (la fase 2 anterior) es crítica. De hecho, este problema ha dado lugar a una gran cantidad de estudios teóricos y prácticos [Dur94, HL79, HL91, HO79, O'D77, O'D85, She94, SRR95, Str88, Str89, TSEP93] cuyos primeros exponentes fueron [HL79, O'D77]. En particular, Huet y Lévy proporcionaron un procedimiento completo de normalización para SRTs ortogonales y fuertemente secuenciales basado en la reducción de redexes situados sobre índices. Dicho procedimiento consta de los siguientes elementos [HL79, HL91]:

1. Una *estructura de datos*; un grafo dirigido acíclico (con algunos arcos adicionales) llamado *matching dag*, que viene determinado por el SRT concreto cuya ejecución se desea implementar. La misión de este grafo es guiar la búsqueda del redex que deberá ser reducido en cada paso de reducción. Sin embargo, no se trata de un simple autómata que detecta la presencia de un redex en el seno de una determinada expresión inicial (al estilo de los algoritmos presentados en [HO82, NWE97]). Cuando el *matching dag* indica la presencia de un redex en la expresión a reducir, *también indica que es ese redex el que debe ser reducido para normalizar dicha expresión* (de hecho se trata de un redex situado sobre un índice). Por tanto, el *matching dag* sirve para implementar las dos primeras tareas involucradas en el proceso de reducción.

2. Un *algoritmo* que recorre la expresión de entrada de acuerdo con las transiciones especificadas por el *matching dag*. El *matching dag* está diseñado de modo que cada símbolo del término a normalizar se inspecciona sólo una vez. A ello contribuyen los arcos adicionales añadidos sobre el grafo acíclico básico. Dichos arcos se denominan *arcos de fallo* pues su misión es manejar (eficientemente) la continuación del proceso de búsqueda del redex a reducir cuando un emparejamiento parcialmente completado resulta abortado por la lectura de un símbolo de la expresión a reducir con el cual no es posible continuar el emparejamiento en curso.

La construcción del *matching dag* es una tarea compleja. En [KM91], los autores conjeturaban que, en el caso general, su construcción es un problema NP-completo. Recientemente, Durand ha demostrado que, de hecho, es un problema co-NP-completo [Dur95]. Por tanto, varios autores han estudiado el problema de identificar subclases de SRTs para los cuales la construcción de un *matching dag*, o una estructura equiva-

lente, requiera un esfuerzo computacional inferior [O'D77, Dur94, MS97, Str89]. En este sentido, Strandh introduce los *index trees* [Str89]. Un *index tree* es un autómata de estado finito (y por tanto determinista, por oposición al *matching dag* de Huet y Lévy[1]). Durand ha demostrado que la clase de los SRTs (ortogonales) fuertemente secuenciales que admiten la definición de un *index tree* (que son llamados SRTs *bounded*, clase denotada por la letra B) coincide con la clase SS de los SRTs ortogonales y fuertemente secuenciales, es decir, $B = SS$ [Dur94]. Para una subclase de SS, los SRTs *forward branching* (FB), existe una clase especial de *index trees* (los *forward branching index trees*) que pueden construirse con una complejidad temporal cuadrática en el tamaño de las partes izquierdas del SRT (ver [Dur94] y también [MS97] donde se proponen algunas mejoras adicionales en el algoritmo de construcción). La clase FB está estrictamente contenida en SS: $FB \subset SS$ [Str89].

Otra línea de investigación se ha inspirado en un resultado de S. Thatte que establece la posibilidad de asociar un SRT BC a cualquier SRT de modo que sean semánticamente equivalentes [Tha85]. Durand y Salinier identificaron la subclase de SRTs fuertemente secuenciales cuyo carácter fuertemente secuencial resulta preservado por la transformación de Thatte [DS93, Sal95]. La implementación de los SRT BCs fuertemente secuenciales es más simple: Strandh demostró que la clase de los SRT BCs fuertemente secuenciales está incluida en FB. Recientemente, se ha demostrado [HLM98] que la implementación de los SRT BCs fuertemente secuenciales puede realizarse mediante árboles definicionales [Ant92], una estructura de datos aún más simple que los *index trees*.

Suponiendo la existencia de un *matching dag*, el coste asociado a la realización de un paso de reducción empleando el algoritmo de Huet y Lévy es lineal en el tamaño del término evaluado. Sin embargo, los cómputos no pueden realizarse de forma *incremental* en el caso general. Con esto queremos decir que, tras realizar el reemplazamiento de un redex situado sobre un índice, el proceso de búsqueda de un nuevo redex continúa desde la raíz del término, sin tener en cuenta la parte del término que ya fue inspeccionada y que no ha cambiado. Si podemos reanudar la búsqueda de un nuevo redex situado sobre un índice de forma *local*, es decir, sin volver a empezar en la raíz del término completo, parece claro que obtendremos mejoras en el coste global del proceso de normalización. Por esta razón, otras líneas de trabajo han incidido sobre este tema [O'D77, SPR89, TSEP93]. En particular, Strandh proporciona un algoritmo de *estabilización fuerte* para los *forward branching* SRTs que opera de forma incremental [Str89]. También Toyama et al. han definido los SRTs *transitivos* (TR), una subclase de SS para la cual podemos encontrar índices adecuados para ope-

[1] Sin embargo, tal como apuntan Huet y Lévy, es posible forzar el determinismo en un *matching dag*, ver [HL91], pag. 430.

rar con ellos de forma incremental [TSEP93]. En [Dur94] se demuestra que $FB = TR$.

En este capítulo mostramos que, dada una función de reemplazamiento μ, una implementación de una estrategia de reducción de índices $\mathbb{IF}_{\mathcal{R}}$ (para un SRT $\mathcal{R}$ ortogonal y fuertemente secuencial) empleando un *index tree* puede ser reconvertida de forma inmediata a un *index tree* que implementa la μ-estrategia inducida $\mathbb{IF}^{\mu}_{\mathcal{R}}$ correspondiente (para $\mathcal{S} \sim_l \mathcal{R}^{\mu}$).

9.2 Index trees

Un *index tree* $IT = (Q, \Sigma, \delta, q_0, F, \phi)$ es un autómata finito [HU79] constituido por un conjunto de estados Q, un alfabeto Σ, una función (parcial) de transferencia $\delta : Q \times \Sigma \rightarrow Q$, un estado inicial $q_0 \in Q$ y un conjunto de estados finales $F \subseteq Q$ [Str89, Dur94]. Además de la función de transferencia habitual δ, un *index tree* incorpora también una *función de fallo* $\phi : Q \rightarrow Q$. El conjunto de estados $Q = Q_{IP} \uplus F$, consta de pares $\langle \pi, u \rangle \in Q_{IP}$ denominados *index points* y estados finales $F = L_{\Omega}(\mathcal{R})$. Los *index points* se definen a partir de las nociones de ocurrencia fuertemente estable y término firme que presentamos a continuación.

Definición 9.2.1 [Dur94] *Sea $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$. La ocurrencia $u \in O(t)$ es una ocurrencia fuertemente estable de t si y sólo si $t|_u$ es fuertemente estable.*

Se dice que el Ω-término t es firme *si y sólo si $\exists u \in O_{\Omega}(t)$ tal que, $\forall v \in \overline{O}_{\Omega}(t)$, o bien v es fuertemente estable o bien $v < u$; en ese caso, a la ocurrencia u se le llama una extensión firme de t.*

Ahora estamos en condiciones de definir el concepto de *index point* que desempeña un papel esencial en la definición de los *index trees*.

Definición 9.2.2 (*Index point* [Dur94]) *Un* index point *es un par $\langle \pi, u \rangle$, donde $\pi \in L^{<}_{\Omega}(\mathcal{R})$ es un Ω-término firme y u es una extensión firme de π tal que $u \in \mathcal{I}(\pi)$.*

Escribimos $IP(\mathcal{R})$ para referirnos al conjunto de *index points* de un SRT $\mathcal{R}$. El alfabeto de símbolos que el autómata lee es precisamente la signatura Σ. Dado un estado $q = \langle \pi, u \rangle$ y un símbolo de función f, la *función de transferencia* proporciona el nuevo estado en el que se instalará el autómata:

$$\delta(\langle \pi, u \rangle, f) = \langle \pi[f(\widetilde{\Omega})]_u, v \rangle$$

o simplemente

$$\delta(\langle \pi, u \rangle, f) = \langle \pi[f(\widetilde{\Omega})]_u \rangle$$

si la transición lleva a un estado final $\pi[f(\widetilde{\Omega})]_u \in L_{\Omega}(\mathcal{R})$.

El estado inicial es $q_0 = \langle \Omega, \epsilon \rangle \in Q_{IP}$ (que trivialmente es un *index point*).

La *función de fallo*, ϕ, es $\phi(q) = q'$ si y sólo si q' es un *punto de recuperación* (*failure point*) inmediato de q. Los puntos de recuperación son nodos del *index tree* que se encargarán de manejar las situaciones de emparejamiento parcial fallido, forzando la búsqueda de un nuevo emparejamiento sobre un subtérmino del término inspeccionado.

Definición 9.2.3 (Punto de recuperación [Dur94]**)** *Sea* $p = \langle \pi, w \rangle$ *un* index point *tal que* $\pi \neq \Omega$. *Un* index point $q = \langle \pi', v \rangle$ *es un* punto de recuperación *de* p *si y sólo si existe una* $u \neq \epsilon$ *tal que* $w = u.v$ *y* $\pi' = \pi|_u$. *Un punto de recuperación* q *de* p *es el* punto de recuperación inmediato *de* p *si y sólo si todo punto de recuperación de* p *distinto a* q *es un punto de recuperación de* q.

Una vez definido el concepto de punto de recuperación, la función de fallo ϕ se define como sigue: para todo *index point* p,

$\phi(p) = q$ si y sólo si q es el punto de recuperación inmediato de p

Para un estado final, tanto la función de transición como la función de fallo permanecen indefinidas.

Los SRTs ortogonales que admiten la definición de un *index tree* que representa sus reglas de reescritura (recogiendo todos los esquemas de redexes asociados al SRT, y sólo éstos, como estados finales del *index tree*) se denominan SRTs *bounded* (B). Durand ha demostrado que dicha clase coincide con la clase de SRTs ortogonales fuertemente secuenciales ($B = SS$). La demostración se realiza mostrando que hay una correspondencia inmediata entre el *matching dag* de Huet y Lévy y los *index trees* de Strandh.

Ejemplo 9.2.4 *Consideremos el siguiente SRT* bounded *(tomado de [Dur94]):*

```
f(g(x,a),a) → a        g(b,b) → a
f(g(a,x),b) → a
```

El index tree *asociado a este SRT es el presentado en la Figura 9.1.*

9.3 Implementación de μ-estrategias inducidas

Dado un SRT ortogonal $\mathcal{R} \in \mathsf{T}(CL)$ para una clase $CL \subseteq SS$, deseamos asociarle un *index tree* (inducido) $IT = (Q, \Sigma, \delta, q_0, F, \phi)$, que implemente la μ-estrategia $\mathbb{F}^{\mu}_{\mathcal{R}}$ inducida a partir de la estrategia de reducción de índices $\mathbb{F}_{\mathcal{S}}$ para un SRT $\mathcal{S} \sim_l \mathcal{R}^{\mu}$ de CL (ver la Definición 8.2.22) cuya implementación se realiza mediante un *index tree* $IT^{\mu} = (Q^{\mu}, \Sigma^{\mu}, \delta^{\mu}, q_0^{\mu}, F^{\mu}, \phi^{\mu})$. Veremos que esto es posible después de establecer algunos resultados auxiliares.

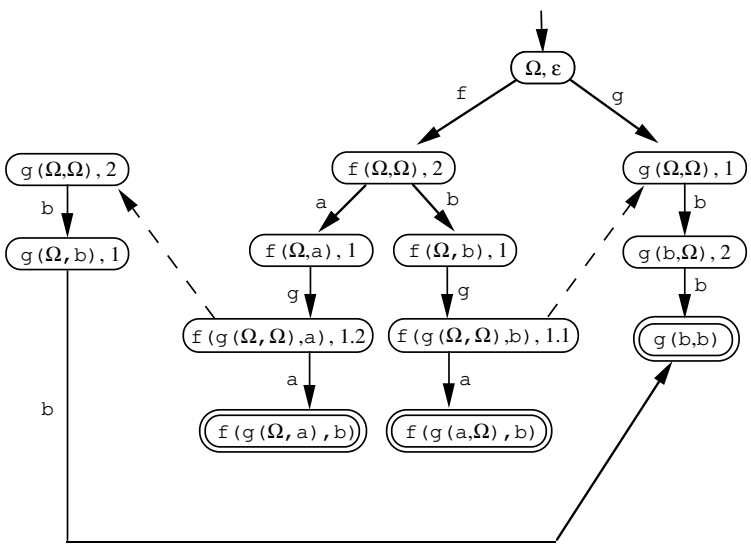

Figura 9.1: Ejemplo de *index tree*

Notación 9.3.1 *En las proposiciones siguientes, dados los conjuntos de términos* $T \subseteq \mathcal{T}(\Sigma \cup \{\Omega\}, V)$ *y* $T^\mu \subseteq \mathcal{T}(\Sigma^\mu \cup \{\Omega_\mu\}, V)$*, denotamos como* $T \simeq_{\tau_\mu} T^\mu$ *el hecho de que la restricción de* $\tau_\mu : \mathcal{T}(\Sigma \cup \{\Omega\}, V) \to \mathcal{T}(\Sigma^\mu \cup \{\Omega_\mu\}, V)$ *al conjunto* T *es una biyección* $\tau_\mu : T \to T^\mu$ *entre* T *y* T^μ*.*

A continuación demostramos que τ_μ es una biyección entre los conjuntos de esquemas de redexes asociados a un SRTE $\mathcal{R}$ y su μ-contracción $\mathcal{R}^\mu$ (siempre que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$).

Proposición 9.3.2 *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRTE y* $\mu \in M_\Sigma$ *tal que* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$*. Entonces* $L_\Omega(\mathcal{R}) \simeq_{\tau_\mu} L_{\Omega_\mu}(\mathcal{R}^\mu)$*.*

DEMOSTRACIÓN. Tenemos que $L_{\Omega_\mu}(\mathcal{R}^\mu) = \tau_\mu(L_\Omega(\mathcal{R}))$ por definición de $\mathcal{R}^\mu$. Entonces, la sobreyectividad es inmediata. Puesto que, para todo $l, l' \in L_\Omega(\mathcal{R})$, tenemos que $l, l' \in \mathcal{T}(\Sigma \cup \{\Omega\})$ y $\mu_l \sqcup \mu_{l'} \sqsubseteq \mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ (recordemos la Observación 5.7.1), por el Corolario 5.7.11, $l = l'$ si y sólo si $\tau_\mu(l) = \tau_\mu(l')$. Esto concluye la demostración. □

De forma análoga, existe una biyección entre los preredexes de un SRTE y su μ-contracción.

Proposición 9.3.3 *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRTE y* $\mu \in M_\Sigma$ *tal que* $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$*. Entonces* $L^<_\Omega(\mathcal{R}) \simeq_{\tau_\mu} L^<_{\Omega_\mu}(\mathcal{R}^\mu)$*.*

DEMOSTRACIÓN. En primer lugar, demostramos que $\pi \in L^<_\Omega(\mathcal{R})$ implica que $\tau_\mu(\pi) \in L^<_{\Omega_\mu}(\mathcal{R}^\mu)$, es decir, que la restricción de τ_μ a $L^<_\Omega(\mathcal{R})$ es una función de $L^<_\Omega(\mathcal{R})$ a

$L^{<}_{\Omega_\mu}(\mathcal{R}^\mu)$. Por definición, $\pi \in L^{<}_{\Omega}(\mathcal{R})$ si y sólo si $\pi < l$ para algún $l \in L_\Omega(\mathcal{R})$ y $\pi \neq \Omega$. Si $\pi \neq \Omega$, entonces $\tau_\mu(\pi) \neq \Omega_\mu$. Por otro lado, por la Proposiciones 4.1.5 y 4.1.10, $\mu_\pi \sqsubseteq \mu$. Dado que $\mu_l \sqsubseteq \mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, por la Proposición 5.7.4, $\tau_\mu(\pi) < \tau_\mu(l) \in L_{\Omega_\mu}(\mathcal{R}^\mu)$. Por tanto, $\tau_\mu(\pi) \in L^{<}_{\Omega_\mu}(\mathcal{R}^\mu)$.

Ahora demostramos la sobreyectividad. Por definición, $\tau_\mu(\pi) \in L^{<}_{\Omega_\mu}(\mathcal{R}^\mu)$ si y sólo si $\tau_\mu(\pi) < \tau_\mu(l)$ para algún $l \in L_\Omega(\mathcal{R})$ y $\tau_\mu(\pi) \neq \Omega_\mu$. Debemos demostrar que π es un preredex en $L^{<}_{\Omega}(\mathcal{R})$ (para algún π en su $\equiv_{\tau_\mu}$-clase). Como el único π cuya imagen por τ_μ es Ω_μ es Ω, se sigue que $\pi \neq \Omega$. Consideremos que $\pi = C[\widetilde{\Omega}]$, donde $C[\,]$ es el contexto reemplazante máximo de esta $\equiv_{\tau_\mu}$-clase. Puesto que $l \in \mathcal{T}(\Sigma \cup \{\Omega\})$, tenemos $\pi \in \mathcal{T}(\Sigma \cup \{\Omega\})$. De no ser así, si hay una ocurrencia de variable en π debe ser reemplazante y, por tanto, sería imposible que $\tau_\mu(\pi) < \tau_\mu(l)$, porque no hay ocurrencias de variables en un esquema de redex l. Por este motivo, y aplicando la Proposición 5.7.5, $\pi < l$; es decir, $\pi \in L^{<}_{\Omega}(\mathcal{R})$.

Finalmente demostraremos la inyectividad. Hay que probar que, para todo $\pi, \pi' \in L^{<}_{\Omega}(\mathcal{R})$, $\tau_\mu(\pi) = \tau_\mu(\pi') \Rightarrow \pi = \pi'$. Puesto que $\pi, \pi' \in \mathcal{T}(\Sigma \cup \{\Omega\})$, y $\pi, \pi' \in L^{<}_{\Omega}(\mathcal{R})$, por las Proposiciones 4.1.5 y 4.1.10, $\mu_\pi \sqsubseteq \mu^{com}_{\mathcal{R}}$ y $\mu_{\pi'} \sqsubseteq \mu^{com}_{\mathcal{R}}$, es decir, $\mu_\pi \sqcup \mu_{\pi'} \sqsubseteq \mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Empleando el Corolario 5.7.11, concluimos la demostración. □

Los resultados anteriores permiten demostrar la existencia de una biyección entre los conjuntos de *index points* asociados a $\mathcal{R}$ y $\mathcal{R}^\mu$.

Proposición 9.3.4 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Entonces $IP(\mathcal{R}) \simeq IP(\mathcal{R}^\mu)$.*

DEMOSTRACIÓN. Definimos una biyección entre $IP(\mathcal{R})$ e $IP(\mathcal{R}^\mu)$ mediante la función $\langle \pi, u\rangle \mapsto \langle \tau_\mu(\pi), \nu_\pi(u)\rangle$. En primer lugar demostramos que, si $\langle \pi, u\rangle \in IP(\mathcal{R})$, entonces $\langle \tau_\mu(\pi), \nu_\pi(u)\rangle \in IP(\mathcal{R}^\mu)$. Dado que $u \in \mathcal{I}(\pi)$ y $\pi \in L^{<}_{\Omega}(\mathcal{R})$, debe tenerse $u \in O^\mu(\pi)$. De no ser así, puesto que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, $\pi|_u$ correspondería a una variable de la regla que introduce el preredex $l \in L^{<}_{\Omega}(\mathcal{R})$ para el cual $\pi < l$. Pero, en ese caso, u no puede ser un índice de π porque $\pi[\bullet]_u \uparrow l$. Por tanto, $u \in \mathcal{I}^\mu(\pi)$ y, de acuerdo con la Proposición 8.2.2, $\nu_\pi(u)$ es un índice de $\tau_\mu(\pi)$. A continuación demostramos que $\nu_\pi(u)$ es una extensión firme de $\tau_\mu(\pi)$ que, a su vez, es un término firme. Procediendo por contradicción, supongamos que $\nu_\pi(u)$ no es una extensión firme de $\tau_\mu(\pi)$. Entonces existe una $u' \in \overline{O}_\Omega(\pi)$, con $u \parallel u'$, tal que $\tau_\mu(\pi|_{u'})$ no es fuertemente estable (observemos que esta descripción es correcta puesto que, siendo $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, tenemos que $\overline{O}_\Omega(\pi) = \overline{O}^\mu_\Omega(\pi)$). Por tanto, es un término frágil; es decir, $\omega_\mu(\tau_\mu(\pi|_{u'})) = \Omega_\mu$ y, por la Proposición 5.7.16, $\omega(\pi|_{u'}) = \Omega$. En ese caso, $\pi|_u$ no es firme y u no es una extension firme de π; es decir, $\langle \pi, u\rangle$ no es un *index point*.

Ahora demostramos, por contradicción, que $\langle \tau_\mu(\pi), \nu_\pi(u)\rangle \in IP(\mathcal{R}^\mu)$ implica que $\langle \pi, u\rangle \in IP(\mathcal{R})$. Si u no es una extension firme de π, existe una $u' \in \overline{O}_\Omega(\pi)$, con $u \parallel u'$, tal que $\pi|_{u'}$ no es fuertemente estable. Por tanto $\omega(\pi|_{u'}) = \Omega$ y, por la Proposición

5.7.15, tenemos que $\tau_\mu(\pi|_{u'}) \rightarrow^*_{\Omega_\mu} \Omega_\mu$. Puesto que $u' \in O^\mu(\pi)$, por la Proposición 5.2.8, tenemos que $\tau_\mu(\pi|_{u'}) = \tau_\mu(\pi)|_{\nu_\pi(u')}$; es decir, no es fuertemente estable y, por tanto, $\langle \tau_\mu(\pi), \nu_\pi(u)\rangle$ no es un *index point*.

De acuerdo con esto, concluimos la demostración empleando el hecho de que $L^<_\Omega(\mathcal{R}) \simeq_{\tau_\mu} L^<_{\Omega_\mu}(\mathcal{R}^\mu)$ (Proposición 9.3.3) y $O^\mu(\pi) \simeq_{\nu_\pi} O(\tau_\mu(\pi))$. □

Ahora podemos establecer los resultados principales de este capítulo: la 'equivalencia' de la implementación de una estrategia de reducción de índices y la implementación de su μ-estrategia inducida cuando empleamos *index trees* para localizar los índices.

Teorema 9.3.5 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Sea $\mathcal{S} \sim_l \mathcal{R}^\mu$ un SRT cuyo* index tree *asociado es $IT^\mu = (Q^\mu, \Sigma^\mu, \delta^\mu, q^\mu_0, F^\mu, \phi^\mu)$. Entonces, $IT = (Q, \Sigma, \delta, q_0, F, \phi)$ donde:*

$Q = \{\langle \pi, u\rangle \mid \langle \pi, u\rangle \in IP(\mathcal{R}) \wedge \langle \tau_\mu(\pi), \nu_\pi(u)\rangle \in Q^\mu_{IP}\} \cup \tau^{-1}_\mu(F^\mu)$,

$\delta(\langle \pi, u\rangle, f) = \langle \pi', u'\rangle$ *si y sólo si* $\langle \pi, u\rangle \in Q \wedge \delta^\mu(\langle \tau_\mu(\pi), \nu_\pi(u)\rangle, f_\mu) = \langle \tau_\mu(\pi'), \nu_{\pi'}(u')\rangle$,

$\phi(\langle \pi, u\rangle) = \langle \pi', u'\rangle$ *si y sólo si* $\langle \pi, u\rangle \in Q \wedge \phi^\mu(\langle \tau_\mu(\pi), \nu_\pi(u)\rangle) = \langle \tau_\mu(\pi'), \nu_{\pi'}(u')\rangle$

es un index tree *para $\mathcal{R}$.*

DEMOSTRACIÓN. Por la Proposición 9.3.4, concluimos que el componente Q_{IP} de Q está bien definido. La Proposición 9.3.2 asegura que los estados finales $F \subseteq Q$ están bien definidos, puesto que $F^\mu = L_{\Omega_\mu}(\mathcal{R}^\mu)$. Para demostrar que δ está bien definida, debemos asegurarnos de que es determinista; es decir, $\forall q \in Q, \forall f \in \Sigma$, si $\delta(q, f) = p$ y $\delta(q, f) = p'$, entonces $p = p'$. Esto es inmediato puesto que δ^μ es una función y, por tanto, determinista. La función ϕ también es determinista, debido a que $L^<_\Omega(\mathcal{R}) \simeq_{\tau_\mu} L^<_{\Omega_\mu}(\mathcal{R}^\mu)$ (Proposición 9.3.3). Ahora demostramos que, si $\phi(\langle \pi, u\rangle) = \langle \pi', u'\rangle$, entonces $\langle \pi', u'\rangle$ es un punto de recuperación inmediato de $\langle \pi, u\rangle$. Por la Proposición 9.3.4, $\langle \pi', u'\rangle$ es un *index point*. Puesto que, por la Definición 9.2.3, existe un v tal que $u = v.u'$ y $\tau_\mu(\pi)|_{\nu_\pi(u')} = \tau_\mu(\pi')$, concluimos que $\langle \pi', u'\rangle$ es un punto de recuperación. No es difícil ver que, además, debe ser el punto de recuperación inmediato de $\langle \pi, u\rangle$. □

De acuerdo con el Teorema 9.3.5, podemos obtener una implementación para una μ-estrategia inducida $\mathbb{F}^\mu_{\mathcal{R}}$ para un SRT $\mathcal{R}$ a partir de la implementación correspondiente a la estrategia de reducción de índices $\mathbb{F}_{\mathcal{S}}$ para un SRT $\mathcal{S} \sim_l \mathcal{R}^\mu$. De hecho, la correspondencia entre las implementaciones es todavía más estrecha.

Teorema 9.3.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Sea $IT = (Q, \Sigma, \delta, q_0, F, \phi)$ un* index tree *para $\mathcal{R}$. Sea $\mathcal{S} \sim_l \mathcal{R}^\mu$ un SRT. Entonces, $IT^\mu = (Q^\mu, \Sigma^\mu, \delta^\mu, q^\mu_0, F^\mu, \phi^\mu)$, donde*

$Q^\mu = \{\langle \tau_\mu(\pi), \nu_\pi(u)\rangle \mid \langle \pi, u\rangle \in Q_{IP}\} \cup \tau_\mu(F) = Q^\mu_{IP} \cup F^\mu$.

$\delta^\mu(\langle \tau_\mu(\pi), \nu_\pi(u)\rangle, f_\mu) = \langle \tau_\mu(\pi'), \nu_{\pi'}(u')\rangle$ *si y sólo si* $\delta(\langle \pi, u\rangle, f) = \langle \pi', u'\rangle$,

$\delta^{\mu}(\langle\tau_{\mu}(\pi), \nu_{\pi}(u)\rangle) = \langle\tau_{\mu}(\pi'), \nu_{\pi'}(u')\rangle$ *si y sólo si* $\phi(\langle\pi, u\rangle) = \langle\pi', u'\rangle$ *es un* index tree *para* $\mathcal{S}$.

DEMOSTRACIÓN. Similar a la demostración del Teorema 9.3.5. □

9.4 Trabajos relacionados

Algunas propuestas interesantes de implementación de lenguajes de programación que permiten especificar restricciones de reemplazamiento sintácticas en los programas pueden estudiarse en [GKM87, OF97]. Sin embargo, el enfoque de estas propuestas de implementación es distinto al presentado aquí. Nuestra investigación se ha centrado en el estudio de la implementación de estrategias de reducción de índices (que se demuestran μ-normalizantes cuando μ es menos restrictivo que la correspondiente función de reemplazamiento canónica) mediante estructuras de datos y algoritmos, de modo que se puedan simular en una máquina convencional las operaciones básicas de reescritura (búsqueda de redexes y reemplazamiento) que no pueden realizarse directamente sobre una máquina así. Sin embargo, en [GKM87] se propone una implementación sobre una máquina de reducción de grafos (la *Rewrite Rule Machine*, o *RRM*) que permite realizar directamente las operaciones anteriores. Además, dicha máquina incorpora la posibilidad de permitir o inhibir los reemplazamientos que pueden realizarse sobre los (sub)términos a evaluar (mediante el *Rewrite Inhibit flag* y el *Subterm Inhibited flag* [GKM87]). Esto permitiría implementar directamente las restricciones de reemplazamiento sensibles al contexto sobre dicha máquina de reducción. La propuesta de [OF97] utiliza una máquina de reducción abstracta, la *TRAM* (*Term Rewriting Abstract Machine*), aunque la forma de incorporar las restricciones de reemplazamiento no es tan directa como en el caso anterior. No existe ningún *bit* que permita simular directamente la presencia o ausencia de restricciones de reemplazamiento. En su lugar, la máquina maneja directamente listas de ocurrencias (reemplazantes) que dirigen la evaluación del término (*strategy lists*).

Capítulo 10

Ejemplos de aplicación

En este capítulo ofrecemos algunos ejemplos de aplicación del método presentado en el Capítulo 8 para inducir estrategias genéricas *sc* para una clase $VP^{ind}(CL)$ de pares válidos a partir de estrategias genéricas para una clase de SRTs CL.

La Sección 10.1 analiza la extensión *sc* de los SRTs lineales por la izquierda y normales por la izquierda ($\overline{LN}$) proporcionando una caracterización sintáctica para ésta y demostrando que la clase extendida $\mathsf{T}(\overline{LN})$ es un superconjunto propio de $\overline{LN}$. Como consecuencia, la estrategia *sc* inducida correspondiente (obtenida a partir de $\mathbb{IF}_{lo}$, que es una estrategia genérica de reducción de índices para todo SRT en $\overline{LN}$) es una estrategia genérica *sc* de reducción de índices para una clase de SRTs más amplia que $\overline{LN}$ (Sección 10.1.2).

En la Sección 10.2 consideramos la extensión *sc* de la clase $\overline{GR}$ de los SRTs básicos. Damos su caracterización sintáctica y demostramos que es un superconjunto propio de $\overline{GR}$. Demostramos también que cualquier estrategia genérica para la clase $\overline{GR}$ induce una estrategia genérica *sc* de reducción de índices para la clase extendida.

En la Sección 10.3 mostramos también las limitaciones de nuestro método al analizar la extensión *sc* de la clase *FB* de SRTs *forward-branching*. Veremos que la extensión *sc* de dicha clase *no* añade nada a la misma.

10.1 Extensión de los SRTs normales por la izquierda

Una clase interesante de SRTs fuertemente secuenciales es la de los SRTs normales por la izquierda. Para SRTs ortogonales y normales por la izquierda (LN), la estrategia *leftmost-outermost*, $\mathbb{IF}_{lo}$, es una estrategia genérica de reducción de índices [O'D77, O'D85]. Este resultado ha sido extendido a SRTs lineales por la izquierda y normales

por la izquierda ($\overline{LN}$) por Toyama [Toy92].

10.1.1 Caracterización sintáctica

Demostramos que $\mathsf{T}(\overline{LN})$, la extensión *sc* de los SRTs lineales por la izquierda y normales por la izquierda, puede caracterizarse mediante una condición sintáctica simple que presentamos a continuación.

Definición 10.1.1 (SRTE μ-normal por la izquierda) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE y $\mu \in M_\Sigma$. Decimos que $\mathcal{R}$ es* μ-normal por la izquierda *si para cada $l \in L(\mathcal{R})$, y para todo $u \in O_\Sigma(l)$ y $v \in O^\mu(l)$, siempre que $v \leq_L u$ se verifica $v \in O_\Sigma(l)$.*

De forma análoga a los SRTEs normales por la izquierda, un SRTE es μ-normal por la izquierda si no existen ocurrencias de variables *reemplazantes* a la izquierda de los símbolos de función de las *lhs*'s del SRTE.

Proposición 10.1.2 *Si $\mathcal{R}$ es μ-normal por la izquierda y $\mu' \sqsubseteq \mu$, entonces $\mathcal{R}$ es μ'-normal por la izquierda.*

DEMOSTRACIÓN. Inmediata, puesto que, por la Proposición 3.1.12, $O^{\mu'}(t) \subseteq O^\mu(t)$ para todo t. □

Si tomamos $\mu = \mu_\mathsf{T}$, obtenemos la definición usual de SRT normal por la izquierda. Por tanto, todo SRT normal por la izquierda es μ-normal por la izquierda respecto a cualquier función de reemplazamiento μ (aplicando la Proposición 10.1.2, puesto que $\mu \sqsubseteq \mu_\mathsf{T}$).

Ejemplos de SRTEs μ-normales por la izquierda que no son normales por la izquierda son los SRTs de los Ejemplos 1.1.2 y 3.4.4 (respecto a sus respectivas funciones de reemplazamiento).

A continuación demostramos que todo SRT lineal por la izquierda y μ-normal por la izquierda pertenece a la extensión *sc* de los SRTs normales por la izquierda.

Proposición 10.1.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_\mathcal{R} \sqsubseteq \mu$. Si $\mathcal{R}$ es μ-normal por la izquierda, entonces $\mathcal{R} \in \mathsf{T}(\overline{LN})$ y $\mu \in m(\mathcal{R})$.*

DEMOSTRACIÓN. Demostramos que cualquier SRT $\mathcal{S} \sim_l \mathcal{R}^\mu$ es normal por la izquierda por contradicción. Supongamos que, dada una regla $l \to r \in R$, la regla $\tau_\mu(l) \to \tau_\mu(r)$ no es normal por la izquierda. Entonces existen $v \in O_V(\tau_\mu(l))$ y $v' \in O_{\Sigma^\mu}(\tau_\mu(l))$ tales que $v <_L v'$. Sean $u, u' \in O^\mu(l)$ las ocurrencias de l tales que $v = \nu_l(u)$ y $v' = \nu_l(u')$. Claramente, a partir de la definición de ν_l y $\leq_L$, tenemos que $u <_L u'$ y también $u \in O_V(l)$ y $u' \in O_\Sigma(l)$, es decir, llegamos a una contradicción. Por tanto, $\mathcal{R} \in \mathsf{T}(\overline{LN})$ y $\mu \in m(\mathcal{R})$. □

A su vez, todo SRT perteneciente a la extensión *sc* de los SRTs lineales por la izquierda y normales por la izquierda es μ-normal por la izquierda.

Proposición 10.1.4 *Si $\mathcal{R} \in \mathsf{T}(\overline{LN})$, entonces, para todo $\mu \in m(\mathcal{R})$, $\mathcal{R}$ es μ-normal por la izquierda.*

DEMOSTRACIÓN. Sea $\mathcal{R} = (\Sigma, R) \in \mathsf{T}(\overline{LN})$. Si $\mu \in m(\mathcal{R})$, entonces existe un $\mathcal{S} \sim_l \mathcal{R}^\mu$ tal que $\mathcal{S} \in \overline{LN}$, es decir, $\mathcal{S}$ es normal por la izquierda. Consideremos la μ-contracción $\tau_\mu(l)$ de la *lhs* l de una regla $l \to r \in R$. Sea $v \in O_{\Sigma^\mu}(\tau_\mu(l))$ tal que, para todo $v' \in O(\tau_\mu(l))$, $v' \leq_L v$ implica que $v \in O_{\Sigma^\mu}(\tau_\mu(l))$. Sean $u, u' \in O^\mu(l)$ tales que $v = \nu_l(u)$ y $v' = \nu_l(u')$. Dado que τ_μ preserva (sólo) los subtérminos reemplazantes, si $v' \in O_{\Sigma^\mu}(\tau_\mu(l))$, entonces $u' \in O_\Sigma(l)$. Con esto concluimos la demostración. □

Por tanto, las Proposiciones 10.1.3 y 10.1.4 implican el siguiente resultado.

Teorema 10.1.5 *$\mathsf{T}(\overline{LN})$ coincide con el conjunto de los SRTs lineales y μ-normales por la izquierda $\mathcal{R}$ tales que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$.*

A continuación demostramos que $\mathsf{T}(\overline{LN})$ es una extensión *sc* propia de $\overline{LN}$.

Proposición 10.1.6 *$\overline{LN} \subset \mathsf{T}(\overline{LN})$.*

DEMOSTRACIÓN. Puesto que T es un operador de cierre, $\overline{LN} \subseteq \mathsf{T}(\overline{LN})$. El SRT del Ejemplo 3.4.4 (que es μ-normal por la izquierda pero no es normal por la izquierda) confirma que la extensión *sc* incluye $\overline{LN}$ de forma estricta. □

10.1.2 La estrategia inducida

Nuestra primera observación sobre la estrategia genérica *sc* $\mathbb{F}_{lo}^{ind}$ para SRTs lineales por la izquierda y μ-normales por la izquierda es que $\mathbb{F}_{lo}^{ind}$ es una estrategia genérica *sc* estructural. El redex que se encuentra en el extremo izquierdo de t está situado sobre la ocurrencia $u = min_{\leq_L}(O_{\mathcal{R}}(t))$ que es un índice de $t[\Omega]_u$ si $\mathcal{R}$ es un SRT lineal por la izquierda y normal por la izquierda. Primero demostramos que $min_{\leq_L}$ conmuta con ν:

Proposición 10.1.7 *Sea $t \in \mathcal{T}(\Sigma, V)$, $\mu \in M_\Sigma$ y $U \subseteq O^\mu(t)$ tal que $U \neq \emptyset$. Entonces, $\nu_t(min_{\leq_L}(U)) = min_{\leq_L}(\nu_t(U))$.*

DEMOSTRACIÓN. Por inducción sobre $min_{\leq_L}(U)$. Si $\epsilon = min_{\leq_L}(U)$, dado que $\epsilon \in \nu_t(U)$, tenemos $\nu_t(min_{\leq_L}(U)) = \nu_t(\epsilon) = \epsilon = min_{\leq_L}(\nu_t(U))$.

Si $i.u = min_{\leq_L}(U)$, sea $U = U_i \uplus \overline{U}_i$, donde $U_i = \{i.u' \in U\} = i.U_i'$ y $\overline{U}_i = U \backslash U_i$. Por definición de $\leq_L$, $min_{\leq_L}(U) = min_{\leq_L}(U_i) = min_{\leq_L}(i.U_i') = i.min_{\leq_L}(U_i')$, donde $min_{\leq_L}(U_i') = u$.

Por otro lado, si $t = f(\tilde{t})$, entonces $\nu_t(min_{\leq_L}(U)) = \nu_t(i.u) = |\mathbb{N}_i^+ \cap \mu(f)|.\nu_{t_i}(u) = j.\nu_{t_i}(u)$. Por la hipótesis de inducción, $\nu_{t_i}(u) = \nu_{t_i}(min_{\leq_L}(U_i')) = min_{\leq_L}(\nu_{t_i}(U_i'))$. Por tanto, $\nu_t(min_{\leq_L}(U)) = j.min_{\leq_L}(\nu_{t_i}(U_i')) = min_{\leq_L}(j.\nu_{t_i}(U_i')) = min_{\leq_L}(\nu_t(U_i))$. Ahora notemos que, para todo $p.\bar{u} \in \overline{U}_i$, se tiene que $i < p$. Por tanto, puesto que $\nu_t(p.\bar{u}) = q.\nu_{t_p}(u)$ también verifica $j < q$, podemos añadir $\nu_t(\overline{U}_i)$ a $\nu_t(U_i)$ y escribir: $min_{\leq_L}(\nu_t(U_i)) = min_{\leq_L}(\nu_t(U_i) \uplus \nu_t(\overline{U}_i)) = min_{\leq_L}(\nu_t(U))$. □

Por tanto, de acuerdo con las Proposiciones 10.1.7 y 8.2.25, concluimos que la estrategia genérica *leftmost-outermost* $\mathbb{F}_{lo}$ induce una estrategia genérica *sc* estructural $\mathbb{F}_{lo}^{ind}$.

Teorema 10.1.8 *$\mathbb{F}_{lo}^{ind}$ es una estrategia genérica* sc *estructural de reducción de índices (basada en $min_{\leq_L}$) para $VP^{ind}(\overline{LN})$.*

DEMOSTRACIÓN. Dado que $\mathbb{F}_{lo}$ es una estrategia genérica de reducción de índices para $\overline{LN}$, por el Teorema 8.2.27 $\mathbb{F}_{lo}^{ind}$ es una estrategia genérica *sc* de reducción de índices para $VP^{ind}(\overline{LN})$. La Proposición 10.1.7 y la Proposición 8.2.25 permiten afirmar que $\mathbb{F}_{lo}^{ind}$ es genérica y que, de forma análoga a $\mathbb{F}_{lo}$, está basada en la función de selección $\{min_{\leq_L}\}$. □

Una vez obtenido este resultado, la estrategia inducida $\mathbb{F}_{lo}^{ind}$ hereda las buenas propiedades de la estrategia inductora $\mathbb{F}_{lo}$ sobre la clase de SRTs extendida.

Teorema 10.1.9 *$\mathbb{F}_{lo}^{ind}$ es estabilizante y* cs*-normalizante para $VP^{ind}(LN)$.*

DEMOSTRACIÓN. Inmediata, empleando los Teoremas 10.1.8 y 8.2.28. □

Cuando consideramos SRTs ortogonales y normales por la izquierda, no hay diferencia entre la estrategia *leftmost-outermost* y la estrategia *sc* inducida (que coindide, de acuerdo con la Proposición 10.1.7, con la estrategia *sc* restringida). Esto se debe a que ambas seleccionan exactamente las mismas ocurrencias de redexes para realizar la reducción (mientras el término no está en forma estable).

Proposición 10.1.10 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT ortogonal y normal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma, V)$ es un término no estable, entonces $\mathbb{P}_{lo}(t) = \mathbb{P}_{lo}^{\mu}(t)$.*

DEMOSTRACIÓN. Si t es un redex, es inmediato puesto que $\mathbb{P}_{lo}(t) = min_{\leq_L}(O_{\mathcal{R}}(t)) = \{\epsilon\} = min_{\leq_L}(O_{\mathcal{R}}^{\mu}(t)) = \mathbb{P}_{lo}^{\mu}(t)$. Si t no es un redex, por el Lema 2.3.6, para cualquier secuencia de reducción $A : t \xrightarrow{>\epsilon}{}^{*} \sigma(l)$, con $l \in L(\mathcal{R})$, l es única. Puesto que $\mathbb{F}_{lo}$ es estabilizante, podemos considerar que A es la secuencia de $\mathbb{F}_{lo}$-reducción de t. Sea $u = min_{\leq_L}(O_{\mathcal{R}}(t)) = \mathbb{P}_{lo}(t)$, con $u > \epsilon$. Podemos escribir $u = v.w$, donde v es la ocurrencia máxima que satisface $sprefix_t(v) = sprefix_l(v)$. Debe tenerse $v \in O_\Sigma(l)$;

de no ser así, tendríamos que $v \in O_V(l)$ y, entonces, t es un redex (de l), debido a que, siendo $\mathcal{R}$ normal por la izquierda, se tiene que para todo $v' >_L v$, $v' \in O_V(l)$. Por tanto debe cumplirse que $w = \epsilon$, $u \in O_\Sigma(l)$ y, puesto que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$, $u \in O^\mu(l)$, es decir, $u \in O^\mu(t)$. Por tanto, $u = min_{\leq_L}(O^\mu_\mathcal{R}(t)) = \mathbb{P}^\mu_{lo}(t)$. □

10.2 Extensión de los SRTs básicos

A continuación estudiamos la extensión *sc* de la clase $\overline{GR}$ de los SRTs básicos[1]. Aunque $\overline{GR}$ es una subclase de $\overline{LN}$, demostramos que $\overline{GR}$ exhibe una propiedad interesante: *cualquier* estrategia de reescritura definible para estos SRTs es una estrategia de reducción de índices. Esta propiedad motiva el análisis de la extensión *sc* de esta clase de SRTs.

10.2.1 Caracterización sintáctica

Empecemos por la caracterización sintáctica de una clase de SRTs que demostraremos que coincide con la extensión *sc* de $\overline{GR}$.

Definición 10.2.1 (SRTE *sc*-básico) *Un SRTE $\mathcal{R}$ es* sc-básico *si, para toda $l \in L(\mathcal{R})$, se verifica $O_\Sigma(l) = O^{\mu^{com}_\mathcal{R}}(l)$ (de forma equivalente, si $O_V(l) = \overline{O^{\mu^{com}_\mathcal{R}}(l)}$).*

En otras palabras, todas (y sólo) las ocurrencias no variables de las *lhs*'s de un SRTE *sc*-básico son $\mu^{com}_\mathcal{R}$-reemplazantes. De acuerdo con la Definición 10.2.1, $\mu^{com}_\mathcal{R}$ es la función de reemplazamiento que se *asume* que va a ser utilizada al realizar cómputos por reescritura sensible al contexto con estos SRTEs. El SRT del Ejemplo 3.4.4 es *sc*-básico. Observemos que este SRT no es normal por la izquierda. Por tanto, los SRTs lineales por la izquierda y *sc*-básicos ($\overline{SR}$) no están incluidos en $\overline{LN}$. Sin embargo, existen SRTs normales por la izquierda que no son *sc*-básicos.

Ejemplo 10.2.2 *El SRT normal por la izquierda*

```
f(a,x) → a
f(a,c) → b
```

no es sc-*básico.*

Por tanto, tenemos que $\overline{SR} \not\subseteq \overline{LN}$ y $\overline{LN} \not\subseteq \overline{SR}$. Cualquier SRTE *sc*-básico es $\mu^{com}_\mathcal{R}$-normal por la izquierda (puesto que $O^{\mu^{com}_\mathcal{R}}(l) = \emptyset$ para toda *lhs* l). Por tanto,

[1] Aunque se trata de una clase de SRTs muy restrictiva y de poca utilidad práctica (pues, al carecer de variables, cada regla sólo tiene asociado un redex), los SRTs básicos son interesantes desde el punto de vista teórico. La propiedad más importante que poseen es que la relación de reescritura asociada es decidible [DTHL87]. También la confluencia y la terminación son decidibles [DTHL87, Oya87, HL78].

$\overline{SR} \subseteq \mathsf{T}(\overline{LN})$. Sin embargo, observemos que el SRT $\mu_{\mathcal{R}}^{com}$-normal por la izquierda del Ejemplo 1.1.2 *no* es *sc*-básico. Por tanto, $\overline{SR} \subset \mathsf{T}(\overline{LN})$. De acuerdo con esto, los SRTs lineales por la izquierda y *sc*-básicos son fuertemente secuenciales y, como subconjunto de los SRTs lineales por la izquierda y μ-normales por la izquierda, admiten la estrategia inducida $\mathbb{F}_{lo}^{\mu_{\mathcal{R}}^{com}}$ como estrategia *sc* de reducción de índices para dichos SRTs.

Observación 10.2.3 *Todo esquema de programa recursivo (*recursive program scheme *[Cou90, Gue81, Niv75]) es* sc*-básico, puesto que las lhs's de las reglas de reescritura que constituyen dichos programas son de la forma $f(x_1, \ldots, x_k)$ para variables distintas $x_1, \ldots, x_k$.*

Los SRTs *sc*-básicos tienen mejores propiedades que los SRTs normales por la izquierda. Para ponerlas de manifiesto, primero demostramos que $\overline{SR}$ es la extensión *sc* de la clase $\overline{GR}$ de los SRTs *básicos*.

Proposición 10.2.4 *$\overline{SR} = \mathsf{T}(\overline{GR})$ y, para todo $\mathcal{R} \in \mathsf{T}(\overline{GR})$, $m(\mathcal{R}) = \{\mu_{\mathcal{R}}^{com}\}$.*

DEMOSTRACIÓN. Sea $\mathcal{R} \in \overline{SR}$ y supongamos que $\mu = \mu_{\mathcal{R}}^{com}$. Por la Definición 10.2.1, cada variable en una *lhs* $l \in L(\mathcal{R})$ es no reemplazante. Puesto que τ_μ preserva únicamente los subtérminos reemplazantes, $\tau_\mu(l)$ no tiene variables. Por tanto, $\mathcal{R}^\mu$ es equivalente por la izquierda a un SRT básico $\mathcal{S} \in \overline{GR}$. Luego $\mathcal{R} \in \mathsf{T}(\overline{GR})$ y $\mu_{\mathcal{R}}^{com} \in m(\mathcal{R})$.

Consideremos ahora que $\mathcal{R} = (\Sigma, R) \in \mathsf{T}(\overline{GR})$ y $\mu \in m(\mathcal{R})$. Recordemos que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Si $\mu \neq \mu_{\mathcal{R}}^{com}$, existe una *lhs* $l \in L(\mathcal{R})$ que tiene alguna ocurrencia reemplazante $u \in O^\mu(l)$ de una variable. Por tanto, la *lhs* $\tau_\mu(l)$ de cualquier SRT $\mathcal{S} \sim_l \mathcal{R}^\mu$ también contiene esa variable, lo que significa que $\mathcal{S} \notin \overline{GR}$ y, por consiguiente, $\mu \notin m(\mathcal{R})$. □

En el Ejemplo 3.4.4 mostramos un SRT $\mathcal{R}$ cuya $\mu_{\mathcal{R}}^{com}$-contracción $\mathcal{R}^{\mu_{\mathcal{R}}^{com}}$ se presenta en el Ejemplo 8.2.7. Observemos que $L(\mathcal{R}^{\mu_{\mathcal{R}}^{com}})$ consta únicamente de términos básicos, lo cual indica (siendo inmediato encontrar un SRT básico $\mathcal{S}$ tal que $\mathcal{S} \sim_l \mathcal{R}^{\mu_{\mathcal{R}}^{com}}$) que $\mathcal{R} \in \mathsf{T}(\overline{GR})$.

10.2.2 La estrategia inducida

A continuación demostramos la propiedad que motiva nuestro interés por la extensión de los SRTs básicos: que todo redex de un término respecto a un SRT básico está situado sobre una ocurrencia índice.

Teorema 10.2.5 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT básico y $t \in \mathcal{T}(\Sigma \cup \{\Omega\}, V)$. Si $u \in O_{\mathcal{R}}(t)$, entonces $u \in \mathcal{I}(t[\Omega]_u)$.*

DEMOSTRACIÓN. Sea s un Ω-término tal que $s|_u$ es un redex y sea $t = s[\Omega]_u$. Demostramos que $u \in \mathcal{I}(t)$ por inducción sobre la estructura de t. Si $t = \Omega$, es inmediato. Si $t = f(\widetilde{t})$ y $u = i.u'$, por la hipótesis de inducción, u' es un índice de t_i. Esto significa que $u' \in O(\omega(t_i[\bullet]_{u'}))$. Sea $t' = f(t'_1, \ldots, t'_k)$ donde $t'_i = t_i[\bullet]_{u'}$ y $t'_j = \Omega$ si $i \neq j$. Demostramos que $u \in \mathcal{I}(t')$ por contradicción. Puesto que, por la Proposición 2.5.9

$$\omega(t') = \omega(f(\omega(t'_1), \ldots, \omega(t'_i), \ldots, \omega(t'_k))) = \omega(f(\Omega, \ldots, \omega(t'_i), \ldots, \Omega))$$

y $\omega(t'_i)$ contiene $\bullet$, para eliminar $\bullet$ de t' debemos aplicar un paso de Ω-reducción sobre la ocurrencia ϵ de t', es decir, debe existir un $l \in L_\Omega(\mathcal{R})$ tal que $t' \uparrow l$. Esto implica que $t'_i \uparrow l_i$ si $l = f(\widetilde{l})$ lo cual es imposible porque siendo $\mathcal{R}$ un SRT básico, l_i no tiene ocurrencias de Ω y, puesto que $\bullet$ es un símbolo nuevo, tampoco hay ocurrencias de $\bullet$ en l_i. Luego t_i no es compatible con l_i y, por tanto, t' no es compatible con l. De acuerdo con esto, $\omega(t')|_u = \bullet$, es decir $u \in \mathcal{I}(t')$. Puesto que $t' \leq t$, por la Proposición 2.5.17, $u \in \mathcal{I}(t)$. □

La propiedad de los SRTs básicos que acabamos de demostrar permite establecer el siguiente resultado análogo para los SRTs *sc*-básicos: que todo redex reemplazante está situado sobre un índice.

Teorema 10.2.6 *Si $\mathcal{R}$ es un SRT lineal por la izquierda y* sc*-básico, para todo término t, $O_{\mathcal{R}}^{\mu_{\mathcal{R}}^{com}}(t)$ es un conjunto de redexes situados sobre índices.*

DEMOSTRACIÓN. Por la Proposición 10.2.4, existe un $\mathcal{S} \sim_l \mathcal{R}^{\mu_{\mathcal{R}}^{com}}$ tal que $\mathcal{S}$ es un SRT básico. Sea $u \in O_{\mathcal{R}}(t)$. Dado que $t|_u$ es un redex de $\mathcal{R}$, por el Teorema 5.7.9, $\tau_\mu(t|_u) = \tau_\mu(t)|_{\nu_t(u)}$ también es un redex de $\mathcal{R}^{\mu_{\mathcal{R}}^{com}}$ (o, lo que es lo mismo, de $\mathcal{S}$). Si $\nu_t(u) \in O_{\mathcal{S}}(\tau_\mu(t))$, por el Teorema 10.2.5, $\nu_t(u) \in \mathcal{J}(\tau_\mu(t[\Omega]_u))$. Por el Teorema 8.2.5, $u \in \mathcal{I}^\mu(t[\Omega]_u)$. Empleando la Proposición 2.5.17, concluimos el resultado. □

El teorema que acabamos de demostrar permite concluir que cualquier $\mu_{\mathcal{R}}^{com}$-estrategia que podamos emplear con un SRT *sc*-básico $\mathcal{R}$ sólo reduce redexes situados sobre índices.

Teorema 10.2.7 *Si $\mathcal{R}$ es un SRT lineal por la izquierda y* sc*-básico, toda $\mu_{\mathcal{R}}^{com}$-estrategia es de reducción de índices.*

Consecuentemente, toda $\mu_{\mathcal{R}}^{com}$-estrategia disfruta de las buenas propiedades esperadas.

Teorema 10.2.8 *Si $\mathcal{R}$ es un SRT cuasi ortogonal y* sc*-básico, toda $\mu_{\mathcal{R}}^{com}$-estrategia es estabilizante y $\mu_{\mathcal{R}}^{com}$-normalizante.*

DEMOSTRACIÓN. Inmediata, empleando los Teoremas 10.2.7, 7.5.2 y 7.5.7 □

También podemos establecer la siguiente consecuencia inmediata de los resultados anteriores, relacionada con la $\mu_{\mathcal{R}}^{com}$-terminación un SRT *sc*-básico $\mathcal{R}$.

Teorema 10.2.9 *Un SRT $\mathcal{R}$ cuasi ortogonal y* sc*-básico es $\mu_{\mathcal{R}}^{com}$-terminante si y sólo si $\mathcal{R}$ es $\mu_{\mathcal{R}}^{com}$-normalizante.*

DEMOSTRACIÓN. La parte *sólo si* es evidente. Para demostrar la parte *si*, consideremos un término t cualquiera. Puesto que $\mathcal{R}$ es $\mu_{\mathcal{R}}^{com}$-normalizante, dicho término tiene una forma $\mu_{\mathcal{R}}^{com}$-normal. Por el Teorema 10.2.6, toda secuencia de $\mu_{\mathcal{R}}^{com}$-reducción que arranca de t es de hecho una secuencia de reducción de índices que, por el Teorema 7.5.7, lleva a una forma $\mu_{\mathcal{R}}^{com}$-normal. Por tanto, no es posible definir secuencias de reducción infinitas y esto significa que $\mathcal{R}$ es $\mu_{\mathcal{R}}^{com}$-terminante. □

10.3 SRTs forward-branching

En la definición más general de un *index tree* (Capítulo 9), si seguimos únicamente las transiciones indicadas por la función de transferencia δ, algunos estados pueden *no* ser alcanzables a partir del estado inicial $\langle \Omega, \epsilon \rangle$. Por tanto, sólo la utilización adicional de la función de fallo ϕ permite acceder a dichos nodos del *index tree*. Strandh ha definido los *forward-branching index trees*, cuya característica distintiva es que cualquier estado del *index tree* puede ser alcanzado a partir del estado inicial mediante transiciones sucesivas ordenadas por la función de transferencia δ [Str89]. La clase *FB* de SRTs *forward-branching* es aquélla cuyos SRTs componentes admiten un *forward-branching index tree*.

En esta sección mostramos las limitaciones de nuestro método de extensión *sc* de clases de SRTs fuertemente secuenciales mediante el operador de cierre T. Demostramos que la clase *FB* no puede ser extendida mediante T.

En primer lugar, vamos a dar una caracterización alternativa de los SRTs *forward-branching* debida a Durand.

Definición 10.3.1 [Dur94] *Sea $\mathcal{R}$ un SRT y $\pi \in L_{\Omega}^{<}(\mathcal{R})$. Decimos que $P1(\pi)$ se satisface si*

$$\exists u \in O_{\Omega}(\pi), \forall \pi' \in LS_{\Omega}(\mathcal{R}), \pi < \pi' \Rightarrow u \in \overline{O}_{\Omega}(\pi')$$

Proposición 10.3.2 [Dur94] *Un SRT $\mathcal{R}$ es* forward-branching *si y sólo si $\forall \pi \in L_{\Omega}^{<}(\mathcal{R})$. $P1(\pi)$.*

El siguiente lema es auxiliar.

Lema 10.3.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT ortogonal y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Sea $\pi \in L^<_\Omega(\mathcal{R})$ y $\pi' \in LS_\Omega(\mathcal{R})$. Entonces,*

$$\pi < \pi' \Rightarrow u \in \overline{O}_\Omega(\pi')$$

si y sólo si

$$\tau_\mu(\pi) < \tau_\mu(\pi') \Rightarrow \nu_{\pi'}(u) \in \overline{O}_{\Omega_\mu}(\tau_\mu(\pi')).$$

DEMOSTRACIÓN. Puesto que $\pi \in \mathcal{T}(\Sigma \cup \{\Omega\})$ y siendo $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, por la Proposición 4.1.5 se tiene $\mu_\pi \sqsubseteq \mu$ y $\mu_{\pi'} \sqsubseteq \mu$ e incluso $\mu_\pi \sqsubseteq \mu_{\pi'} \sqsubseteq \mu$. Por las Proposiciones 5.7.4 y 5.7.5, $\pi < \pi' \Leftrightarrow \tau_\mu(\pi) < \tau_\mu(\pi')$. Dado que $\mu_{\pi'} \sqsubseteq \mu$, tenemos $\overline{O}_\Omega(\pi') = \overline{O}^\mu_\Omega(\pi')$. Por tanto, $u \in \overline{O}_\Omega(\pi') \Leftrightarrow u \in \overline{O}^\mu_\Omega(\pi') \Leftrightarrow \nu_{\pi'}(u) \in \overline{O}_{\Omega_\mu}(\tau_\mu(\pi'))$. □

Proposición 10.3.4 *Si $\mathcal{R} = (\Sigma, R)$ es un SRT y $\mu \in M_\Sigma$ es tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, entonces $LS_\Omega(\mathcal{R}) \simeq_{\tau_\mu} LS_{\Omega_\mu}(\mathcal{R}^\mu)$.*

DEMOSTRACIÓN. Consideremos $\pi \in LS_\Omega(\mathcal{R})$. Existen $l \in L_\Omega(\mathcal{R})$ y $u \in \overline{O}_\Omega(l)$ tales que $l|_u = \pi$. Puesto que $\mu_l \sqsubseteq \mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, tenemos que $u \in O^\mu(l)$. Por tanto, por la Proposición 5.2.8, $\nu_t(u) \in \overline{O}_{\Omega_\mu}(\tau_\mu(l))$ y $\tau_\mu(l|_u) = \tau_\mu(l)|_{\nu_l(u)} = \tau_\mu(\pi)$, es decir, $\tau_\mu(\pi) \in LS_{\Omega_\mu}(\mathcal{R}^\mu)$.

Para demostrar la sobreyectividad, consideremos $\varpi \in LS_{\Omega_\mu}(\mathcal{R}^\mu)$. Entonces, existe un $l \in L_\Omega(\mathcal{R})$ tal que $\varpi = \tau_\mu(l)|_v$ para alguna $v \in O(\tau_\mu(l))$ y $v \notin O_{\Omega_\mu}(\tau_\mu(l))$. Por tanto, por la Proposición 5.2.8, si $v = \nu_l(u)$, $\varpi = \tau_\mu(l)|_v = \tau_\mu(l|_u) = \tau_\mu(\pi)$ y $\pi \in LS_\Omega(\mathcal{R})$ (debido a que $u \notin O_\Omega(l)$).

Para demostrar la inyectividad, sean $\pi, \pi' \in LS_\Omega(\mathcal{R})$. Puesto que $\forall l \in L_\Omega(\mathcal{R})$, $l \in \mathcal{T}(\Sigma \cup \{\Omega\})$, tenemos que $\pi, \pi' \in \mathcal{T}(\Sigma \cup \{\Omega\})$. Por la Proposición 4.1.5, $\mu_\pi \sqcup \mu_{\pi'} \sqsubseteq \mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Por tanto, por el Corolario 5.7.11, $\tau_\mu(\pi) = \tau_\mu(\pi') \Rightarrow \pi = \pi'$. □

Finalmente podemos establecer el resultado que anunciábamos: que la aplicación del operador de extensión *sc* a la clase FB deja a ésta inalterada.

Teorema 10.3.5 $\mathsf{T}(FB) = FB$.

DEMOSTRACIÓN. Puesto que T es extensiva, tenemos que $FB \subseteq \mathsf{T}(FB)$. Demostramos que $\mathsf{T}(FB) \subseteq FB$. Si $\mathcal{R} \in \mathsf{T}(FB)$, entonces existe $\mu \sqsupseteq \mu^{com}_{\mathcal{R}}$ tal que $\mathcal{R}^\mu \sim_l \mathcal{S} \in FB$. Puesto que $L(\mathcal{R}^\mu) = L(\mathcal{S})$ y no necesitaremos considerar las rhs's en lo que queda de demostración, podemos considerar $L(\mathcal{R}^\mu)$. Por la Proposición 10.3.2, hay que demostrar que $\forall \pi \in L^<_\Omega(\mathcal{R}).P1(\pi)$. Puesto que $\mathcal{S} \in FB$, tenemos que $\forall \varpi \in L^<_{\Omega_\mu}(\mathcal{R}^\mu).P1_\mu(\varpi)$ (escribimos $P1_\mu$ para denotar la propiedad $P1$ referida al SRT $\mathcal{R}^\mu$). Debemos demostrar que $\forall \varpi \in L^<_{\Omega_\mu}(\mathcal{R}^\mu).P1_\mu(\varpi) \Rightarrow \forall \pi \in L^<_\Omega(\mathcal{R}).P1(\pi)$. Por la Proposición 9.3.3, τ_μ es una biyección entre los conjuntos $L^<_\Omega(\mathcal{R})$ y $L^<_\Omega(\mathcal{R}^\mu)$. Por tanto, es suficiente demostrar que para un π arbitrario, $P1_\mu(\tau_\mu(\pi)) \Rightarrow P1(\pi)$.

Esto significa demostrar que, dado $v \in O_{\Omega_\mu}(\tau_\mu(\pi))$, si se cumple que $\forall \tau_\mu(\pi') \in LS_{\Omega_\mu}(\mathcal{R}^\mu), \tau_\mu(\pi) < \tau_\mu(\pi') \Rightarrow v \in \overline{O}_{\Omega_\mu}(\tau_\mu(\pi'))$, entonces, siendo $v = \nu_\pi(u)$, tenemos que $\forall \pi' \in LS_\Omega(\mathcal{R}), \pi < \pi' \Rightarrow u \in \overline{O}_\Omega(\pi')$. Una vez más, empleando la Proposición 10.3.4, sólo necesitamos demostrar que, dado un $\pi' \in LS_\Omega(\mathcal{R})$ arbitrario, $\tau_\mu(\pi) < \tau_\mu(\pi') \Rightarrow v \in \overline{O}_{\Omega_\mu}(\tau_\mu(\pi'))$ implica $\pi < \pi' \Rightarrow u \in \overline{O}_\Omega(\pi')$. Esto es inmediato, de acuerdo con el Lema 10.3.3. □

Otro resultado de interés relaciona la extensión *sc* de los SRTs ortogonales y normales por la izquierda con los SRTs *forward-branching*.

Proposición 10.3.6 $\mathsf{T}(LN) \subseteq FB$.

DEMOSTRACIÓN. Dado que $LN \subseteq FB$, por monotonía de T, $\mathsf{T}(LN) \subseteq \mathsf{T}(FB)$. Por el Teorema 10.3.5, $\mathsf{T}(LN) \subseteq \mathsf{T}(FB) = FB$. □

Podemos establecer mediante un ejemplo que, de hecho, $\mathsf{T}(LN) \subset FB$.

Ejemplo 10.3.7 *Consideremos el SRT* forward-branching $\mathcal{R}$ *[Dur94]:*

```
f(g(x,a),a) → x
f(g(a,x),b) → g(x,x)
g(b,b) → b
```

Este SRT no es μ-normal por la izquierda para ningún μ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, puesto que $\mu^{com}_{\mathcal{R}} = \mu_\mathsf{T}$ y $\mathcal{R}$ no es (μ_T-)normal por la izquierda.

10.4 Trabajos relacionados

El Teorema 10.2.6 muestra que es posible realizar reducción de índices en paralelo. Esto matiza la afirmación de Kennaway [Ken94]: "*in an orthogonal TRS, no term can have more than one strongly root-needed redex*". Kennaway arguye que esta afirmación justifica la existencia de un 'conflicto' entre la realización de reducciones necesarias y la posibilidad de llevar a cabo éstas en paralelo. La noción de SRT *sc*-básico demuestra que este conflicto sólo surge en el trabajo realizado por Kennaway debido a su elección de una aproximación computable a los redexes necesarios para estabilización (*strongly root-needed redexes* [Ken94]).

Por ejemplo, si consideramos el SRT $\mathcal{R}$ del Ejemplo 3.4.4, el término $t = \mathtt{sel}(\mathtt{0} + \mathtt{0}, \mathtt{from}(\mathtt{0}))$ no es estable y tenemos que $O^{\mu^{com}_{\mathcal{R}}}_{\mathcal{R}}(t) = \{1, 2\}$. Por el Teorema 10.2.6, 1 y 2 son ocurrencias de redexes necesarios para su estabilización. Por tanto, ambos pueden ser reducidos en paralelo contribuyendo a la estabilización del término.

Respecto a la estrategia E empleada en OBJ y CafeOBJ, cuya formalización por reescritura puede encontrarse en [OF97], los resultados establecidos en esta sección

pueden utilizarse para asegurar que, en la terminología de [OF97], la estrategia E siempre obtendrá la forma E-normal de un término cuando ésta exista. Esto puede asegurarse para todo programa cuasi ortogonal $\mathcal{R}$ tal que, para cada símbolo k-ario f su estrategia local asociada $(i_1, \ldots, i_n)$ (sea explícita o implícitamente) verifique las condiciones siguientes:

1. $n \leq k + 2$
2. para todo $p < n - 1$, se verifica $i_p < i_{p+1}$
3. $i_1 = 0$ e $i_n = 0$.

Además, si obtenemos a partir de estas estrategias la función de reemplazamiento μ asociada a éstas (procediendo tal como se indicó en la Sección 3.5) debemos asegurar que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ y que $\mathcal{R}$ es μ-normal por la izquierda. En ese caso, la estrategia E coincide con $\mathbb{F}^{\mu}_{lo}$ y el Teorema 10.1.9 asegura que la estrategia E es ENF-normalizante. Algo similar cabe decir si el SRT es sc-básico, cuando las estrategias locales correspondientes consideran todos (y sólo) los índices presentes en $\mu^{com}_{\mathcal{R}}$ (más el 0).

Puesto que estos resultados son aplicables también si $\mathcal{R}$ no es μ-terminante, constituyen una nueva aportación al análisis de la estrategia E que complementa la metodología expuesta en la Sección 5.6, basada en la demostración de μ-terminación del SRT.

Parte III

Formas canónicas y reescritura sensible al contexto

Capítulo 11

Valores y formas estables constructoras

El cálculo del valor asociado a un término es una tarea de gran interés. De hecho, en programación funcional [FH88, PE93, Rea93], resulta más importante que el puro cómputo de formas normales puesto que, en la descripción semántica usual de los lenguajes funcionales, son los términos constituidos por símbolos constructores (es decir, los *valores básicos*) los que adquieren contenido semántico preciso como representantes de objetos del dominio denotado por una determinada definición de tipo de dato algebraico. Por este motivo, es interesante analizar en profundidad la posibilidad de *evaluar*[1] términos de forma eficiente empleando la *rsc*.

En la Sección 11.1 establecemos las condiciones para la obtención de formas estables constructoras y valores mediante la *rsc*.

En la Sección 11.2 se discute el problema que supone el hecho de que, para obtener el valor asociado a una expresión inicial mediante la *rsc*, haya que relajar las restricciones de reemplazamiento respecto a las necesarias para obtener una forma estable constructora. Suponiendo esto un riesgo de realización de cómputos innecesarios o quizá la generación de secuencias de reducción infinitas, se propone resolver esta discrepancia mediante una *transformación de programas*.

La transformación se define a partir de un análisis aproximado de los símbolos definidos externos que pueden surgir en una derivación de reescritura. Las ideas y nociones que preparan dicho análisis se presentan en la Sección 11.3.

En la Sección 11.4 se describe el cálculo de la aproximación a los símbolos definidos externos. La corrección de la aproximación obtenida mediante dicho cálculo se demuestra en la Sección 11.5.

[1] En este capítulo, la palabra *evaluación* adquiere el sentido preciso de obtención de un valor, es decir, de un término constructor $\delta \in \mathcal{T}(\mathcal{C}, V)$, mediante un SRT $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$.

En la Sección 11.6 se presenta la transformación, cuya corrección y completitud se demuestra en la Sección 11.7.

11.1 Obtención directa de valores y formas estables constructoras

Al realizar cómputos por reescritura, los programas canónicos (terminantes y confluentes) son los más interesantes desde el punto de vista computacional: las propiedades de terminación y confluencia garantizan un comportamiento operacional consistente con la semántica de los programas. Debido a la terminación, toda secuencia de reducción iniciada en una expresión se detendrá en algún momento. Debido a la confluencia, cualquier estrategia escogida para efectuar el cálculo obtendrá el mismo resultado final, sea cual sea.

Podría pensarse que la μ-canonicidad, es decir, la conjunción de las propiedades de μ-terminación y μ-confluencia para un SRT y una función de reemplazamiento μ dada, desempeñarán un papel análogo en la ejecución de programas mediante la *rsc* garantizando la corrección y completitud de los cómputos. Sin embargo, un ejemplo sencillo muestra que esta aparente bondad de las propiedades operacionales de la *rsc* no resulta suficientemente potente desde el punto de vista semántico.

Ejemplo 11.1.1 *Consideremos el SRT canónico $\mathcal{R}$:*

$$\texttt{f(x,y)} \rightarrow \texttt{g(x,y)} \qquad \texttt{a} \rightarrow \texttt{b}$$
$$\texttt{g(b,b)} \rightarrow \texttt{b}$$

Si la función de reemplazamiento μ es tal que $\mu(\texttt{f}) = \mu(\texttt{g}) = \{1\}$, $\mathcal{R}$ tiene variables μ-reemplazantes homogéneas por la izquierda y, dado que es lineal por la izquierda y no tiene pares μ-críticos (Definición 6.1.5), es μ-ortogonal (Definición 6.3.3). De acuerdo con el Teorema 6.4.4 es μ-confluente. Más aún, por el Teorema 5.1.4 también es μ-terminante. Se trata de un SRT μ-canónico. Sin embargo, tenemos lo siguiente:

1. $\underline{\texttt{f(b,a)}} \hookrightarrow \texttt{g(b,a)}$ *y* $\texttt{g(b,a)}$ *es una forma μ-normal, pero* no *es la forma normal de* $\texttt{f(b,a)}$ *puesto que*

2. $\underline{\texttt{f(b,a)}} \rightarrow \texttt{g(b,}\underline{\texttt{a}}\texttt{)} \rightarrow \underline{\texttt{g(b,b)}} \rightarrow \texttt{b}$.

En otras palabras, a pesar de que tanto $\hookrightarrow$ como $\rightarrow$ son terminantes y confluentes, las formas normales respectivas computadas para el mismo término inicial son distintas. En particular, en este caso, la forma normal obtenida es un valor. Por tanto, la μ-canonicidad no *constituye una garantía de plena capacidad computacional de la* rsc *desde un punto de vista semántico.*

Al considerar la descomposición estándar $\Sigma = \mathcal{C} \uplus \mathcal{F}$ de una signatura Σ en presencia de un SRTE en símbolos constructores $\mathcal{C}$ y símbolos definidos $\mathcal{F}$ (ver la Sección 2.3.1), es posible formular algunos resultados de interés sobre la completitud de los cómputos de evaluación mediante la *rsc*. Podemos resolver el problema apuntado en el Ejemplo 11.1.1 si añadimos el requerimiento (que se demuestra así de carácter eminentemente semántico) de que la función de reemplazamiento μ empleada en los cómputos iguale o exceda la función de reemplazamiento canónica. El primer resultado que obtenemos en este sentido se refiere al cómputo de formas estables constructoras mediante la *rsc*.

Teorema 11.1.2 (Cómputo de formas estables constructoras) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE lineal por la izquierda y $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma, V)$, $x \in V$ y $s = c(\tilde{s})$ para algún $c \in \mathcal{C}$. Si $t \to^* x$, entonces $t \hookrightarrow^* x$. Si $t \to^* s$, entonces existe un término $s' = c(\widetilde{s'})$ tal que $t \hookrightarrow^* s'$ y $s' \overset{>\epsilon}{\longrightarrow}{}^* s$.*

DEMOSTRACIÓN. Consecuencia inmediata del Teorema 4.2.8. □

El siguiente teorema establece las condiciones que aseguran que la *rsc* es suficientemente potente para llevar evaluar un término completamente. Para ello debemos relajar las restricciones de reemplazamiento impuestas por la función de reemplazamiento canónica $\mu_\mathcal{R}^{com}$.

Definición 11.1.3 (Función de reemplazamiento de evaluación) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $\mathcal{B} \subseteq \mathcal{C}$. La función de reemplazamiento de $\mathcal{B}$-evaluación, $\mu_\mathcal{R}^\mathcal{B}$, se define como sigue:*

$$\mu_\mathcal{R}^\mathcal{B} = \mu_\mathcal{R}^{com} \sqcup \mu_\mathcal{B}$$

donde $\mu_\mathcal{B}(c) = \mathbb{N}^+_{ar(c)}$ para todo $c \in \mathcal{B}$ y $\mu_\mathcal{B}(f) = \emptyset$ si $f \notin \mathcal{B}$.

Si restringimos nuestra atención al cómputo del *valor* asociado a un término, la *rsc* es capaz de obtenerlo sin necesidad de incluir recursos operacionales extra. Esta afirmación adquiere su expresión formal en el siguiente teorema.

Teorema 11.1.4 (Cómputo de valores) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE lineal por la izquierda, $\mathcal{B} \subseteq \mathcal{C}$ y $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^\mathcal{B} \sqsubseteq \mu$. Para todo $t \in \mathcal{T}(\Sigma, V)$ y $\delta \in \mathcal{T}(\mathcal{B}, V)$, tenemos que $t \to^* \delta$ si y sólo si $t \hookrightarrow^* \delta$.*

DEMOSTRACIÓN. La parte *si* de la doble implicación es inmediata, puesto que $\hookrightarrow \subseteq \to$. Para demostrar la parte *sólo si*, procedemos por inducción sobre la estructura de δ.

Si $\delta \in \mathcal{B} \cup V$, entonces, dado que $\mu_\mathcal{R}^{com} \sqsubseteq \mu_\mathcal{R}^\mathcal{B} \sqsubseteq \mu$, por el Teorema 11.1.2, existe un s' tal que $t \hookrightarrow^* s'$ y $s' \overset{>\epsilon}{\longrightarrow}{}^* \delta$. Puesto que $\delta \in \mathcal{B} \cup V$, $s' = \delta$.

Si $\delta = c(\tilde{\delta})$, con $c \in \mathcal{B}$, descomponemos $t \to^* \delta$ como sigue: $t \to^* c(\tilde{t}) \to^* \delta$. Dado que $\mu_\mathcal{R}^{com} \sqsubseteq \mu_\mathcal{R}^\mathcal{B} \sqsubseteq \mu$, el Teorema 11.1.2 asegura la existencia de un término $c(\tilde{s})$ tal

que $t \hookrightarrow^* c(\tilde{s})$ y $c(\tilde{s}) \overset{>\epsilon}{\longrightarrow}{}^* c(\tilde{t})$. Por tanto, $c(\tilde{s}) \overset{>\epsilon}{\longrightarrow}{}^* c(\tilde{\delta})$ y, consecuentemente, $s_i \rightarrow^* \delta_i$ para todo $1 \leq i \leq ar(c)$. Por la hipótesis de inducción, $s_i \hookrightarrow^* \delta_i$. Puesto que $\mu^{\mathcal{B}}_{\mathcal{R}} \sqsubseteq \mu$, tenemos que $i \in \mu(c)$ para todo i, $1 \leq i \leq ar(c)$. De esta manera, empleando la Proposición 3.3.6, completamos la demostración. □

Para poder utilizar el Teorema 11.1.4 en la práctica, debemos conocer $\mathcal{B}$ antes de iniciar la μ-evaluación, de forma que podamos establecer $\mu^{\mathcal{B}}_{\mathcal{R}}$ y, a partir suyo, la función de reemplazamiento μ empleada, de forma que verifique $\mu^{\mathcal{B}}_{\mathcal{R}} \sqsubseteq \mu$. Esto no resulta tan difícil si disponemos de alguna información sobre los *tipos* asociados a los símbolos de la signatura.

Observación 11.1.5 *Si trabajamos con SRTEs con tipos algebraicos (como es habitual en implementaciones basadas en reescritura de lenguajes funcionales [PE93]) el símbolo de función* `sel` *definido mediante el programa del Ejemplo 3.4.3, adquiere el tipo:*

$$\texttt{sel} : \mathsf{Nat} \times [\mathsf{Nat}] \rightarrow \mathsf{Nat}$$

donde Nat *y* [Nat] *denotan los tipos número natural y lista de naturales respectivamente. En ese caso podemos asegurar que cualquier término bien tipificado* $t =$ `sel(`*Num*`,`*ListNum*`)`*, donde* Num *y* ListNum *son términos de tipo* Nat *y* [Nat] *respectivamente, debe proporcionar (en su caso) un valor de tipo* Nat*. Dado que el tipo de una función establece (en los sistemas de tipificación algebraicos) qué símbolos constructores intervienen en la constitución de valores de este tipo, esto nos permitirá establecer el conjunto* $\mathcal{B}$*. Por ejemplo,* `0` *y* `s` *son los constructores de datos del tipo* Nat*.*

El Teorema 11.1.4 proporciona la cobertura formal necesaria para generalizar el procedimiento de evaluación del término `sel(s(s(0)),from(0))` del Ejemplo 3.4.3. La función de reemplazamiento μ empleada en dicho ejemplo satisface la condición $\mu^{\mathcal{B}}_{\mathcal{R}} \sqsubseteq \mu$, sin más que tomar $\mathcal{B} = \{\texttt{0}, \texttt{s}\}$. Tal como se demostró en el Ejemplo 5.5.3, el SRT $\mathcal{R}$ es μ-terminante. Esto significa que podemos evaluar todas las llamadas a función de la forma `sel(`*Num*`,`*ListNum*`)` sin riesgo de iniciar secuencias de reducción infinitas y con la seguridad de que el valor asociado al término inicial (cuando exista) va a ser obtenido por la *rsc*.

Observación 11.1.6 *Merece la pena observar que el SRT confluente* $\mathcal{R}$ *del Ejemplo 3.4.3* no *tiene variables* μ*-reemplazantes homogéneas por la izquierda, debido a la regla de reescritura*

```
from(x) → x::from(s(x))
```

que tiene una ocurrencia reemplazante de la variable `x` *en la parte izquierda que no aparece como reemplazante en una de sus ocurrencias (la* 2.1.1*) en la parte derecha.*

Por tanto, no podemos asegurar la μ-confluencia. Sin embargo, el Teorema 11.1.4 nos asegura que el valor (unico) asociable a cada expresión inicial del tipo Nat *se puede computar mediante la* rsc.

Por tanto, *no* necesitamos imponer que los SRTs sean μ-confluentes para garantizar la completitud y corrección de las μ-evaluaciones. La confluencia del SRT es suficiente para asegurar que todas las μ-derivaciones que arranquen de un término t pueden hacerse converger a su valor $\delta \in \mathcal{T}(\mathcal{B}, V)$ (si dicho valor existe y $\mu^{\mathcal{B}}_{\mathcal{R}} \sqsubseteq \mu$). Esto es fácil de establecer: si t se evalúa a δ, y tenemos que $t \hookrightarrow^* t'$ y $t \hookrightarrow^* t''$, la confluencia del SRT asegura que el valor, tanto de t' como de t'', es δ. Por tanto, por el Teorema 11.1.4 $t' \hookrightarrow^* \delta$ y $t'' \hookrightarrow^* \delta$ también.

A su vez, esto sugiere que tener variables μ-reemplazantes homogéneas por la izquierda *no es una condición necesaria* para asegurar la μ-confluencia de un SRT $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$. De hecho, si consideramos SRTs canónicos y completamente definidos (para los cuales todo término tiene una forma normal única que además es un valor), el Teorema 11.1.4 implica que la confluencia del SRT asegura la μ-confluencia del mismo cuando μ satisface $\mu^{\mathcal{C}}_{\mathcal{R}} \sqsubseteq \mu$. Por ejemplo:

Ejemplo 11.1.7 *Consideremos el SRT terminante y completamente definido $\mathcal{R}$:*

```
f(x)      → g(x,x)      h(0)    → 0
g(0,x)    → 0           h(s(x)) → 0
g(s(x),y) → s(x)
```

Escogiendo $\mu(\mathtt{f}) = \mu(\mathtt{g}) = \mu(\mathtt{h}) = \mu(\mathtt{s}) = \{1\}$, $\mathcal{R}$ no tiene variables μ-reemplazantes homogéneas por la izquierda (debido a la regla `f(x) → g(x,x)`*). Sin embargo, siendo $\mathcal{C} = \{\mathtt{0}, \mathtt{s}\}$, tenemos que $\mu^{\mathcal{C}}_{\mathcal{R}} \sqsubseteq \mu$ y, por tanto, el Teorema 11.1.4 implica la μ-confluencia de $\mathcal{R}$.*

11.1.1 *Strictness* y completitud de los cómputos con *rsc*

Con relación al argumento esgrimido en diversos trabajos [EP87, HW87, KW95, Mar90] a favor de la especificación de anotaciones sintácticas asociadas a los símbolos de función basadas en información de *strictness*, es sencillo ver que, en el caso de la *rsc*, esta práctica no es recomendable a priori, pues no asegura la completitud de los cómputos ni siquiera a una forma estable.

Ejemplo 11.1.8 *Tal como se mencionó en el Ejemplo 4.2.5, si establecemos la función de reemplazamiento μ de forma que emplee exclusivamente la información de* strictness *obtenida para el SRT $\mathcal{R}$ del Ejemplo 4.2.2 (haciendo $\mu = \mu^{str}_{\mathcal{R}}$), no se verifica que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Por tanto, ni siquiera puede asegurarse que las μ-derivaciones obtienen formas estables, si existen. De hecho, el término* `first(s(0),from(0))` *no*

puede reducirse a la forma estable `0::first(0,from(s(0)))` *porque, debido a que* $\mu(\texttt{first}) = \{1\}$*, el término* `first(s(0),from(0))` *ya es una forma* μ*-normal.*

La insuficiencia de la información de *strictness* para asegurar la completitud de los cómputos mediante la *rsc* (por contraposición al uso habitual de este tipo de información en otros sistemas de cómputo), se debe a la interpretación operacional de las anotaciones sintácticas. Tal como se indicó en la introducción, con la *rsc* la presencia explícita de anotaciones (recogidas por la función de reemplazamiento) constituyen la *única* forma de activar el cómputo de los argumentos (subtérminos) de una llamada a función (o expresión). Sin embargo, en programación funcional, por ejemplo, las anotaciones son auxiliares. Asumen un mecanismo de cómputo preexistente (normalmente una estrategia perezosa) que garantiza la completitud del cálculo.

11.2 Planteamiento de una transformación de programas para la evaluación eficiente de expresiones

El hecho de que, de acuerdo con el Teorema 11.1.2, seamos capaces de obtener una forma estable constructora empleando la función de reemplazamiento $\mu^{com}_{\mathcal{R}}$, contrasta con la necesidad, de acuerdo con el Teorema 11.1.4, de utilizar $\mu^{\mathcal{B}}_{\mathcal{R}}$ para la evaluación completa. Puesto que, en general, ésta es menos restrictiva que aquélla, es decir $\mu^{com}_{\mathcal{R}} \sqsubset \mu^{\mathcal{B}}_{\mathcal{R}}$, el empleo de $\mu^{\mathcal{B}}_{\mathcal{R}}$ supone un riesgo de realización de reducciones inútiles o, lo que es peor, de iniciar secuencias de reducción infinitas. Para ilustrar esta afirmación, veamos algunos siguientes ejemplos.

Ejemplo 11.2.1 *Consideremos el SRT* $\mathcal{R}$*:*

```
dbl(0)          → 0             dbls([])      → []
dbl(s(x))       → s(s(dbl(x)))  dbls(x::y)    → dbl(x)::dbls(y)
sel(0,x::y)     → x             indx([],x)    → []
sel(s(x),y::z)  → sel(x,z)      indx(x::y,z)  → sel(x,z)::indx(y,z)
from(x)         → x::from(s(x))
```

La función de reemplazamiento canónica para $\mathcal{R}$ *es:*

$$\mu^{com}_{\mathcal{R}}(\texttt{s}) = \mu^{com}_{\mathcal{R}}(\texttt{::}) = \mu^{com}_{\mathcal{R}}(\texttt{from}) = \emptyset,$$

$$\mu^{com}_{\mathcal{R}}(\texttt{dbl}) = \mu^{com}_{\mathcal{R}}(\texttt{dbls}) = \mu^{com}_{\mathcal{R}}(\texttt{indx}) = \{1\} \text{ y } \mu^{com}_{\mathcal{R}}(\texttt{sel}) = \{1,2\}$$

Podemos evaluar el término `sel(s(0),dbls(from(0)))` *hasta una forma estable constructora empleando* $\mu_{\mathcal{R}}^{com}$*:*

$$
\begin{array}{ll}
\texttt{sel(s(0),dbls(}\underline{\texttt{from(0)}}\texttt{))} & \\
\quad \hookrightarrow \texttt{sel(s(0),}\underline{\texttt{dbls(0::from(s(0)))}}\texttt{)} & (*) \\
\quad \hookrightarrow \underline{\texttt{sel(s(0),dbl(0)::dbls(from(s(0))))}} & (**) \\
\quad \hookrightarrow \texttt{sel(0,dbls(}\underline{\texttt{from(s(0))}}\texttt{))} & \\
\quad \hookrightarrow \texttt{sel(0,}\underline{\texttt{dbls(s(0)::from(s(s(0))))}}\texttt{)} & (*) \\
\quad \hookrightarrow \underline{\texttt{sel(0,dbl(s(0))::dbls(from(s(s(0))))}} & \\
\quad \hookrightarrow \underline{\texttt{dbl(s(0))}} & \\
\quad \hookrightarrow \texttt{s(s(dbl(0)))} &
\end{array}
$$

La derivación no puede continuar puesto que `s(s(dbl(0)))` *es una forma* $\mu_{\mathcal{R}}^{com}$*-normal.*

Si consideramos ahora $\mu_{\mathcal{R}}^{com}$ tal como se definió en el Ejemplo 11.2.1 y $\mathcal{B} = \{\texttt{0}, \texttt{s}\}$ (dado que el tipo de la expresión de entrada es `Nat` cuyos constructores son `0` y `s`), entonces $\mu_{\mathcal{R}}^{\mathcal{B}}(\texttt{s}) = \{1\}$ (pero recordemos que $\mu_{\mathcal{R}}^{com}(\texttt{s}) = \emptyset$). Ahora podemos completar la derivación interrumpida en el Ejemplo 11.2.1,

$$\texttt{s(s(}\underline{\texttt{dbl(0)}}\texttt{))} \hookrightarrow \texttt{s(s(0))}$$

hasta obtener el valor final `s(s(0))`.

Al considerar el primer tramo de la derivación como una $\mu_{\mathcal{R}}^{\mathcal{B}}$-derivación en lugar de como una $\mu_{\mathcal{R}}^{com}$-derivación, aún podemos evitar las reducciones no terminantes o inútiles que podrían iniciarse en los pasos marcados con $(*)$ y $(**)$ en el Ejemplo 11.2.1, respectivamente. Esto se debe a que $\mu_{\mathcal{R}}^{\mathcal{B}}(\texttt{::}) = \emptyset$. Más aún, podemos demostrar que $\mathcal{R}$ es $\mu_{\mathcal{R}}^{\mathcal{B}}$-terminante (empleando el método de Zantema, ver la Sección 5.5). No obstante, pueden darse situaciones más comprometidas.

Ejemplo 11.2.2 *Consideremos el SRT* $\mathcal{R}$ *del Ejemplo 11.2.1 y la expresión inicial* $t =$ `indx(s(0)::[],dbls(from(0)))`*. Podemos evaluar* t *hasta una forma estable constructora empleando* $\mu_{\mathcal{R}}^{com}$*:*

$$
\begin{array}{l}
\underline{\texttt{indx(s(0)::[],dbls(from(0)))}} \\
\quad \hookrightarrow \texttt{sel(s(0),dbls(from(0)))::indx([],dbls(from(0)))}
\end{array}
$$

Si deseamos obtener el valor de t*, una lista de números naturales de acuerdo con el perfil*

$$\texttt{indx} : [\mathsf{Nat}] \times [\mathsf{Nat}] \rightarrow [\mathsf{Nat}],$$

definiremos $\mathcal{B} = \{\mathtt{0}, \mathtt{s}, \mathtt{[]}, \mathtt{::}\}$. *Por tanto,* $\mu^{\mathcal{B}}_{\mathcal{R}}(\mathtt{s}) = \{1\}$ *y* $\mu^{\mathcal{B}}_{\mathcal{R}}(\mathtt{::}) = \{1,2\}$. *Esto nos permite continuar hasta el final:*

```
sel(s(0),dbls(from(0)))::indx([],dbls(from(0)))
    ↪ sel(s(0),dbls(from(0)))::[]
    ↪ sel(s(0),dbls(0::from(s(0))))::[]                (*)
    ↪ sel(s(0),dbl(0)::dbls(from(s(0))))::[]           (**)
    ↪ sel(0,dbls(from(s(0))))::[]
    ↪ sel(0,dbls(s(0)::from(s(s(0)))))::[]             (*)
    ↪ sel(0,dbl(s(0))::dbls(from(s(s(0)))))::[]
    ↪ dbl(s(0))::[]
    ↪ s(s(dbl(0)))::[]
    ↪ s(s(0))::[]
```

Sin embargo, $\mathcal{R}$ no *es* $\mu^{\mathcal{B}}_{\mathcal{R}}$*-terminante ahora: puesto que* $\mu^{\mathcal{B}}_{\mathcal{R}}(\mathtt{::}) = \{1,2\}$, *es posible aplicar a los términos de los pasos marcados con* $(*)$ *la regla de reescritura para la función* `from` *de forma reiterada. Ello hace posible definir una secuencia de reducción infinita que con* $\mu^{com}_{\mathcal{R}}$ *no podía darse. Además también se facilita la realización de pasos de reducción innecesarios al quedar abierta la posibilidad de reducir el redex* `double(0)` *en el paso marcado con* $(**)$. *Claramente, actuar sobre dicho redex no contribuye a la evaluación de la expresión inicial.*

Para evitar situaciones como las indicadas en el Ejemplo 11.2.2, vamos a introducir una transformación de programas. Dicha transformación se inspira en una observación muy simple: el proceso de evaluación de un término t puede descomponerse en dos fases:

1. Obtener una forma estable constructora $s = c(\tilde{s})$ de t para algún $c \in \mathcal{C}$.

2. Evaluar (recursivamente) los subtérminos inmediatos $s_1, \ldots, s_{ar(c)}$ de s.

Este esquema de evaluación puede implementarse directamente mediante la *rsc* sin emplear control adicional. Para hacerlo eficientemente, necesitamos manejar de forma separada las restricciones que son necesarias para obtener una forma estable constructora a partir de la expresión inicial (representadas por $\mu^{com}_{\mathcal{R}}$) y las que son necesarias para propagar dicha evaluación incompleta sobre los subtérminos no evaluados que se encuentran en el seno de un contexto constituido únicamente por constructores. Para formalizar dicha diferenciación, introducimos el siguiente concepto.

Definición 11.2.3 (Contexto constructor máximo de un término)
Sea $\Sigma = \mathcal{C} \uplus \mathcal{F}$ *una signatura y* $t \in \mathcal{T}(\Sigma, V)$. *El* contexto constructor máximo $CCM(t)$

de t es:

$$CCM(t) = \begin{cases} x & si\ t = x \in V \\ c(CCM(t_1), \ldots, CCM(t_k)) & si\ t = c(t_1, \ldots, t_k)\ y\ c \in \mathcal{C} \\ \Box & si\ root(t) \in \mathcal{F} \end{cases}$$

A veces escribiremos CCM para abreviar la expresión contexto constructor máximo.

Definición 11.2.4 (Símbolo definido externo) *Sea $\Sigma = \mathcal{C} \uplus \mathcal{F}$ una signatura y $t \in \mathcal{T}(\Sigma, V)$. Si $C[\,] = CCM(t)$ y $t = C[t_1, \ldots, t_n]$, a los símbolos $root(t_i) = f_i \in \mathcal{F}$ se les denomina los* símbolos definidos externos *de t.*

Por debajo de los símbolos definidos externos de un término $t = f(\tilde{t})$, las tareas de reducción de los subtérminos t_i, para $1 \leq i \leq n$, deben avanzar sólo hasta obtener una forma estable constructora de cada uno de ellos. Por tanto, sólo necesitamos emplear $\mu_{\mathcal{R}}^{com}$ para reducirlos suficientemente.

11.3 Diferenciando los símbolos internos y externos

En programación funcional, toda expresión tiene un tipo [FH88, PE93, Rea93]. Los tipos monomórficos corresponden a dominios de valores. Si trabajamos con tipos de datos algebraicos, podemos identificar los términos constructores básicos (esto es, sin variables) que pueden definirse empleando los símbolos constructores de la correspondiente definición de tipo, con los valores del dominio que corresponde al tipo monomórfico en cuestión. A partir de ahora, en este capítulo, asumiremos la existencia de un sistema de tipos de datos algebraico. La forma general de una declaración de un tipo algebraico monomórfico es [FH88]:

$$\texttt{data}\ \tau\ =\ c_1(\tau_{11}, \ldots, \tau_{1k_1})\ |\ \cdots\ |\ c_n(\tau_{n1}, \ldots, \tau_{nk_n})$$

donde los c_i son símbolos constructores de datos del tipo τ cuya aridad es $ar(c_i) = k_i \geq 0$ para todo i, $1 \leq i \leq n$. Emplearemos en los ejemplos algunos tipos básicos como Nat, Bool, etc. También algunos tipos estructurados como [Nat], el tipo lista de números naturales.

En una implementación por reescritura de un lenguaje funcional [PE93], emplearemos SRTEs tipificados. Los símbolos $f \in \Sigma$ tienen asociado un tipo:

$$f : \tau_1 \times \ldots \times \tau_k \rightarrow \tau$$

El *tipo* de f es $type(f) = \tau$. Los tipos de los argumentos de f vienen recogidos en el conjunto $typearg(f) = \{\tau_1, \ldots, \tau_k\}$. Las variables $x \in V$ también poseen un tipo, $type(x)$. Siempre asumiremos que los términos construidos empleando estos símbolos están bien tipificados. El tipo de un término $t = f(\tilde{t})$ es el tipo del símbolo que se encuentra en la raíz de dicho término: $type(t) = type(f)$.

Ejemplo 11.3.1 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 11.2.1. Siguiendo la notación introducida, caracterizamos $\mathcal{R}$ como un SRT tipificado como sigue:*

$$\begin{array}{lcl} \texttt{dbl} & : & \mathsf{Nat} \to \mathsf{Nat} \\ \texttt{dbls} & : & [\mathsf{Nat}] \to [\mathsf{Nat}] \\ \texttt{sel} & : & \mathsf{Nat} \times [\mathsf{Nat}] \to \mathsf{Nat} \\ \texttt{indx} & : & [\mathsf{Nat}] \times [\mathsf{Nat}] \to [\mathsf{Nat}] \\ \texttt{from} & : & \mathsf{Nat} \to [\mathsf{Nat}] \end{array}$$

De acuerdo con esto, resulta sencillo emplear el Teorema 11.1.4 puesto que, dado un término t de tipo $\tau = type(t)$, podemos establecer un subconjunto máximo de símbolos constructores $\mathcal{B} \subseteq \mathcal{C}$ que formarán parte de cualquier valor de tipo τ que pueda obtenerse como resultado de evaluar t. Para formalizar la diferenciación de símbolos que vamos a realizar, introducimos los siguientes conjuntos:

Notación 11.3.2 *Dado un tipo τ, denotamos como $\mathcal{C}_\tau$ el conjunto de símbolos constructores de dato de tipo τ:*

$$\mathcal{C}_\tau = \{c \in \mathcal{C} \mid type(c) = \tau\}$$

Notación 11.3.3 *El conjunto de símbolos definidos que ocupan posiciones 'externas' en un término t es*

$$outer_{\mathcal{F}}(t) = \begin{cases} \emptyset & si\ t \in V \\ \{f\} & si\ f = root(t) \in \mathcal{F} \\ \cup_{1 \le i \le ar(c)} outer_{\mathcal{F}}(t_i) & si\ c = root(t) \notin \mathcal{F} \end{cases}$$

Notación 11.3.4 *Las variables pertenecientes al conjunto*

$$CVar(t) = Var(\,CCM(t))$$

de variables de t que se encuentran presentes en el contexto constructor máximo de t, se llaman variables constructoras *(o simplemente* $\mathcal{C}$-variables*).*

Notación 11.3.5 *El conjunto de tipos asociados a los símbolos que pueden surgir por instanciación de una variable constructora x es $CVType(type(x))$, donde, dado un tipo τ, $CVType(\tau)$ es el menor conjunto de tipos que satisface la siguiente ecuación:*

$$CVType(\tau) = \{\tau\} \cup \bigcup_{\substack{c \in \mathcal{C}_\tau \\ \tau' \in typearg(c)}} CVType(\tau')$$

Notación 11.3.6 *El conjunto $\mathcal{C}^*_\tau$ de símbolos constructores que pueden encontrarse en algún término de tipo τ es:*

$$\mathcal{C}^*_\tau = \{c \in \mathcal{C} \mid type(c) \in CVType(\tau)\}$$

El siguiente lema relaciona los símbolos constructores que aparecen en el contexto constructor maximal de un término t y el conjunto $\mathcal{C}^*_{type(t)}$.

Lema 11.3.7 *Sea $\Sigma = \mathcal{C} \uplus \mathcal{F}$ una signatura. Sea $t \in \mathcal{T}(\Sigma, V)$ un término (bien tipificado). Si $C[\,] = CCM(t)$, entonces $C[\,] \in \mathcal{T}(\mathcal{C}^*_{type(t)} \cup \{\Box\}, V)$.*

Notación 11.3.8 *El conjunto de símbolos definidos que pueden pasar a ocupar posiciones externas por instanciación de las variables constructoras existentes en un símbolo t es:*

$$Vouter(t) = \bigcup_{x \in CVar(t)} \{f \in \mathcal{F} \mid type(f) \in CVType(type(x))\}$$

Ejemplo 11.3.9 *Consideremos el SRT del Ejemplo 11.2.1. Calculemos el conjunto $Vouter(x)$. Supongamos que $type(x) = \mathsf{Nat}$. Adviértase que $\mathcal{C}^*_{\mathsf{Nat}} = \{\mathsf{0}, \mathsf{s}\}$. Dado que*

$$CVType(type(x)) = CVType(\mathsf{Nat}) = \{\mathsf{Nat}\} \cup CVType(\mathsf{Nat}) = \{\mathsf{Nat}\},$$

obtenemos $Vouter(x) = \{\mathtt{dbl}, \mathtt{sel}\}$.

11.4 Aproximando los símbolos definidos externos presentes en una secuencia de reescritura

Dado un SRTE $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ y un símbolo definido $f \in \mathcal{F}$, nos interesa determinar los símbolos definidos que pueden pasar a ocupar posiciones externas[2] (en el sentido de la Definición 11.2.4) en alguno de los términos de que consta una secuencia de reducción arbitraria que parta de un término t cuyo símbolo a la raíz es precisamente f, es decir, $root(t) = f \in \mathcal{F}$. Para acometer esta tarea, centraremos primero nuestra atención en secuencias simples que consten de un solo paso de reescritura aplicado a un redex $t = f(\tilde{t})$. Podrán surgir nuevos símbolos definidos externos como resultado de los siguientes mecanismos:

1. La aplicación de una regla $f(\tilde{l}) \to r$ a t introduce los nuevos símbolos definidos en posiciones externas que ya se encuentran en la *rhs* r de la regla. El conjunto $outdefrhs_{\mathcal{R}}(f) \subseteq \mathcal{F}$ contiene los símbolos definidos externos que se encuentran en las *rhs*'s de las reglas que definen el símbolo f:

   $$outdefrhs_{\mathcal{R}}(f) = \cup_{f(\tilde{l}) \to r \in R} outer_{\mathcal{F}}(r)$$

[2] Algunos análisis similares al que presentamos aquí pueden encontrarse en [BEØ93, DS87]. Sin embargo, no podemos emplearlos tal cual los proponen los autores pues no son suficientemente precisos para nuestros fines. En dichas aproximaciones, los autores buscan encontrar únicamente el conjunto de símbolos situados *en la raíz* de los posibles reductos de un término dado, o el conjunto de símbolos constructores externos (hasta cierto nivel de profundidad) presentes en cualquier derivación que parte de un término.

2. La aplicación de una regla $f(\tilde{l}) \to r$ a t introduce nuevos símbolos definidos externos debidos a la instanciación de las $\mathcal{C}$-variables existentes en r. Ahora, $instvarrhs_{\mathcal{R}}(f) \subseteq \mathcal{F}$ es el conjunto de dichos símbolos definidos externos para cada rhs de las reglas que definen f:

$$instvarrhs_{\mathcal{R}}(f) = \cup_{f(\tilde{l})\to r\in R} Vouter(r)$$

Notación 11.4.1 *Recogemos todos estos nuevos símbolos definidos externos resultantes de* una *aplicación de alguna de las reglas que definen f en un único conjunto*

$$\begin{aligned} newouter_{\mathcal{R}}(f) &= outdefrhs_{\mathcal{R}}(f) \cup instvarrhs_{\mathcal{R}}(f) \\ &= \cup_{f(\tilde{l})\to r\in R}\,(outer_{\mathcal{F}}(r) \cup Vouter(r)) \end{aligned}$$

Para obtener los símbolos definidos externos que pueden surgir en una secuencia de reducción que arranque en t y que involucre múltiples pasos de reducción elementales, introducimos la siguiente definición.

Definición 11.4.2 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE. Definimos la función $\mathcal{F}_{\mathcal{R}} : \mathcal{P}(\mathcal{F}) \to \mathcal{P}(\mathcal{F})$ como la menor solución de la ecuación:*

$$\mathcal{F}_{\mathcal{R}}(F) = F \;\cup \bigcup_{g\in F} \mathcal{F}_{\mathcal{R}}(newouter_{\mathcal{R}}(g))$$

Proposición 11.4.3 (Continuidad de $\mathcal{F}_{\mathcal{R}}$) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE. Si $\mathcal{G} \subseteq \mathcal{P}(\mathcal{F})$ es una colección de conjuntos de símbolos definidos, entonces*

$$\mathcal{F}_{\mathcal{R}}(\bigcup \mathcal{G}) = \bigcup \mathcal{F}_{\mathcal{R}}(\mathcal{G})$$

DEMOSTRACIÓN.

$$\begin{aligned} \mathcal{F}_{\mathcal{R}}(\textstyle\bigcup \mathcal{G}) &= \textstyle\bigcup \mathcal{G} \cup \bigcup_{g\in\bigcup \mathcal{G}} \mathcal{F}_{\mathcal{R}}(newouter_{\mathcal{R}}(g)) \\ &= \textstyle\bigcup_{G\in\mathcal{G}}[G \cup \bigcup_{g\in G} \mathcal{F}_{\mathcal{R}}(newouter_{\mathcal{R}}(g))] \\ &= \textstyle\bigcup_{G\in\mathcal{G}} \mathcal{F}_{\mathcal{R}}(G) \\ &= \textstyle\bigcup \mathcal{F}_{\mathcal{R}}(\mathcal{G}) \end{aligned}$$

□

La función $\mathcal{F}_{\mathcal{R}}$ es un operador de cierre: la continuidad de $\mathcal{F}_{\mathcal{R}}$ implica su *monotonía* (siempre que $F \subseteq G$, tenemos $\mathcal{F}_{\mathcal{R}}(F) \subseteq \mathcal{F}_{\mathcal{R}}(G)$). Es muy sencillo confirmar, a partir de la definición del operador, que éste es extensivo ($F \subseteq \mathcal{F}_{\mathcal{R}}(F)$ para todo $F \in \mathcal{P}(\mathcal{F})$). A continuación demostramos su idempotencia, empleando para ello una definición equivalente de $\mathcal{F}_{\mathcal{R}}$ que se obtiene fácilmente al desarrollar la solución de la ecuación de definición dada en la Definición 11.4.2 mediante las técnicas usuales de punto fijo.

Proposición 11.4.4 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE. Entonces,*

$$\mathcal{F}_{\mathcal{R}}(F) = \bigcup_{n \geq 0} H_n(F)$$

donde

$$H_0(F) = F \quad y$$

$$H_{n+1}(F) = \bigcup_{g \in H_n(F)} newouter_{\mathcal{R}}(g) \qquad si\ n > 0$$

DEMOSTRACIÓN. La función $\mathcal{F}_{\mathcal{R}}$ es el menor punto fijo de la funcional continua

$$\mathsf{F} = \lambda f.\lambda F.F \cup \bigcup_{g \in F} f(newouter_{\mathcal{R}}(g))$$

que puede obtenerse como el supremo (en el orden parcial completo $[\mathcal{P}(\mathcal{F}) \to \mathcal{P}(\mathcal{F})]$ de las funciones continuas ordenadas mediante el orden $\sqsubseteq$, extensión punto a punto del orden $\subseteq$ sobre $\mathcal{P}(\mathcal{F})$) de la secuencia de Kleene

$$\mathsf{F}^0(f_\perp), \mathsf{F}^1(f_\perp), \ldots, \mathsf{F}^n(f_\perp), \ldots$$

donde $\mathsf{F}^0(f) = f$ y $\mathsf{F}^{n+1}(f) = \mathsf{F}(\mathsf{F}^n(f))$ para todo $f \in \mathcal{P}(\mathcal{F}) \to \mathcal{P}(\mathcal{F})$ y $n \geq 0$. La función $f_\perp \in \mathcal{P}(\mathcal{F}) \to \mathcal{P}(\mathcal{F})$ (elemento mínimo del orden) viene dada por $f_\perp(F) = \emptyset$ para todo $F \in \mathcal{P}(\mathcal{F})$. Formalmente:

$$\mathcal{F}_{\mathcal{R}} = \bigsqcup_{n \geq 0} \mathsf{F}^n(f_\perp)$$

Si desarrollamos esta expresión obtenemos:

$$\begin{array}{rcl}
\mathcal{F}_{\mathcal{R}}(F) & = & F \cup \\
 & & F \cup \bigcup_{g \in F} newouter_{\mathcal{R}}(g) \cup \\
 & & F \cup \bigcup_{g \in F} (newouter_{\mathcal{R}}(g) \cup \bigcup_{g' \in newouter_{\mathcal{R}}(g)} newouter_{\mathcal{R}}(g')) \cup \\
 & & \cdots \\
 & = & F \cup \\
 & & F \cup \bigcup_{g \in F} newouter_{\mathcal{R}}(g) \cup \\
 & & F \cup \bigcup_{g \in F} newouter_{\mathcal{R}}(g) \cup \bigcup_{\substack{g \in F \\ g' \in newouter_{\mathcal{R}}(g)}} newouter_{\mathcal{R}}(g') \cup \\
 & & \cdots \\
 & = & H_0(F) \cup H_1(F) \cup H_2(F) \cup \cdots \\
 & = & \bigcup_{n \geq 0} H_n(F)
\end{array}$$

□

Ahora estamos en condiciones de establecer el siguiente resultado:

Proposición 11.4.5 (Idempotencia de $\mathcal{F}_\mathcal{R}$) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{F} \uplus \mathcal{C}, R)$ un SRTE. Si $F \in \mathcal{P}(\mathcal{F})$, entonces $\mathcal{F}_\mathcal{R}(\mathcal{F}_\mathcal{R}(F)) = \mathcal{F}_\mathcal{R}(F)$.*

DEMOSTRACIÓN. Puesto que $F \subseteq \mathcal{F}_\mathcal{R}(F)$ para cada $F \in \mathcal{P}(\mathcal{F})$, sólo necesitamos demostrar que $\mathcal{F}_\mathcal{R}(\mathcal{F}_\mathcal{R}(F)) \subseteq \mathcal{F}_\mathcal{R}(F)$. Si $g \in \mathcal{F}_\mathcal{R}(\mathcal{F}_\mathcal{R}(F))$, por la Proposición 11.4.4, existe un $q \geq 0$ tal que $g \in H_q(\mathcal{F}_\mathcal{R}(F))$. Demostramos, por inducción sobre q, que $g \in \mathcal{F}_\mathcal{R}(F)$.

Si $q = 0$, entonces $H_q(\mathcal{F}_\mathcal{R}(F)) = H_0(\mathcal{F}_\mathcal{R}(F)) = \mathcal{F}_\mathcal{R}(F)$.

Si $q > 0$, existe un $f \in H_{q-1}(\mathcal{F}_\mathcal{R}(F))$ tal que $g \in newouter_\mathcal{R}(f)$. Por la hipótesis de inducción, $f \in \mathcal{F}_\mathcal{R}(F)$; es decir, existe un $p \geq 0$ tal que $f \in H_p(F)$. Por tanto, $g \in H_{p+1}(F) \subseteq \mathcal{F}_\mathcal{R}(F)$, lo que concluye la demostración. □

Los resultados anteriores nos permiten enunciar el siguiente teorema.

Teorema 11.4.6 *$\mathcal{F}_\mathcal{R}$ es un operador de cierre.*

La función $\mathcal{F}_\mathcal{R}$ permite obtener una cierta aproximación al proceso de reescritura: $\mathcal{F}_\mathcal{R}(F)$ proporciona el conjunto de símbolos definidos externos que pueden aparecer en cualquier secuencia de reducción que arranque en un término t cuyo símbolo a la raíz sea un símbolo definido $root(t) \in F$. Esta información se utiliza para definir una transformación de programas que proporciona una versión extendida del programa original, especialmente adecuada para evaluar términos t cuyo símbolo a la raíz sea un determinado símbolo f. Por este motivo, centraremos nuestra atención en analizar el resultado de aplicar $\mathcal{F}_\mathcal{R}$ sobre el conjunto unitario $\{f\}$.

Definición 11.4.7 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Definimos el conjunto $\mathcal{F}^f_\mathcal{R}$ como*

$$\mathcal{F}^f_\mathcal{R} = \mathcal{F}_\mathcal{R}(\{f\})$$

El conjunto $\mathcal{F}^f_\mathcal{R}$ consta de los símbolos definidos que pueden ocupar posiciones externas en alguna secuencia de reducción que arranque de un término $f(\tilde{t})$ cuyo símbolo a la raíz es precisamente f. Al hacer esto, podemos demostrar la siguiente propiedad que, de hecho, es una definición alternativa de $\mathcal{F}^f_\mathcal{R}$ que no requiere utilizar la función $\mathcal{F}_\mathcal{R}$.

Proposición 11.4.8 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE. Si $f \in \mathcal{F}$, entonces*

$$\mathcal{F}^f_\mathcal{R} = \{f\} \cup \bigcup_{g \in newouter_\mathcal{R}(f)} \mathcal{F}^g_\mathcal{R}$$

DEMOSTRACIÓN. Utilizando la Proposición 11.4.3 escribimos:

$$\begin{array}{rcl} \mathcal{F}^f_\mathcal{R} & = & \mathcal{F}_\mathcal{R}(\{f\}) \\ & = & \{f\} \cup \mathcal{F}_\mathcal{R}(newouter_\mathcal{R}(f)) \\ & = & \{f\} \cup \bigcup_{g \in newouter_\mathcal{R}(f)} \mathcal{F}_\mathcal{R}(\{g\}) \\ & = & \{f\} \cup \bigcup_{g \in newouter_\mathcal{R}(f)} \mathcal{F}^g_\mathcal{R} \end{array}$$

lo cual concluye la demostración. □

La ventaja de esta nueva definición es que sólo necesitamos aplicar las técnicas de punto fijo para resolver *ecuaciones sobre conjuntos* que involucran operaciones simples como la unión de conjuntos. Con la definición original, nos veríamos obligados a trabajar con *ecuaciones sobre funciones de conjuntos.*

Al emplear la Proposición 11.4.8, obtenemos un sistema de ecuaciones sobre conjuntos cuya solución es una colección de conjuntos que satisfacen dichas ecuaciones. Podemos encontrar dichas soluciones, ahora de forma más simple, mediante técnicas de punto fijo [Man74].

Ejemplo 11.4.9 *Consideremos el SRT $\mathcal{R}$ en el Ejemplo 11.2.1. Obtengamos $\mathcal{F}_{\mathcal{R}}^{\mathtt{sel}}$ y $\mathcal{F}_{\mathcal{R}}^{\mathtt{indx}}$.*

1. *Tenemos que*

   $$outdefrhs_{\mathcal{R}}(\mathtt{sel}) = \{\mathtt{sel}\} \text{ e } instvarrhs_{\mathcal{R}}(\mathtt{sel}) = \{\mathtt{sel}, \mathtt{dbl}\}.$$

   También,

   $$outdefrhs_{\mathcal{R}}(\mathtt{dbl}) = \{\mathtt{dbl}\} \text{ e } instvarrhs_{\mathcal{R}}(\mathtt{dbl}) = \emptyset.$$

   Por tanto, $newouter_{\mathcal{R}}(\mathtt{sel}) = \{\mathtt{sel}, \mathtt{dbl}\}$ y, empleando la Proposición 11.4.8, obtenemos:

   $$\mathcal{F}_{\mathcal{R}}^{\mathtt{sel}} = \{\mathtt{sel}\} \cup \mathcal{F}_{\mathcal{R}}^{\mathtt{sel}} \cup \mathcal{F}_{\mathcal{R}}^{\mathtt{dbl}}.$$

   También,

   $$\mathcal{F}_{\mathcal{R}}^{\mathtt{dbl}} = \{\mathtt{dbl}\} \cup \mathcal{F}_{\mathcal{R}}^{\mathtt{dbl}} = \{\mathtt{dbl}\}.$$

   Por tanto,

   $$\mathcal{F}_{\mathcal{R}}^{\mathtt{sel}} = \{\mathtt{sel}, \mathtt{dbl}\}.$$

2. *Tenemos $outdefrhs_{\mathcal{R}}(\mathtt{indx}) = \{\mathtt{sel}, \mathtt{indx}\}$ e $instvarrhs_{\mathcal{R}}(\mathtt{indx}) = \emptyset$. Entonces, tendremos que $newouter_{\mathcal{R}}(\mathtt{sel}) = \{\mathtt{sel}, \mathtt{indx}\}$ y:*

   $$\mathcal{F}_{\mathcal{R}}^{\mathtt{indx}} = \{\mathtt{indx}\} \cup \mathcal{F}_{\mathcal{R}}^{\mathtt{indx}} \cup \mathcal{F}_{\mathcal{R}}^{\mathtt{sel}} = \{\mathtt{indx}, \mathtt{sel}, \mathtt{dbl}\}$$

De manera informal, los resultados del Ejemplo 11.4.9 pueden obtenerse de forma rápida si conjeturamos los conjuntos $X = \mathcal{F}_{\mathcal{R}}^{\mathtt{sel}}$, $Y = \mathcal{F}_{\mathcal{R}}^{\mathtt{dbl}}$, y $Z = \mathcal{F}_{\mathcal{R}}^{\mathtt{sel}}$ que son las soluciones *mínimas* (en el sentido de la inclusión conjuntista) de las ecuaciones

$$\begin{aligned} X &= \{\mathtt{sel}\} \cup X \cup Y \\ Y &= \{\mathtt{dbl}\} \cup Y \\ Z &= \{\mathtt{indx}\} \cup Z \cup X \end{aligned}$$

Estos conjuntos pueden calcularse como el menor punto fijo de la función

$$\mathsf{G} = \lambda\langle X, Y, Z\rangle.\langle\{\mathtt{sel}\} \cup X \cup Y, \{\mathtt{dbl}\} \cup Y, \{\mathtt{indx}\} \cup Z \cup X\rangle$$

mediante la secuencia de Kleene:

$$\begin{aligned} \mathsf{G}^0(\langle\emptyset,\emptyset,\emptyset\rangle) &= \langle\emptyset,\emptyset,\emptyset\rangle \\ \mathsf{G}^1(\langle\emptyset,\emptyset,\emptyset\rangle) &= \langle\{\mathtt{sel}\},\{\mathtt{dbl}\},\{\mathtt{indx}\}\rangle \\ \mathsf{G}^2(\langle\emptyset,\emptyset,\emptyset\rangle) &= \langle\{\mathtt{sel},\mathtt{dbl}\},\{\mathtt{dbl}\},\{\mathtt{indx},\mathtt{sel}\}\rangle \\ \mathsf{G}^3(\langle\emptyset,\emptyset,\emptyset\rangle) &= \langle\{\mathtt{sel},\mathtt{dbl}\},\{\mathtt{dbl}\},\{\mathtt{indx},\mathtt{sel},\mathtt{dbl}\}\rangle \\ \mathsf{G}^4(\langle\emptyset,\emptyset,\emptyset\rangle) &= \mathsf{G}^3(\langle\emptyset,\emptyset,\emptyset\rangle) \end{aligned}$$

La siguiente proposición establece que el conjunto $\mathcal{F}_{\mathcal{R}}^f$ contiene los componentes $\mathcal{F}_{\mathcal{R}}^g$ de cada miembro $g \in \mathcal{F}_{\mathcal{R}}^f$.

Proposición 11.4.10 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Si $g \in \mathcal{F}_{\mathcal{R}}^f$, entonces $\mathcal{F}_{\mathcal{R}}^g \subseteq \mathcal{F}_{\mathcal{R}}^f$.*

DEMOSTRACIÓN. Dado que $g \in \mathcal{F}_{\mathcal{R}}^f$, tenemos $\{g\} \subseteq \mathcal{F}_{\mathcal{R}}(\{f\})$. Por monotonía de $\mathcal{F}_{\mathcal{R}}$, $\mathcal{F}_{\mathcal{R}}(\{g\}) \subseteq \mathcal{F}_{\mathcal{R}}(\mathcal{F}_{\mathcal{R}}(\{f\}))$. Por idempotencia, $\mathcal{F}_{\mathcal{R}}(\{g\}) \subseteq \mathcal{F}_{\mathcal{R}}(\{f\})$. Por tanto, $\mathcal{F}_{\mathcal{R}}^g \subseteq \mathcal{F}_{\mathcal{R}}^f$. □

Proposición 11.4.11 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Si $g \in \mathcal{F}_{\mathcal{R}}^f$, entonces $\mathcal{C}^*_{type(g)} \subseteq \mathcal{C}^*_{type(f)}$.*

11.5 Corrección de la aproximación

Afirmamos que, al reescribir mediante un SRTE $\mathcal{R}$ un término t que verifica la condición $root(t) = f$, únicamente encontraremos términos cuyos símbolos definidos externos pertenezcan al conjunto $\mathcal{F}_{\mathcal{R}}^f$. De hecho, tal como demostramos más abajo, esto también es cierto para cualquier secuencia de reducción iniciada a partir de un término $t = g(\tilde{t})$ tal que $g \in \mathcal{F}_{\mathcal{R}}^f$, para un símbolo $f \in \mathcal{F}$ determinado.

Podemos ser más concretos aún y caracterizar el conjunto $ev^f(\Sigma, V)$ de términos que intervienen en las derivaciones mencionadas.

Definición 11.5.1 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Se define el conjunto $ev^f(\Sigma, V) \subseteq \mathcal{T}(\Sigma, V)$ como el menor conjunto que satisface las condiciones:*

$V \subseteq ev^f(\Sigma, V)$

$g(\tilde{t}) \in ev^f(\Sigma, V)$ *si* $g \in \mathcal{F}^f_{\mathcal{R}}$ *y* $t_1, \ldots, t_{ar(g)} \in \mathcal{T}(\Sigma, V)$

$c(\tilde{t}) \in ev^f(\Sigma, V)$ *si* $c \in \mathcal{C}^*_{type(f)}$ *y* $t_1, \ldots, t_{ar(c)} \in ev^f(\Sigma, V)$

(Como en el resto del capítulo, asumimos que la construcción de estos términos respeta las reglas de tipificación.)

Si no exigimos $V \subseteq ev^f(\Sigma, V)$, estaremos definiendo el conjunto $gev^f(\Sigma, V)$.

Definición 11.5.2 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Se define el conjunto $gev^f(\Sigma, V) \subseteq \mathcal{T}(\Sigma, V)$ como el menor conjunto que satisface las condiciones:*

$g(\tilde{t}) \in gev^f(\Sigma, V)$ *si* $g \in \mathcal{F}^f_{\mathcal{R}}$ *y* $t_1, \ldots, t_{ar(g)} \in \mathcal{T}(\Sigma, V)$

$c(\tilde{t}) \in gev^f(\Sigma, V)$ *si* $c \in \mathcal{C}^*_{type(f)}$ *y* $t_1, \ldots, t_{ar(c)} \in gev^f(\Sigma, V)$

Los términos que integran el conjunto $gev^f(\Sigma, V)$ no son necesariamente básicos. En particular, $gev^f(\Sigma, V) \supseteq ev^f(\Sigma, \emptyset)$. No obstante, es evidente que $ev^f(\Sigma, V) \cap \mathcal{T}(\Sigma, \emptyset) \subseteq gev^f(\Sigma, V)$. Los términos en $gev^f(\Sigma, V)$ son 'básicos' en el siguiente sentido.

Proposición 11.5.3 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Entonces, $t \in gev^f(\Sigma, V)$ si y sólo si $t \in ev^f(\Sigma, V)$ y t no contiene $\mathcal{C}$-variables.*

DEMOSTRACIÓN. Por inducción sobre la estructura de t en $gev^f(\Sigma, V)$. El caso base, cuando $t = g(\tilde{t})$ para algún $g \in \mathcal{F}^f_{\mathcal{R}}$, es inmediato. Para dar el paso de inducción, sea $t = c(\tilde{t})$ con $c \in \mathcal{C}^*_{type(f)}$. Por la hipótesis de inducción, $t_i \in gev^f(\Sigma, V)$ si y sólo si $t_i \in ev^f(\Sigma, V)$ y t_i no contiene $\mathcal{C}$-variables, para $1 \leq i \leq ar(c)$. Esto completa la demostración. □

La siguiente proposición establece que $ev^f(\Sigma, V)$ posee la propiedad deseada: es cerrado bajo reducción.

Proposición 11.5.4 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Sean $t \in ev^f(\Sigma, V)$ y $s \in \mathcal{T}(\Sigma, V)$. Si $t \to s$, entonces $s \in ev^f(\Sigma, V)$.*

DEMOSTRACIÓN. Por inducción en la longitud $|u|$ de la ocurrencia redex $u \in O_{\mathcal{R}}(t)$ que se considera para reducir el término.

Si $|u| = 0$, entonces $u = \epsilon$ y $t = g(\tilde{t})$, con $g \in \mathcal{F}^f_{\mathcal{R}}$, es un redex $t = \sigma(l)$ de la regla $l \to r \in R$ y $s = \sigma(r)$. Sea $C[\,] = CCM(s) \in \mathcal{T}(\mathcal{C}^*_{type(f)} \cup \{\Box\}, V)$. Mostraremos que $s \in ev^f(\Sigma, V)$ por inducción sobre la estructura de $C[\,]$.

1. Si $C[\,] = x \in V$ es inmediato. Si $C[\,] = \Box$, entonces $g' = root(s) \in newouter_{\mathcal{R}}(g)$ y, por la Proposición 11.4.8, $g' \in \mathcal{F}^g_{\mathcal{R}}$. Dado que $g \in \mathcal{F}^f_{\mathcal{R}}$, por la Proposición 11.4.10, tenemos que $g' \in \mathcal{F}^g_{\mathcal{R}} \subseteq \mathcal{F}^f_{\mathcal{R}}$. Por tanto $s \in ev^f(\Sigma, V)$.

2. Si $C[\,] = c(C_1[\,], \ldots, C_{ar(c)}[\,])$, entonces $s = c(\tilde{s})$ y, por el Lema 11.3.7, $c \in \mathcal{C}^*_{type(s)}$. Puesto que las reglas de reescritura están bien tipificadas y la reducción se lleva a cabo a la raíz del término, $type(s) = type(t) = type(g)$. Así, $c \in \mathcal{C}^*_{type(g)}$ y, por la Proposición 11.4.11, $\mathcal{C}^*_{type(g)} \subseteq \mathcal{C}^*_{type(f)}$; es decir, $c \in \mathcal{C}^*_{type(f)}$. Por definición de CCM, $C_i[\,] = CCM(s_i)$ para $1 \leq i \leq ar(c)$ y, por la hipótesis de inducción, se tiene $s_i \in ev^f(\Sigma, V)$ para $1 \leq i \leq ar(c)$. Por tanto, por definición de $ev^f(\Sigma, V)$, se sigue la conclusión.

Si $|u| > 0$, cuando $root(s) = root(t) \in \mathcal{F}^f_{\mathcal{R}}$, por definición de $ev^f(\Sigma, V)$, obtenemos el resultado deseado.

Por otro lado, si $t = c(\tilde{t})$ y $c \in \mathcal{C}^*_{type(f)}$, entonces $t_i \in ev^f(\Sigma, V)$ para $1 \leq i \leq ar(c)$. Existe un subtérmino t_i, para algún i, $1 \leq i \leq ar(c)$, tal que $t_i \to s_i$. Entonces, por la hipótesis de inducción, $s_i \in ev^f(\Sigma, V)$. Puesto que $t_j = s_j$ para todo $j \neq i$, $t_i \in ev^f(\Sigma, V)$ para $1 \leq i \leq ar(c)$. Esto concluye la demostración. □

Teorema 11.5.5 (Corrección de la aproximación) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Sea $t \in ev^f(\Sigma, V)$ y $s \in \mathcal{T}(\Sigma, V)$. Si $t \to^* s$, entonces $s \in ev^f(\Sigma, V)$.*

DEMOSTRACIÓN. Por inducción sobre la longitud m de la derivación $t \to^* s$, y utilizando la Proposición 11.5.4. □

El Teorema 11.5.5 también implica que el conjunto $\mathcal{F}^f_{\mathcal{R}}$ es cerrado bajo aplicación de la operación *root* sobre subtérminos maximales (cuyo símbolo a la raíz es no constructor) contenidos en reductos de términos t que satisfacen $root(t) \in \mathcal{F}^f_{\mathcal{R}}$.

Observación 11.5.6 *El Teorema 11.5.5 permite afirmar que $\mathcal{F}^f_{\mathcal{R}}$ contiene* todos *los símbolos que aparecen en posiciones externas dentro de los términos presentes en cualquier secuencia de reducción que parta de un término t cuyo símbolo a la raíz sea f. De otro modo, si existiera un símbolo $g \in \mathcal{F} \backslash \mathcal{F}^f_{\mathcal{R}}$ externo en algún término s, tal que $t \to^* s$, se tendría que $s \notin ev^f(\Sigma, V)$, violándose el citado teorema.*

El conjunto $gev^f(\Sigma, V)$ no es cerrado bajo reescritura. Por ejemplo, si consideramos una regla $g(\tilde{l}) \to x$ del SRTE, y un símbolo $g \in \mathcal{F}^f_{\mathcal{R}}$ para algún $f \in \mathcal{F}$, entonces $g(\tilde{l}) \in gev^f(\Sigma, V)$, pero $g(\tilde{l}) \to x$ y $x \notin gev^f(\Sigma, V)$. Sin embargo, $gev^f(\Sigma, V)$ es cerrado bajo *reducción inversa*.

Proposición 11.5.7 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Sean $t \in ev^f(\Sigma, V)$ y $s \in gev^f(\Sigma, V)$. Si $t \to^* s$, entonces $t \in gev^f(\Sigma, V)$.*

DEMOSTRACIÓN. Por contradicción: si $t \notin gev^f(\Sigma, V)$, entonces, por la Proposición 11.5.3, t contiene una ocurrencia de una $\mathcal{C}$-variable. Esta ocurrencia de variable no puede ser eliminada por reducción, puesto que sólo es posible encontrar símbolos constructores por encima de ella. Por tanto, s contiene una $\mathcal{C}$-variable y, de acuerdo con la Proposición 11.5.3, $s \notin gev^f(\Sigma, V)$. □

11.6 La transformación V

Hemos introducido una diferenciación de símbolos externos e internos a lo largo de una secuencia de reducción para emplearla en la definición de la transformación V que presentamos más abajo.

El tratamiento formal realizado hasta aquí podría simplificarse si no persiguiéramos el incremento mínimo de las reglas de reescritura añadidas al sistema transformado. Si introducimos nuevas reglas $g'(\tilde{l}) \to r'$ para dar semántica a cada nuevo símbolo definido g' que representa la versión externa de un determinado símbolo g *para cada* $g \in \mathcal{F}$ (en vez de para cada $g \in \mathcal{F}^f_{\mathcal{R}}$ como vamos a hacer), la definición de la transformación sería más simple ya que no necesitaríamos muchas de las notaciones y resultados establecidos en las Secciones 11.4 y 11.5. Sin embargo, procediendo así podemos encontrarnos con situaciones peligrosas.

Ejemplo 11.6.1 *Al evaluar* $t =$ `indx(s(0)::[],dbls(from(0)))` *tal como se ha mostrado en el Ejemplo 11.2.2, el símbolo* `from` nunca *llega a ocupar posiciones externas (adviértase que* `from` $\notin \mathcal{F}^{\mathtt{indx}}_{\mathcal{R}}$ *como puede verse en el Ejemplo 11.4.9). Por tanto, no necesitamos la siguiente regla (el significado de* $\bar{\mathtt{T}}$ *se aclarará en breve):*

`from'(x)` $\to$ $\bar{\mathtt{T}}$`(x)::'from'(s(x))`

que introduce un comportamiento no terminante, *debido a que necesitamos imponer que* `::'` *(que va a representar la versión externa de* `::`*) sea reemplazante tanto en su primer como en su segundo argumento, con el fin de posibilitar la evaluación completa de* t. *No obstante, esta regla* no *añade potencia computacional a la transformación, porque no va a surgir ninguna ocurrencia externa de* `from` *en ninguna secuencia de reducción que arranque del término* t *(y, por tanto, no es necesario introducir la 'versión externa',* `from'` *de* `from`*).*

Por este motivo, hemos efectuado un análisis más preciso que permite realizar una selección más fina de las reglas que necesitamos añadir para alcanzar nuestro objetivo.

La definición de la trasformación de programas que vamos a presentar empieza con la introducción de una nueva signatura.

Definición 11.6.2 *Sea* $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ *un SRTE y* $f \in \mathcal{F}$. *La signatura* $\Sigma^f = \Sigma \uplus \mathcal{C}' \uplus \mathcal{F}'$, *es tal que*

$$c' \in \mathcal{C}' \Leftrightarrow c \in \mathcal{C}^*_{type(f)} \wedge ar(c') = ar(c)$$

y

$$g' \in \mathcal{F}' \Leftrightarrow g \in \mathcal{F}^f_{\mathcal{R}} \wedge ar(g') = ar(g)$$

estando tanto $\mathcal{C}'$ *y* $\mathcal{C}^*_{type(f)}$, *por una parte, como* $\mathcal{F}'$ *y* $\mathcal{F}^f_{\mathcal{R}}$, *por otra, en correspondencia biunívoca.*

En la Definición 11.6.2, los nuevos constructores $c' \in \mathcal{C}'$ representan símbolos constructores $c \in \mathcal{C}$ situados sobre símbolos definidos externos. Los símbolos $g' \in \mathcal{F}'$ representan aplicaciones externas de $g \in \mathcal{F}^f_{\mathcal{R}}$. Definimos una función τ^f para transformar términos de $ev^f(\Sigma, V)$ en términos de $\mathcal{T}(\Sigma^f, V)$:

Definición 11.6.3 *Sea* $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ *un SRTE y* $f \in \mathcal{F}$. *Se define*

$$\tau^f : ev^f(\Sigma, V) \rightarrow \mathcal{T}(\Sigma^f, V)$$

como sigue:

$$\begin{array}{lcl} \tau^f(x) & = & x \\ \tau^f(g(\tilde{t})) & = & g'(\tilde{t}) \ si \ g \in \mathcal{F}^f_{\mathcal{R}} \\ \tau^f(c(\tilde{t})) & = & c'(\tau^f(\tilde{t})) \ si \ c \in \mathcal{C}^*_{type(f)} \end{array}$$

El conjunto de ocurrencias de un término dado coincide con el conjunto de ocurrencias del término transformado por τ^f.

Lema 11.6.4 *Sea* $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ *un SRTE y* $f \in \mathcal{F}$. *Si* $t \in ev^f(\Sigma, V)$, *entonces* $O(\tau^f(t)) = O(t)$.

La función τ^f distingue los símbolos definidos externos de los no externos y las ocurrencias externas de símbolos constructores de las interiores. Sin embargo, no distingue las variables constructoras de las demás. Una leve modificación de τ^f lo hará.

Definición 11.6.5 *Sea* $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ *un SRTE y* $f \in \mathcal{F}$. *Se define*

$$\kappa^f : ev^f(\Sigma, V) \rightarrow \mathcal{T}(\Sigma^f \uplus \{\bar{\mathtt{T}}\}, V)$$

que emplea un símbolo nuevo $\bar{\mathtt{T}}$ *de aridad uno:*

$$\begin{array}{lcl} \kappa^f(x) & = & \bar{\mathtt{T}}(x) \\ \kappa^f(g(\tilde{t})) & = & g'(\tilde{t}) \ si \ g \in \mathcal{F}^f_{\mathcal{R}} \\ \kappa^f(c(\tilde{t})) & = & c'(\tau^f(\tilde{t})) \ si \ c \in \mathcal{C}^*_{type(f)} \end{array}$$

La función κ^f distingue las ocurrencias de variables constructoras de las demas ocurrencias de variables. Esto se consigue mediante el nuevo símbolo $\bar{\mathtt{T}}$ que etiqueta las ocurrencias de variables constructoras presentes en un término $t \in ev^f(\Sigma, V)$. Tanto τ^f como κ^f son inyectivas (esto puede demostrarse fácilmente por inducción sobre $ev^f(\Sigma, V)$). Las funciones τ^f y κ^f sólo coinciden al actuar sobre $gev^f(\Sigma, V)$.

Lema 11.6.6 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Sea $t \in ev^f(\Sigma, V)$. Entonces, $\tau^f(t) = \kappa^f(t)$ si y sólo si $t \in gev^f(\Sigma, V)$.*

El conjunto de ocurrencias no variables de un término coincide con el conjunto de ocurrencias no variables del término transformado por κ^f.

Lema 11.6.7 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Si $t \in ev^f(\Sigma, V)$, entonces $O_\Sigma(\kappa^f(t)) = O_\Sigma(t)$.*

La transformación que vamos a presentar introduce nuevas reglas en el SRTE para diferenciar, de forma efectiva, el tratamiento de las distintas clases de símbolos que acabamos de formalizar. Las reglas para los representantes $f' \in \mathcal{F}'$ de los símbolos definidos externos introducen nuevos representantes $c' \in \mathcal{C}'$ y $f' \in \mathcal{F}'$ de constructores externos y símbolos definidos externos, respectivamente. Las reglas para los símbolos definidos internos son las ya presentes en el SRTE original.

Definición 11.6.8 (Transformación V) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Se define el SRTE $\mathsf{V}^f(\mathcal{R}) = (\Sigma^f \uplus \{\bar{\mathtt{T}}\}, R \cup S^f \cup T^f)$, donde:*

1. *$\bar{\mathtt{T}}$ es un nuevo símbolo de aridad uno (un símbolo por cada tipo empleado en las reglas del SRTE; no los distinguiremos más en lo sucesivo), es decir, $ar(\bar{\mathtt{T}}) = 1$.*

2. *El conjunto de reglas S^f se define como sigue:*

$$S^f = \{g'(\tilde{l}) \to \bar{\mathtt{T}}(r) \mid g(\tilde{l}) \to r \in R \wedge g \in \mathcal{F}^f_{\mathcal{R}}\}$$

3. *El conjunto de reglas T^f se define como sigue*

$$\begin{aligned} T^f \quad = \quad & \{\bar{\mathtt{T}}(c(x_1, \ldots, x_{ar(c)})) \to c'(\bar{\mathtt{T}}(x_1), \ldots, \bar{\mathtt{T}}(x_{ar(c)})) \mid c \in \mathcal{C}^*_{type(f)}\} \cup \\ & \{\bar{\mathtt{T}}(g(x_1, \ldots, x_{ar(g)})) \to g'(x_1, \ldots, x_{ar(g)}) \mid g \in \mathcal{F}^f_{\mathcal{R}}\} \end{aligned}$$

El símbolo $\bar{\mathtt{T}}$ de la Definición 11.6.8 es un nuevo símbolo cuyo comportamiento viene descrito por las reglas del conjunto T^f (que constituyen un SRT). Su misión es realizar, en tiempo de ejecución, el renombramiento de símbolos constructores $c \in \mathcal{C}^*_{type(f)}$ como constructores $c' \in \mathcal{C}'$. Las aplicaciones externas de los símbolos definidos $g \in \mathcal{F}^f$ son reconvertidas a aplicaciones de los símbolos $g' \in \mathcal{F}'$ correspondientes. Las reglas pertenecientes al conjunto S^f asignan significado a los símbolos $g' \in \mathcal{F}'$, imitando la

semántica establecida en el conjunto R del SRTE original para los símbolos $g \in \mathcal{F}^f_{\mathcal{R}}$ correspondientes.

Para simplificar la notación empleada, escribiremos V, en vez de V^f, cuando el parámetro f de la transformación quede bien determinado por el contexto donde se hace uso de ella. Los siguientes ejemplos sirven para ilustrar la transformación. No obstante, por simplicidad, en estos ejemplos utilizamos la versión *desplegada* de la transformación V (ver la Sección 11.7).

Ejemplo 11.6.9 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 11.2.1. Si tomamos $f =$* `sel` *(cuyo tipo es* Nat*), entonces $\mathcal{C}^*_{\mathsf{Nat}} = \{\mathtt{0},\mathtt{s}\} \subseteq \{\mathtt{0},\mathtt{s},\mathtt{[]},\mathtt{::}\} = \mathcal{C}$ y $\mathcal{F}^{\mathtt{sel}}_{\mathcal{R}} = \{\mathtt{sel},\mathtt{dbl}\}$ (ver el Ejemplo 11.4.9). Las reglas* adicionales *necesarias para evaluar un término $t =$* `sel(`*Num*`,`*ListNum*`)` *de acuerdo con la transformación* V *son:*

```
dbl'(0)     → 0'                 sel'(0,x::y)     → T̄(x)
dbl'(s(x)) → s'(s'(dbl'(x)))     sel'(s(x),y::z) → sel'(x,z)

T̄(0)        → 0'                 T̄(dbl(x))        → dbl'(x)
T̄(s(x))     → s'(T̄(x))           T̄(sel(x,y))      → sel'(x,y)
```

que de hecho son suficientes (y válidas) para evaluar términos como `dbl(`*Num*`)` *(ver la Sección 11.7) .*

Ejemplo 11.6.10 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 11.2.1. Si $f =$* `indx` *(cuyo tipo es* [Nat]*), entonces $\mathcal{C}^*_{[\mathsf{Nat}]} = \{\mathtt{0},\mathtt{s},\mathtt{[]},\mathtt{::}\} = \mathcal{C}$ y $\mathcal{F}^{\mathtt{indx}}_{\mathcal{R}} = \{\mathtt{indx},\mathtt{sel},\mathtt{dbl}\}$ (ver el Ejemplo 11.4.9). Las reglas* adicionales *necesarias para evaluar un término $t =$* `indx(`*ListNum*`,`*ListNum*`)` *son:*

```
dbl'(0)          → 0'
dbl'(s(x))       → s'(s'(dbl'(x)))
sel'(0,x::y)     → T̄(x)
sel'(s(x),y::z) → sel'(x,z)
indx'([],x)      → []'
indx'(x::y,z)    → sel'(x,z)::'indx'(y,z)

T̄(0)     → 0'                  T̄(dbl(x))     → dbl'(x)
T̄(s(x)) → s'(T̄(x))            T̄(indx(x,y)) → indx'(x,y)
T̄([])    → []'                 T̄(sel(x,y))   → sel'(x,y)
T̄(x::y) → T̄(x)::'T̄(y)
```

Dichas reglas también son adecuadas para evaluar términos como `sel(`*Num*`,`*ListNum*`)` *y* `dbl(`*Num*`)`*.*

Dada una función de reemplazamiento μ, definimos la función de reemplazamiento $rmap^f(\mu)$, que emplearemos con el SRTE transformado $\mathsf{V}^f(\mathcal{R})$ para conseguir la

evaluación completa de términos $f(\tilde{t})$ bajo la asunción de que μ es suficiente para obtener *formas estables constructoras* en $\mathcal{R}$.

Definición 11.6.11 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mu \in M_\Sigma$ y $f \in \mathcal{F}$. La función $rmap^f : M_\Sigma \to M_{\Sigma^f \uplus \{\bar{\mathtt{T}}\}}$ viene dada por: $rmap^f(\mu) = \mu \sqcup \mu_{\mathcal{F}'} \sqcup \mu_{\mathcal{C}'}$, donde*

- *$\mu_{\mathcal{F}'}(g') = \mu(g)$ para todo $g' \in \mathcal{F}'$ que corresponde a $g \in \mathcal{F}^f_\mathcal{R}$, y $\mu_{\mathcal{F}'}(g) = \emptyset$ si $g \notin \mathcal{F}'$.*
- *$\mu_{\mathcal{C}'}(c') = \mathbb{N}^+_{ar(c')}$ para todo $c' \in \mathcal{C}'$, y $\mu_{\mathcal{C}'}(g) = \emptyset$ si $g \notin \mathcal{C}'$.*

Observemos que si $\mu' = rmap^f(\mu)$, entonces $\mu'(\bar{\mathtt{T}}) = \emptyset$.

Ejemplo 11.6.12 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 11.2.1. Si $f = \mathtt{indx}$, $\mathcal{C}^*_{[\mathsf{Nat}]} = \{\mathtt{0}, \mathtt{s}, \mathtt{[]}, \mathtt{::}\}$ y $\mu' = rmap^f(\mu^{com}_\mathcal{R})$, entonces*

$$\mu'(\mathtt{s}) = \mu'(\mathtt{::}) = \mu'(\mathtt{from}) = \mu'(\bar{\mathtt{T}}) = \emptyset,$$

$$\mu'(\mathtt{s'}) = \mu'(\mathtt{dbl}) = \mu'(\mathtt{dbl'}) = \mu'(\mathtt{dbls}) = \mu'(\mathtt{indx}) = \mu'(\mathtt{indx'}) = \{1\},$$

$$\mu'(\mathtt{::'}) = \mu'(\mathtt{sel}) = \mu'(\mathtt{sel'}) = \{1, 2\}$$

Las ocurrencias de redexes μ-reemplazantes de $t = f(\tilde{t})$ (respecto al SRTE $\mathcal{R}$) son exactamente las mismas que las ocurrencias de redexes $rmap^f(\mu)$-reemplazantes de $\tau^f(t)$ (respecto al SRTE $\mathsf{V}(\mathcal{R})$). Por tanto, bajo el control de la función de reemplazamiento $rmap^f(\mu)$, y en posiciones situadas bajo un símbolo definido externo, las reducciones sensibles al contexto tienen lugar exactamente en la misma medida que bajo el control de la función de reemplazamiento μ. De esta manera evitamos reducciones inútiles. Este hecho viene formalizado por la siguiente proposición.

Proposición 11.6.13 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mu \in M_\Sigma$ y $f \in \mathcal{F}$. Sea $t \in ev^f(\Sigma, V)$ y $\mu' = rmap^f(\mu)$. Sea $C[\,] = CCM(t)$ y $t = C[t_1, \ldots, t_n]$ donde $u_1, \ldots, u_n \in O(t)$ son tales que $t|_{u_i} = t_i$ para $1 \leq i \leq n$. Entonces,*

$$O^{\mu'}_{\mathsf{V}(\mathcal{R})}(\tau^f(t)) = \cup_{1 \leq i \leq n} u_i . O^\mu_\mathcal{R}(t_i) \quad y \quad O^{\mu'}_{\mathsf{V}(\mathcal{R})}(\kappa^f(t)) = \cup_{1 \leq i \leq n} u_i . O^\mu_\mathcal{R}(t_i)$$

Demostración. Por inducción sobre la estructura de $t \in ev^f(\Sigma, V)$. Si $t = x \in V$, es inmediato. Si $t = g(\tilde{t})$, con $g \in \mathcal{F}^f_\mathcal{R}$, entonces $C[\,] = CCM(t) = \Box$, y las ocurrencias $u_1, \ldots, u_n$ de los huecos del contexto $C[\,]$ coinciden en una única ocurrencia ϵ. Siendo $t = g(\tilde{t})$, con $g \in \mathcal{F}^f_\mathcal{R}$, tenemos que $\tau^f(t) = g'(\tilde{t})$. Por el Lema 11.6.4, $O(\tau^f(t)) = O(t)$ y, puesto que $\mu'(g') = \mu(g)$ para todo $g \in \mathcal{F}^f_\mathcal{R}$ y $\mu'(h) = \mu(h)$ para cada $h \in \Sigma$, también tenemos que $O^{\mu'}(\tau^f(t)) = O^{\mu'}(\tau^f(g(\tilde{t}))) = O^{\mu'}(g'(\tilde{t})) = O^\mu(g(\tilde{t})) = O^\mu(t)$. Por definición de $\mathsf{V}(\mathcal{R})$, un subtérmino $\tau^f(t)|_u$ es un redex de $\mathsf{V}(\mathcal{R})$ si y sólo si el subtérmino $t|_u$ es un redex de $\mathcal{R}$. Esto se debe a que, si la ocurrencia del redex es

$u = \epsilon$, entonces, puesto que $g \in \mathcal{F}^f_{\mathcal{R}}$, y de acuerdo con la Definición 11.6.8, a una regla $g(\tilde{l}) \to r \in R$ tal que $t = \sigma(g(\tilde{l}))$ le corresponde una regla $g'(\tilde{l}) \to \bar{\mathtt{T}}(r)$ en el componente S^f de $\mathsf{V}(\mathcal{R})$ y viceversa. También, si $u > \epsilon$, únicamente las reglas de R (que están todas presentes en $\mathsf{V}(\mathcal{R})$), pueden ser aplicadas tanto a t como a $\tau^f(t)$. Por tanto, tenemos que $O^{\mu'}_{\mathsf{V}(\mathcal{R})}(\tau^f(t)) = O^{\mu}_{\mathcal{R}}(t) = \epsilon.O^{\mu}_{\mathcal{R}}(t)$.

Si $t = c(\tilde{t})$, con $c \in \mathcal{C}^*_{type(f)}$, entonces $C[\,] = c(C_1[\,], \ldots, C_{ar(c)}[\,])$ y $C_i[\,] = CCM(t_i)$ para $1 \leq i \leq ar(c)$. Entonces, por definición de τ^f, $\tau^f(c(\tilde{t})) = c'(\tau^f(\tilde{t}))$. El conjunto $\{u_1, \ldots, u_n\}$ puede descomponerse como $\{u_{11}, \ldots, u_{1m_1}, \ldots, u_{k1}, \ldots, u_{km_k}\}$, donde se verifica que $n = m_1 + \cdots + m_k$ y $u_{ij} = i.u'_{ij}$ para $1 \leq i \leq k = ar(c)$ y $1 \leq j \leq m_i$. Por la hipótesis de inducción, podemos asumir que $O^{\mu'}_{\mathsf{V}(\mathcal{R})}(\tau^f(t_i)) = \cup_{1 \leq j \leq m_i} u'_{ij}.O^{\mu}_{\mathcal{R}}(t_{ij})$. Por definición de μ', $\mu'(c') = \mathbb{N}^+_{ar(c')}$. Por tanto, por definición del conjunto de ocurrencias reemplazantes, $O^{\mu'}(\tau^f(t)) = \cup_{1 \leq i \leq ar(c')} i.O^{\mu'}(\tau^f(t_i))$. Dado que c' es un constructor, cada redex en $\tau^f(t)$ se encuentra también en algún $\tau^f(t_i)$ para $1 \leq i \leq ar(c')$. De aquí se sigue la conclusión para τ^f.

Respecto a κ^f, concluimos el resultado enunciado empleando el hecho de que los redexes están siempre situados sobre ocurrencias no variables y el Lema 11.6.7. □

Corolario 11.6.14 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mu \in M_\Sigma$ y $f \in \mathcal{F}$. Sea $t \in ev^f(\Sigma, V)$ tal que $root(t) \in \mathcal{F}^f_{\mathcal{R}}$ y $\mu' = rmap^f(\mu)$. Entonces, $O^{\mu'}_{\mathsf{V}(\mathcal{R})}(\tau^f(t)) = O^{\mu}_{\mathcal{R}}(t)$ ($O^{\mu'}_{\mathsf{V}(\mathcal{R})}(\kappa^f(t)) = O^{\mu}_{\mathcal{R}}(t)$).*

Para emplear $\mathsf{V}(\mathcal{R})$ en la evaluación de un término t, podemos reducir directamente $\bar{\mathtt{T}}(t)$ en lugar de aplicar primero τ^f sobre t. Esta propiedad se formaliza en la próxima sección (ver el Corolario 11.7.9) pero la empleamos en el siguiente ejemplo.

Ejemplo 11.6.15 *Consideremos el SRT $\mathcal{R}$ del Ejemplo 11.2.1. Sea $f =$* `indx` *y $\mathcal{C}^*_{[\mathsf{Nat}]} = \{\mathtt{0}, \mathtt{s}, \mathtt{[]}, \mathtt{::}\}$. Sea $\mu' = rmap^f(\mu^{com}_{\mathcal{R}})$ (ver el Ejemplo 11.6.12). Utilizando $\mathsf{V}(\mathcal{R})$ (ver el Ejemplo 11.6.10) y μ', obtenemos:*

```
T̄(indx(s(0)::[],dbls(from(0))))
    ↪ indx'(s(0)::[],dbls(from(0)))
    ↪ sel'(s(0),dbls(from(0))::'indx'([],dbls(from(0)))
    ↪ sel'(s(0),dbls(from(0))::'[]'
    ↪ sel'(s(0),dbls(0::from(s(0))))::'[]'                    (*)
    ↪ sel'(s(0),dbl(0)::dbls(from(s(0))))::'[]'               (**)
    ↪ sel'(0,dbls(from(s(0))))::'[]'
    ↪ sel'(0,dbls(s(0)::from(s(s(0)))))::'[]'                 (*)
    ↪ sel'(0,dbl(s(0))::dbls(from(s(s(0)))))::'[]'
    ↪ T̄(dbl(s(0)))::'[]'
    ↪ dbl'(s(0))::'[]'
    ↪ s'(s'(dbl'(0))::'[]'
    ↪ s'(s'(0'))::'[]'
```

Obsérvese que, en (*) y (**), no es posible iniciar ninguna derivación inútil o infinita. La derivación sensible al contexto se comporta como si fuera guiada fundamentalmente por la función de reemplazamiento $\mu_{\mathcal{R}}^{com}$ (Proposición 11.6.13). En particular, se preserva el carácter óptimo de las reducciones realizadas bajo $\mu_{\mathcal{R}}^{com}$.

11.7 Corrección y completitud de la transformación

Para establecer formalmente la correción y completitud de la transformación V, necesitamos algunos resultados previos. En primer lugar, demostramos la confluencia y terminación de la relación de reducción $\hookrightarrow_{T^f(\mu')}$.

Proposición 11.7.1 *Sea $\mathcal{R} = (\Sigma \cup \{\bar{\mathtt{T}}\}, R) = (\mathcal{C} \uplus \mathcal{F} \uplus \{\bar{\mathtt{T}}\}, R)$ un SRTE y $\mu \in M_{\Sigma \uplus \{\bar{\mathtt{T}}\}}$. Sea $\mu' = rmap^f(\mu)$. El SRT $T^f = (\Sigma^f \uplus \{\bar{\mathtt{T}}\}, T^f)$ es μ'-confluente y μ'-terminante.*

DEMOSTRACIÓN. Puesto que $\mu'(\bar{\mathtt{T}}) = \emptyset$, se tiene que $Var^{\mu'}(l) = \emptyset$ para cada regla $l \to r \in T^f$. Por tanto, el SRT T^f tiene variables μ'-reemplazantes homogéneas por la izquierda (recordemos la Observación 6.1.4). La μ'-confluencia es consecuencia de la ortogonalidad de las reglas de T^f y del Teorema 6.4.4. La terminación de T^f se demuestra con facilidad empleando un *rpo* e imponiendo sobre los símbolos de la signatura $\Sigma^f \uplus \{\bar{\mathtt{T}}\}$ la precedencia $\bar{\mathtt{T}} > c'$ y $\bar{\mathtt{T}} > g'$ para todo $c' \in \mathcal{C}'$ y $g' \in \mathcal{F}'$. Entonces, de acuerdo con el Teorema 5.1.4, T^f es μ'-terminante. □

Como consecuencia de esta proposición, todo término perteneciente al conjunto $\mathcal{T}(\Sigma^f \uplus \{\bar{\mathtt{T}}\}, V)$ tiene una forma μ'-normal única respecto a la relación de reducción $\hookrightarrow_{T^f(\mu')}$ inducida por el SRT $T^f = (\Sigma^f \uplus \{\bar{\mathtt{T}}\}, T^f)$ y la función de remplazamiento $\mu' = rmap^f(\mu)$ para cualquier μ. Asociaremos a cada término t su forma $\hookrightarrow_{T^f(\mu')}$-normal s escribiendo $t \hookrightarrow^{\omega}_{T^f(\mu')} s$. Emplearemos esta propiedad en la demostración de corrección de la transformación V.

Lema 11.7.2 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mu \in M_\Sigma$, $f \in \mathcal{F}$ y $\mu' = rmap^f(\mu)$. Si $t \in ev^f(t)$, entonces $\kappa^f(t)$ es una forma $\hookrightarrow_{T^f(\mu')}$-normal.*

DEMOSTRACIÓN. Las posibles ocurrencias de subtérminos $\bar{\mathtt{T}}(s)$ en $\kappa^f(t)$ deben cumplir lo siguiente: $s \in V$. Por tanto, no pueden aplicarse más pasos de $\hookrightarrow_{T^f(\mu')}$-reducción sobre el subtérmino $\bar{\mathtt{T}}(s)$. □

A continuación demostramos que el conjunto de reglas T^f proporciona una implementación de la función κ^f.

Lema 11.7.3 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mu \in M_\Sigma$ y $f \in \mathcal{F}$. Sea $\mu' = rmap^f(\mu)$ y $t \in \mathcal{T}(\Sigma, V)$. Entonces, $\bar{\mathtt{T}}(t) \hookrightarrow^{\omega}_{T^f(\mu')} t' \wedge t' = \kappa^f(t)$ si y sólo si $t \in ev^f(\Sigma, V)$.*

DEMOSTRACIÓN. Demostramos la parte *si* de la doble implicación por inducción sobre la estructura de $t \in ev^f(\Sigma, V)$. Para el caso base, consideremos que $t = x \in V$ o bien $t = g(\tilde{t})$ para algún símbolo $g \in \mathcal{F}^f_{\mathcal{R}}$. El primer caso es inmediato, puesto que $\bar{\mathtt{T}}(x) = \kappa^f(x)$ es una forma $\hookrightarrow_{T^f(\mu')}$-normal. En cuanto al segundo caso, tenemos una regla $\bar{\mathtt{T}}(g(x_1, \ldots, x_k)) \to g'(x_1, \ldots, x_k) \in T^f$, donde $k = ar(g)$. Aplicaremos esta regla a $\bar{\mathtt{T}}(g(\tilde{t}))$, obteniendo $t' = g'(\tilde{t}) = \kappa^f(g(\tilde{t}))$, que es una forma $\hookrightarrow_{T^f(\mu')}$-normal, lo cual completa la demostración.

Paso de inducción. Sea $t = c(t_1, \ldots, t_k)$, donde $c \in \mathcal{C}^*_{type(f)}$, $k = ar(c)$ y $t_1, \ldots, t_k \in ev^f(\Sigma, V)$. Puesto que $c \in \mathcal{C}^*_{type(f)}$, tenemos una regla $\bar{\mathtt{T}}(c(x_1, \ldots, x_k)) \to c'(\bar{\mathtt{T}}(x_1), \ldots, \bar{\mathtt{T}}(x_k)) \in T^f$, donde $k = ar(c)$. Al aplicar esta regla a $\bar{\mathtt{T}}(c(t_1, \ldots, t_k))$, obtenemos $c'(\bar{\mathtt{T}}(t_1), \ldots, \bar{\mathtt{T}}(t_k))$. Por la hipótesis de inducción, tenemos $\bar{\mathtt{T}}(t_i) \hookrightarrow^{\omega}_{T^f(\mu')} \kappa^f(t_i)$ para $1 \leq i \leq k$. Dado que $\mu'(c') = \mathbb{N}^+_{ar(c')}$, la Proposición 3.3.6 nos permite concluir que $\bar{\mathtt{T}}(t) \hookrightarrow^{\omega}_{T^f(\mu')} c'(\kappa^f(t_1), \ldots, \kappa^f(t_k))$. Dado que $\kappa^f(c(t_1, \ldots, t_k)) = c'(\kappa^f(t_1), \ldots, \kappa^f(t_k))$, obtenemos el resultado deseado.

La demostración de la parte *sólo si* se realiza por inducción sobre $t \in \mathcal{T}(\Sigma, V)$. Si $t = x \in V$, entonces $\bar{\mathtt{T}}(x)$ es una forma $\hookrightarrow_{T^f(\mu')}$-normal. Si $t = g \in \Sigma$, distinguimos dos casos:

1. $g \in \mathcal{F}^f_{\mathcal{R}} \cup \mathcal{C}'$; es decir, $t \in ev^f(\Sigma, V)$. Entonces, $\bar{\mathtt{T}}(g) \hookrightarrow_{T^f(\mu)} g'$, que es una forma $\hookrightarrow_{T^f(\mu)}$-normal. Por tanto, $\bar{\mathtt{T}}(g) \hookrightarrow^{\omega}_{T^f(\mu)} g' = \kappa^f(t)$, de donde se sigue la conclusión.

2. $g \notin \mathcal{F}^f_{\mathcal{R}} \cup \mathcal{C}'$; es decir, $t \notin ev^f(\Sigma, V)$. Entonces, $\bar{\mathtt{T}}(g)$ es una forma normal, pero $\kappa^f(g)$ no está definida.

Si $t = g(\tilde{t})$, distinguimos los mismos dos casos y, aplicando un razonamiento similar al anterior y empleando la hipótesis de inducción, concluimos la demostración. □

Con el fin de establecer la corrección y completitud de nuestra transformación, demostraremos en primer lugar que, dado un paso de μ-reducción $t \overset{u}{\hookrightarrow} s$ (siendo $t \in ev^f(\Sigma, V)$ tal que $root(t) \in \mathcal{F}^f_{\mathcal{R}}$) en el SRTE $\mathcal{R}$, es posible simular la transformación del término $\kappa^f(t)$ por el SRTE $\mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$ mediante:

- un solo paso de μ'-reescritura que emplee una regla de S^f seguido de (tal vez varios) pasos de μ'-reescritura que sólo empleen reglas de T^f (siempre que $u = \epsilon$) o bien

- efectuar exactamente el mismo paso de μ'-reescritura, utilizando la misma regla de R (cuando $u > \epsilon$).

De acuerdo con esto, definimos la relación $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')}$ como la composición de un paso de reducción que emplea las reglas que se encuentran en R o en S^f, seguido de la $\hookrightarrow_{T^f(\mu')}$-normalización del término obtenido.

Definición 11.7.4 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mu \in M_\Sigma$ y $f \in \mathcal{F}$. Sea $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$. La relación de reducción $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')}$ se define como sigue: $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} = \hookrightarrow_{R \cup S^f(\mu')} \circ \hookrightarrow^{\omega}_{T^f(\mu')}$.*

Adviértase que $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \subseteq \hookrightarrow_{\mathcal{R}'(\mu')}$. La conexión entre $\hookrightarrow_{\mathcal{R}'(\mu')}$ y $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')}$ que resulta de interés para nuestros fines viene expresada por la siguiente proposición.

Proposición 11.7.5 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $\mu \in M_\Sigma$. Sea $f \in \mathcal{F}$ y $t, s \in ev^f(\Sigma, V)$. Sea $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$. Entonces, $\kappa^f(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(s)$ si y sólo si $\kappa^f(t) \hookrightarrow\!\!\!\!\rightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(s)$.*

Demostración. La parte *si* es inmediata, puesto que $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \subseteq \hookrightarrow_{\mathcal{R}'(\mu')}$. Demostramos la parte *sólo si* por inducción sobre el número p de aplicaciones de las reglas en R o en S^f (a lo cual denominaremos un $R \cup S^f$-paso para abreviar) en la derivación:

$$\kappa^f(t) = t'_1 \hookrightarrow_{\mathcal{R}'(\mu')} t'_2 \hookrightarrow_{\mathcal{R}'(\mu')} \cdots \hookrightarrow_{\mathcal{R}'(\mu')} t'_n = \kappa^f(s)$$

Si $p = 1$ es inmediato, porque la derivación puede escribirse como sigue:

$$\kappa^f(t) = t'_1 \hookrightarrow_{R \cup S^f(\mu')} t'_2 \hookrightarrow_{T^f(\mu')} \cdots \hookrightarrow_{T^f(\mu')} t'_n = \kappa^f(s).$$

Esto se debe a que, siendo $t \in ev^f(\Sigma, V)$, el Lema 11.7.2 asegura que $\kappa^f(t)$ es una forma $\hookrightarrow_{T^f(\mu')}$-normal y el único $R \cup S^f$-paso debe realizarse en primer lugar. Por tanto, $t'_2 \hookrightarrow^{\omega}_{T^f(\mu')} \kappa^f(s)$ y $\kappa^f(t) \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(s)$.

Si $p > 1$, el primer paso de reducción $t'_1 \hookrightarrow_{\mathcal{R}'(\mu')} t'_2$ es, de nuevo, un $R \cup S^f$-paso. Distinguimos dos casos:

1. Si $t'_1 \overset{u}{\hookrightarrow}_{R(\mu')} t'_2$, existe un símbolo definido situado sobre una ocurrencia $v < u$ de t. Por tanto, después de efectuar la reducción, el término t'_2 todavía es imagen por κ^f de algún $t_2 \in ev^f(\Sigma, V)$; es decir, $t'_2 = \kappa^f(t_2)$. Por la hipótesis de inducción concluimos lo deseado puesto que, siendo t'_2 una forma $\hookrightarrow_{T^f(\mu')}$-normal (de acuerdo con el Lema 11.7.2), podemos escribir $t'_1 \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} t'_2$.

2. Si $t'_1 \overset{u}{\hookrightarrow}_{S^f(\mu')} t'_2$, se aplica una regla $g'(\tilde{l}) \to \bar{\mathtt{T}}(r) \in S^f$ a un subtérmino $t'_1|_u = \kappa^f(t_1)|_u = \sigma(g'(\tilde{l}))$ cuyo símbolo a la raíz es un representante g' de un símbolo definido externo g. Por ello, $g(\tilde{t}) = t_1|_u \in ev^f(\Sigma, V)$ también es un redex de la regla $g(\tilde{l}) \to r \in R$. Por la Proposición 11.5.4, tenemos que $\sigma(r) \in ev^f(\Sigma, V)$. Por tanto, de acuerdo con el Lema 11.7.3, $\bar{\mathtt{T}}(\sigma(r)) \hookrightarrow^{\omega}_{T^f(\mu)} \kappa^f(\sigma(r))$. Sea q la longitud de esta secuencia de $\hookrightarrow_{T^f(\mu')}$-normalización que se encuentra inmersa en la derivación que arranca en t'_2. Asumamos que el siguiente $R \cup S^f$-paso se realiza sobre el término t'_i para $1 < i \leq n$. Puesto que t'_i es el siguiente término sobre el que se lleva a cabo un $R \cup S^f$-paso, las reducciones sobre los términos t'_j para $2 \leq j < i$ deben pertenecer a la secuencia de $\hookrightarrow_{T^f(\mu')}$-normalización de

$t'_2|_u = \bar{\mathtt{T}}(\sigma(r))$. Si $i-2 = q$, la $\hookrightarrow_{T^f(\mu')}$-normalización concluye inmediatamente antes de que el $R \cup S^f$-paso tenga lugar y esto significa que $t'_i = \kappa^f(t_i)$ para algún $t_i \in ev^f(\Sigma, V)$. De esta manera, por la hipótesis de inducción, concluiríamos la demostración. Si $i - 2 < q$, puesto que $\mu(\bar{\mathtt{T}}) = \emptyset$, el $R \cup S^f$-paso debe efectuarse sobre una ocurrencia $u' \in O(t_i)$ tal que, si $t_j \overset{u_j}{\hookrightarrow}_{T^f(\mu')} t_{j+1}$ y $j > i$, entonces $u' \parallel u_j$. Podemos situar las $q - i + 2 \hookrightarrow_{T^f(\mu')}$-reducciones que faltan para completar la $\hookrightarrow_{T^f(\mu')}$-normalización de t'_2 delante de t_i. De esta manera, podemos aplicar la hipótesis de inducción y concluir la demostración.

□

Existe una conexión simple entre la $\hookrightarrow_{\mathcal{R}(\mu)}$-reducción de términos pertenecientes a $ev^f(\Sigma, V)$ y la $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')}$-reducción de sus imágenes por κ^f empleando $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$.

Proposición 11.7.6 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $\mu \in M_\Sigma$. Sea $f \in \mathcal{F}$ y $t, s \in ev^f(\Sigma, V)$. Sea $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$. Si $t \hookrightarrow^*_{\mathcal{R}(\mu)} s$, entonces $\kappa^f(t) \hookrightarrow\!\!\!\!\rightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(s)$.*

DEMOSTRACIÓN. Por inducción en la longitud n de la derivación $t \hookrightarrow^*_{\mathcal{R}(\mu)} s$. Si $n = 0$ es inmediato. Si $n > 0$, escribimos

$$t \hookrightarrow^*_{\mathcal{R}(\mu)} t' \overset{u}{\hookrightarrow}_{\mathcal{R}(\mu)} s$$

Por el Teorema 11.5.5, $t' \in ev^f(\Sigma, V)$. Entonces, por la hipótesis de inducción, $\kappa^f(t) \hookrightarrow\!\!\!\!\rightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(t')$. Demostramos que $\kappa^f(t') \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(s)$ por inducción sobre la estructura de t' en $ev^f(\Sigma, V)$. El caso $t' \in V$ no puede darse porque, de ser así, el paso de reducción $t' \overset{u}{\hookrightarrow}_{\mathcal{R}(\mu)} s$ no sería posible. Si $t' = g(\tilde{t})$ y $g \in \mathcal{F}^f_{\mathcal{R}}$, distinguimos dos casos, de acuerdo con la ocurrencia $u \in O_{\mathcal{R}}(t')$ del redex considerado para efectuar el paso de reducción.

1. $u = \epsilon$. Entonces $t' = g(\tilde{t}) = \sigma(g(\tilde{l}))$ es un redex de alguna regla $g(\tilde{l}) \to r \in R$. Puesto que $g \in \mathcal{F}^f_{\mathcal{R}}$, la regla $g'(\tilde{l}) \to \bar{\mathtt{T}}(r)$ se encuentra en el componente S^f de $\mathcal{R}'$. Por tanto, $\kappa^f(g(\tilde{t})) = g'(\tilde{t})$ es un redex $\sigma(g'(\tilde{l}))$ de esta regla y se reduce a $\sigma(\bar{\mathtt{T}}(r)) = \bar{\mathtt{T}}(\sigma(r))$; es decir, $\kappa^f(t') \hookrightarrow_{R \cup S^f(\mu')} \bar{\mathtt{T}}(\sigma(r))$. Puesto que $\sigma(r) = s \in ev^f(\Sigma, V)$, por el Lema 11.7.3, obtenemos $\bar{\mathtt{T}}(\sigma(r)) \hookrightarrow^*_{T^f(\mu')} \kappa^f(\sigma(r))$, es decir, $\bar{\mathtt{T}}(\sigma(r)) \hookrightarrow^\omega_{T^f(\mu')} \kappa^f(\sigma(r)) = \kappa^f(s)$ y esto permite concluir el resultado deseado.

2. $u > \epsilon$. En este caso debe tenerse $s = g(\tilde{s})$; es decir, los símbolos que se encuentran en las raíces de t' y s coinciden. Por tanto, $\kappa^f(g(\tilde{t})) = g'(\tilde{t})$ y $\kappa^f(g(\tilde{s})) = g'(\tilde{s})$. Dado que la regla $l \to r \in R$ se aplica a un subtérmino t_i de t', que el conjunto de reglas R es un componente de $\mathcal{R}'$, y que $\mu'(g') = \mu(g)$,

concluimos que $g'(\tilde{t}) \hookrightarrow_{R \cup S^f(\mu')} g'(\tilde{s})$. Puesto que $g'(\tilde{s}) = \kappa^f(g(\tilde{s}))$ es una forma $\hookrightarrow_{T^f(\mu')}$-normal, tenemos que $g'(\tilde{t}) \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} g'(\tilde{s})$, lo cual permite concluir el resultado deseado.

Si $t' = c(\tilde{t})$, con $c \in \mathcal{C}^*_{type(f)}$, debe tenerse $s = c(\tilde{s})$. Por tanto, $\kappa^f(c(\tilde{t})) = c'(\kappa^f(\tilde{t}))$ y $\kappa^f(c(\tilde{s})) = c'(\kappa^f(\tilde{s}))$ y siendo $u = i.u'$, tenemos $t_i \overset{u'}{\hookrightarrow}_{\mathcal{R}(\mu)} s_i$. Dado que $t_i \in ev^f(\Sigma, V)$, por la hipótesis de inducción, $\kappa^f(t_i) \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(s_i)$ y, de acuerdo con la definición de κ^f y puesto que $\kappa^f(t_j) = \kappa^f(s_j)$ para $1 \leq j \leq ar(c)$, $i \neq j$, concluimos la demostración, tras considerar que $\mu'(c') = \mathbb{N}^+_{ar(c')}$ y utilizar la Proposición 3.3.6. □

Es posible establecer la validez de la sentencia opuesta a la expresada en la Proposición 11.7.6 si consideramos $\hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')}$-derivaciones que constan únicamente (salvo tal vez el último término de la secuencia) de términos cuyo símbolo a la raíz es un símbolo definido.

Proposición 11.7.7 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mu \in M_\Sigma$ y $f \in \mathcal{F}$. Sean $t, s \in ev^f(\Sigma, V)$ y sea t' tal que $root(t') \in \mathcal{F}$. Sea $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$. Entonces, $t \hookrightarrow^*_{\mathcal{R}(\mu)} t' \hookrightarrow_{\mathcal{R}(\mu)} s$ si y sólo si $\kappa^f(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(t') \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(s)$.*

Demostración. La parte *sólo si* es inmediata empleando la Proposición 11.7.6. Respecto a la parte *si*, realizamos la demostración por inducción sobre la longitud $n \geq 1$ de la derivación $\kappa^f(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(t') \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(s)$. Si $n = 1$, entonces $\kappa^f(t) = \kappa^f(t')$ y, por inyectividad de κ^f, $t = t'$. Por el Corolario 11.6.14, $O^{\mu'}_{\mathsf{V}(\mathcal{R})}(\kappa^f(t)) = O^{\mu}_{\mathcal{R}}(t)$. Sea $\kappa^f(t)|_u$ el redex considerado para la reducción. Tenemos que $\kappa^f(t)|_u$ es un redex de una regla perteneciente a R o a S^f. Distinguiremos estos dos casos.

1. Si $\kappa^f(t)|_u$ es un R-redex, la misma regla puede aplicarse al subtérmino de t a la ocurrencia $u \in O^\mu(t)$. Por tanto, $t \hookrightarrow_{\mathcal{R}(\mu)} s$.

2. Si $\kappa^f(t)|_u$ es un S^f-redex de una regla $g'(\tilde{l}) \to \bar{\mathtt{T}}(r) \in S^f$, entonces, dado que $root(t') = root(t) \in \mathcal{F}$, debe tenerse $u = \epsilon$ y, puesto que $\kappa^f(t) \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(s)$, por definición de S^f y, por el Lema 11.7.3, obtenemos el resultado deseado.

Si $n > 1$, escribimos la derivación como sigue:

$$\kappa^f(t) \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(t'') \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(t') \hookrightarrow\!\!\!\!\rightarrow_{\mathcal{R}'(\mu')} \kappa^f(s).$$

Tenemos las mismas opciones fundamentales del análisis realizado para el caso base. Por tanto, concluimos que $t \hookrightarrow_{\mathcal{R}(\mu)} t''$. Por la Proposición 11.5.4, $t'' \in ev^f(\Sigma, V)$ y, aplicando la hipótesis de inducción, completamos la demostración. □

Ahora estamos preparados para establecer el resultado principal de esta sección.

Teorema 11.7.8 (Corrección y completitud de V) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE lineal por la izquierda, $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ y $f \in \mathcal{F}$. Sea $t \in ev^f(\Sigma, V)$ y $\delta \in \mathcal{T}(\mathcal{C}, V)$. Sea $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$. Entonces, $t \to^*_{\mathcal{R}} \delta$ si y sólo si $\kappa^f(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(\delta)$.*

DEMOSTRACIÓN. Por inducción sobre la estructura de δ. Si $\delta \in \mathcal{C} \cup V$, por el Teorema 11.1.2, y puesto que $\hookrightarrow \subseteq \to$, tenemos que $t \to^* \delta$ si y sólo si $t \hookrightarrow^*_{\mathcal{R}(\mu)} \delta$. Si $t = \delta$, el resultado es inmediato. Si $t \neq \delta$, debe tenerse $root(t) \in \mathcal{F}$. En ese caso, por la Proposición 11.7.7 y la Proposición 11.7.5, concluimos el resultado deseado.

Si $\delta = c(\tilde{\delta})$, por el Teorema 11.1.2, tenemos que $t \hookrightarrow^*_{\mathcal{R}(\mu)} c(\tilde{s})$ y $c(\tilde{s}) \overset{>\epsilon}{\longrightarrow}{}^* c(\tilde{\delta})$. Por tanto, $s_i \to^* \delta_i$, para $1 \leq i \leq ar(c)$ y, por la hipótesis de inducción, $\kappa^f(s_i) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(\delta_i)$. De nuevo distinguimos dos casos: si $root(t) \in \mathcal{C}$, puesto que $t \to^* \delta$ y $t \in ev^f(\Sigma, V)$, por el Teorema 11.5.5, debe ser $\delta \in ev^f(\Sigma, V)$, luego debe tenerse $root(t) = c \in \mathcal{C}^*_{type(f)}$ y, por definición de κ^f, tenemos que $\kappa^f(t) = \kappa^f(c(\tilde{t}) = c'(\kappa(\tilde{t}))$ y, dado que $\mu'(c') = \mathbb{N}^+_{ar(c)}$, por la Proposición 3.3.6, $\kappa^f(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(\delta)$. Si $root(t) \notin \mathcal{C}$, debe ser $root(t) \in \mathcal{F}$. Sin pérdida de generalidad, podemos asumir que $c(\tilde{s})$ es el primer término de la derivación $t \hookrightarrow^*_{\mathcal{R}(\mu)} c(\tilde{s})$ cuyo símbolo a la raíz es un constructor. Por tanto, todos los términos que le preceden en la derivación tienen un símbolo definido a la raíz. De esta manera, por la Proposición 11.7.7 y la Proposición 11.7.5, $\kappa^f(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(c(\tilde{s}))$; es decir, puesto que $c \in \mathcal{C}^*_{type(f)}$, $\kappa^f(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} c'(\kappa^f(\tilde{s}))$. Por tanto, debido a que $\mu'(c') = \mathbb{N}^+_{ar(c')}$, y haciendo uso de la Proposición 3.3.6, concluimos la demostración. □

Para evaluar un término $t \in ev^f(\Sigma, V)$, sólo necesitamos encerrarlo bajo un símbolo $\bar{\mathtt{T}}$ y evaluar $\bar{\mathtt{T}}(t)$ mediante el SRTE transformado.

Corolario 11.7.9 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE lineal por la izquierda, $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, y $f \in \mathcal{F}$. Sea $t \in ev^f(\Sigma, V)$ y $\delta \in \mathcal{T}(\mathcal{C}, V)$. Sea $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$. Entonces, $t \to^*_{\mathcal{R}} \delta$ si y sólo si $\bar{\mathtt{T}}(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \kappa^f(\delta)$.*

DEMOSTRACIÓN. Inmediata, empleando el Lema 11.7.3 y el Teorema 11.7.8. □

Observación 11.7.10 *Si evaluamos mediante $\mathsf{V}(\mathcal{R})$ un término t cuyo valor δ es un término constructor no básico, obtendremos un término $\kappa^f(\delta)$ con ocurrencias de subtérminos $\bar{\mathtt{T}}(x)$, donde los símbolos $\bar{\mathtt{T}}$ encierran variables $x \in V$. Si reemplazamos estos subtérminos $\bar{\mathtt{T}}(x)$ por la variable x que éstos encierran, obtendremos un término $\tau^f(\delta)$ donde ya no aparecen los símbolos auxiliares $\bar{\mathtt{T}}$.*

El siguiente resultado establece que, al utilizar la transformación V para evaluar términos básicos (lo habitual en programación funcional), no obtendremos símbolos espúreos $\bar{\mathtt{T}}$ en el resultado final (ver el Ejemplo 11.6.15).

Corolario 11.7.11 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE lineal por la izquierda, $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ y $f \in \mathcal{F}$. Sea $t \in gev^f(\Sigma, V)$ y $\delta \in \mathcal{T}(\mathcal{C})$. Sea $\mathcal{R}' = \mathsf{V}(\mathcal{R})$ y $\mu' = rmap^f(\mu)$. Entonces, $t \to^*_{\mathcal{R}} \delta$ si y sólo si $\bar{\mathtt{T}}(t) \hookrightarrow^*_{\mathcal{R}'(\mu')} \tau^f(\delta)$.*

DEMOSTRACIÓN. Inmediata, empleando el Corolario 11.7.9 y el Lema 11.6.6. □

En particular, las expresiones iniciales evaluadas por los programas funcionales son términos básicos. Por tanto, el Corolario 11.7.11 asegura que su evaluación obtiene un término constructor básico (construido a partir de los alias $c' \in \mathcal{C}'$ correspondientes a los constructores $c \in \mathcal{C}$), siempre que esto sea posible.

Observación 11.7.12 *El Teorema 11.7.8 o el Corolario 11.7.9 son la base para el uso de la transformación V. Establecen que el valor (básico o no) de cualquier término $t = f(\tilde{t})$ respecto a un SRTE $\mathcal{R}$ puede ser computado mediante el SRTE transformado $\mathsf{V}(\mathcal{R})$. Si trabajamos con SRTEs* confluentes*, el Teorema 11.7.8 también significa que podemos realizar dicho cómputo empleando cualquier estrategia de μ'-reescritura de la cual podamos asegurar que producirá una secuencia de reducción finita. No necesitamos asegurar la μ'-confluencia de $\mathsf{V}(\mathcal{R})$.*

El Lema 11.7.3 sugiere una mejora interesante de la transformación V: realizar el *desplegado* de las partes derechas de las reglas de reescritura respecto a las reglas del componente T^f del SRTE transformado.

Definición 11.7.13 (Transformación desplegada $\overline{\mathsf{V}}$) *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE y $f \in \mathcal{F}$. Entonces $\overline{\mathsf{V}}^f(\mathcal{R}) = (\Sigma^f \uplus \{\bar{\mathtt{T}}\}, R \cup \bar{S}^f \cup T^f)$. El conjunto de reglas $\bar{S}^f$ viene dado como sigue ($\bar{\mathtt{T}}$ y el conjunto de reglas T^f mantienen el mismo significado asignado por la Definición 11.6.8):*

$$\bar{S}^f = \{g'(\tilde{l}) \to \kappa^f(r) \mid g(\tilde{l}) \to r \in R \wedge g \in \mathcal{F}^f_{\mathcal{R}}\}$$

De hecho, en los ejemplos de aplicación de transformación que se presentaron anteriormente (Ejemplos 11.6.9 y 11.6.10) se ha empleado la transformación $\overline{\mathsf{V}}$ en vez de V.

11.8 Trabajos relacionados

En [KW95] se ha presentado una transformación para obtener evaluaciones perezosas mediante la utilización de estrategias de reducción *voraces*. Esta transformación es más compleja que la transformación V presentada aquí. En el peor caso para esa transformación, añade $3 + n \cdot r + n \cdot 2^l + 2 \cdot s$ reglas al SRT original $\mathcal{R} = (\Sigma, R)$ (ver [KW95]), donde n es el número de reglas en R, s es el número de símbolos de la signatura Σ y r es el número máximo de ocurrencias no reemplazantes de subtérminos

no variables en las partes izquierdas de las reglas en R. La transformación V sólo introduce $n + s$ reglas nuevas en el peor de los casos. Esto se debe a que la ejecución del SRT transformado se lleva a cabo empleando la *rsc* en lugar de la reescritura sin restricciones como se hace en [KW95]. No obstante, una comparación más profunda entre ambas propuestas debería tener en cuenta también la complejidad temporal del cómputo de cada una de las transformaciones a partir del SRT original.

En estos momentos se está desarrollando una implementación de un sistema de cómputo con SRTEs basado en la *rsc* y en el empleo de la trasformación V como base para la evaluación de expresiones respecto a los programas especificados [Gar98]

Capítulo 12

Formas normales y estabilización

En este capítulo discutimos el uso de la *rsc* como herramienta para estabilizar un término o para obtener su forma normal. Este problema es importante puesto que la descripción de los cómputos por reescritura viene dada, por lo general, como "obtención de formas normales" de las expresiones iniciales [Der93, Jou94, KOV96, O'D95]. El cómputo de formas estables es de interés durante las etapas intermedias del cómputo de las formas normales, sobre todo en las implementaciones perezosas de la reescritura.

En la Sección 12.1 demostramos que toda forma estable de un término puede ser aproximada mediante la *rsc*. En algunos casos especiales, es posible obtener formas normales directamente mediante la *rsc*.

En la Sección 12.2 discutimos cómo emplear las formas μ-normales para obtener formas normales.

En la Sección 12.3 establecemos condiciones para asegurar que, para toda forma normal de un término, existe una forma μ-normal cuyo contexto reemplazante máximo es un prefijo de la forma normal en cuestión. Esto demuestra que es posible emplear μ-estrategias de μ-normalización en tareas de normalización.

De acuerdo con lo anterior, en la Sección 12.4 definimos un procedimiento efectivo de μ-normalización basado en la *rsc* y demostramos su corrección y completitud.

12.1 Estabilización y normalización directa

El Teorema 11.1.2 establece condiciones que aseguran la completitud de la *rsc* en las tareas de cómputo de formas estables constructoras.

Para formas estables más generales, debemos limitar la clase de SRTEs considerados a los SRTs cuasi ortogonales.

Teorema 12.1.1 (Cómputo de formas estables) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT cuasi ortogonal y sea $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma, V)$ y sea $s = f(\tilde{s})$ un término estable. Si $t \to^* s$, entonces existe un término estable $s' = f(\tilde{s'})$ tal que $t \hookrightarrow^* s'$ y $s' \overset{>\epsilon}{\longrightarrow}{}^* s$.*

DEMOSTRACIÓN. De acuerdo con el Teorema 4.2.8, existe un término s' tal que $t \hookrightarrow^* s'$ y $s' \overset{>\epsilon}{\longrightarrow}{}^* s$. Por el Lema 2.3.5, s' es estable. □

Conjeturamos que este resultado puede extenderse a SRTs lineales por la izquierda sin requerir que el SRT sea cuasi ortogonal.

Observación 12.1.2 *Intentar extender el Teorema 12.1.1 a SRTs μ-ortogonales (Definición 6.3.3) no supone ningún beneficio adicional. Esto se debe a que si consideramos una función de reemplazamiento μ tal que $\mu_\mathcal{R}^{com} \sqsubseteq \mu$, un SRT μ-ortogonal siempre es ortogonal. Esto es así porque la μ-compatibilidad de las partes izquierdas de las reglas del SRT implica que todas las ocurrencias $u \in O_\Sigma(l)$ (las ocurrencias que pueden dar lugar a pares críticos) son reemplazantes. Por tanto, los pares críticos y los pares μ-críticos son exactamente los mismos.*

En general, no es posible asegurar (de forma no trivial) que los cómputos por *rsc* van a ser capaces de obtener *formas normales* generales.

Ejemplo 12.1.3 *Consideremos el SRT $\mathcal{R}$:*

```
f(x)   → g(x,x)      h(0)    → 0
g(0,x) → 0           h(s(0)) → 0
```

Tomemos $\mu(\mathtt{f}) = \mu(\mathtt{g}) = \mu(\mathtt{h}) = \mu(\mathtt{s}) = \{1\}$. Tanto $\mathcal{R}$ como μ verifican las condiciones del Teorema 11.1.4 (tomando, por ejemplo, $\mathcal{B} = \mathcal{C} = \{\mathtt{0}, \mathtt{s}\}$, lo cual cubre el peor de los casos para $\mathcal{B}$). Tenemos la siguiente derivación:

```
g(s(0),h(0)) → g(s(0),0)
```

que lleva a una forma normal `g(s(0),0)` *que no puede obtenerse mediante la* rsc *porque* `g(s(0),h(0))` *ya es una forma μ-normal. Observemos que* `g(s(0),0)` $\notin \mathcal{T}(\mathcal{C}, V)$.

Sin embargo, tenemos algunas oportunidades en ciertos casos especiales.

Teorema 12.1.4 (Cómputo de formas normales) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda, $t \in \mathcal{T}(\Sigma, V)$ y $s \in \mathcal{T}(\Sigma, \emptyset)$ tal que s es una forma normal. Sea $\mu \in M_\Sigma$ tal que $\mu_\mathcal{R}^{com} \sqcup \mu_s \sqsubseteq \mu$. Entonces, $t \to^* s$ si y sólo si $t \hookrightarrow^* s$.*

DEMOSTRACIÓN. La parte *si* es inmediata, puesto que $\hookrightarrow \subseteq \rightarrow$. Respecto a la parte *sólo si*, puesto que $t \rightarrow^* s$ y s es básico, podemos escribir $t \rightarrow^* \sigma(s)$ para cualquier sustitución σ. Por la Proposición 4.2.7, existe una sustitución θ tal que $t \hookrightarrow^* \theta(s)$. Puesto que s es básico, $\theta(s) = s$. □

Observación 12.1.5 *El Teorema 12.1.4 es difícil de emplear en la práctica porque, dada una expresión inicial t, normalmente desconocemos la estructura de la forma normal s de t. Como consecuencia, no hay una forma sencilla de establecer si s será un término básico (aunque si t es básico y nos limitamos a SRTs, s también lo será) o cuál será la definición de μ_s. Si s es básico, entonces $\mu_s(f) = \mathbb{N}^+_{ar(f)}$ para todo $f \in \Sigma(s)$ con lo que bastaría conocer el conjunto de símbolos de que consta s. Dicho conjunto de símbolos podría ser aproximado empleando información sobre tipos, de forma similar a como se realizó en el Capítulo 11 con los símbolos definidos externos. También puede tenerse en cuenta información acerca de qué funciones están* completamente definidas *en el SRTE. Si una función f está completamente definida por las reglas del SRTE, entonces no puede haber ningún término $f(\tilde{t})$ en forma normal y, puesto que s es una forma normal, puede excluirse el símbolo f del conjunto Δ que aproxima $\Sigma(s)$ (es decir, $\Sigma(s) \subseteq \Delta$).*

No obstante lo dicho, encontraremos una aplicación interesante del Teorema 12.1.4 al discutir la extensión de la *rsc* a SRTs condicionales (Capítulo 14).

Es interesante advertir que la función de reemplazamiento canónica $\mu^{com}_{\mathcal{R}}$ impone un límite a las mejoras en eficiencia, terminación, etc., que podemos esperar obtener al emplear la *rsc* en lugar de la reescritura sin restricción alguna. Si $\mu^{com}_{\mathcal{R}}$ coincide con $\mu_\top$, no podemos introducir ninguna mejora de comportamiento mediante el empleo de la *rsc*, al menos si queremos asegurar unas mínimas prestaciones por lo que respecta a la completitud de las computaciones sensibles al contexto.

Ejemplo 12.1.6 *Consideremos el SRT $\mathcal{R}$ que implementa la función* or paralela *[O'D85]:*

```
or(true,x) → true      or(false,false) → false
or(x,true) → true
```

Tenemos que $\mu^{com}_{\mathcal{R}} = \mu_\top$. Por tanto, la rsc *y la reescritura estándar son una misma cosa si deseamos asegurar alguna clase de completitud en los cómputos. Podemos comparar esta situación con la versión secuencial de este operador empleada en el lenguaje de programación* Lisp *(Ejemplo 3.4.1). Allí teníamos $\mu^{com}_{\mathcal{R}} \sqsubset \mu_\top$, lo cual llevaba a concluir que el empleo de la* rsc *en vez de la reescritura estándar puede ser ventajoso.*

12.2 El problema de la normalización

Si aceptamos introducir algún control adicional más allá del establecido por la función de reemplazamiento, es posible definir *algoritmos* basados en la *rsc* capaces de obtener la forma normal de un término.

Un método de normalización bien conocido es el que normaliza un término t en dos fases [Ken94, Mid97, O'D95]:

1. Estabilizar t, obteniendo una forma estable $f(\tilde{s})$.

2. Aplicar, de forma recursiva, este procedimiento a $s_1, \ldots, s_{ar(f)}$.

Este método es seguro porque, por definición de forma estable, posteriores modificaciones de los subtérminos situados bajo el símbolo que ocupa la raíz de la forma estable no suponen que haya que considerar en el futuro un nuevo paso de reducción a la raíz puesto que no es posible reducir una forma estable a un redex. Este método adolece del siguiente inconveniente: no es decidible si un término es estable o no.

Dado que las formas μ-normales son formas (fuertemente) estables (Teoremas 4.3.7 y 4.3.8), podemos ensayar la utilización de formas μ-normales en etapas intermedias del cómputo de formas normales.

Si lo hacemos así, el procedimiento simple esbozado anteriormente puede ser modificado para evitar reinspeccionar innecesariamente los términos. Podemos tomar el *contexto reemplazante máximo* (Definición 3.2.5) de la forma μ-normal intermedia obtenida como prefijo estable bajo el cual aplicar recursivamente el procedimiento de normalización (en lugar de emplear el símbolo que etiqueta la raíz de la forma μ-normal). Las secciones siguientes establecen los resultados necesarios para justificar la viabilidad de este método.

Observación 12.2.1 *Las implementaciones que obtienen de forma explícita formas estables en algún momento del proceso de normalización de un término suelen emplear formas fuertemente estables como aproximación decidible a éstas [O'D95, Str89]. Puesto que, en este caso, el procedimiento de decisión supone emplear la Ω-reducción para decidir si un término es fuertemente estable o no, tiene sentido utilizar formas μ-normales como aproximación a las formas estables, puesto que decidir si un término es una forma μ-normal o no es menos costoso que decidir si es fuertemente estable. Como contrapartida, por lo general, una forma μ-normal estará "más evaluada" que una forma fuertemente estable. No obstante, esto puede ser ventajoso si deseamos intercalar cómputos (parcialmente) voraces en el proceso de evaluación global.*

12.3 Formas normales y formas μ-normales

A continuación estudiamos la relación existente entre las formas normales y las formas μ-normales. Demostramos que, cuando un término tiene una forma normal (no necesariamente única), siempre tiene una (o varias) forma(s) μ-normal(es) que pueden verse como *aproximaciones* suyas.

En primer lugar, veamos que el análisis de este problema no es trivial. En general, hay términos que tienen forma normal y no tienen forma μ-normal y viceversa.

Ejemplo 12.3.1 *Consideremos el SRT no terminante $\mathcal{R}$:*

$$\begin{array}{ll} \mathtt{f(x,a)} & \to \mathtt{a} \\ \mathtt{g(a)} & \to \mathtt{a} \\ \mathtt{b} & \to \mathtt{b} \end{array}$$

Si imponemos $\mu(\mathtt{f}) = \{1\}$ y $\mu(\mathtt{g}) = \emptyset$, entonces:

1. *El término $t = \mathtt{f(b,g(a))}$ es normalizable:*

   $$\mathtt{f(b,}\underline{\mathtt{g(a)}}\mathtt{)} \to \underline{\mathtt{f(b,a)}} \to \mathtt{a}$$

   Sin embargo, t no tiene forma μ-normal pues sólo existe una secuencia de μ-reescritura que parte de él y es infinita:

   $$\mathtt{f(}\underline{\mathtt{b}}\mathtt{,g(a))} \hookrightarrow \mathtt{f(}\underline{\mathtt{b}}\mathtt{,g(a))} \hookrightarrow \cdots$$

2. *El término $t = \mathtt{g(b)}$ es una forma μ-normal, pero t no tiene forma normal:*

   $$\mathtt{g(}\underline{\mathtt{b}}\mathtt{)} \to \mathtt{g(}\underline{\mathtt{b}}\mathtt{)} \to \cdots$$

Ahora establecemos las condiciones que aseguran que cualquier forma normal $\bar{t}$ tiene una forma μ-normal correspondiente cuyo contexto reemplazante máximo es un prefijo de $\bar{t}$.

Teorema 12.3.2 (Formas normales y formas μ-normales) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma, V)$ y $\bar{t}$ es una forma normal de t, existe una forma μ-normal $\bar{s}$ de t tal que, siendo $C[\,] = CMR^{\mu}(\bar{s})$, se verifica $C[\widetilde{\Omega}] \leq \bar{t}$.*

DEMOSTRACIÓN. Por el Teorema 4.2.8, existe un término s tal que $t \hookrightarrow^* s \overset{>\epsilon}{\longrightarrow}{}^* \bar{t}$ y $root(s) = root(\bar{t})$. Demostramos la existencia de $\bar{s}$ por inducción sobre la estructura de $\bar{t}$. Si $\bar{t}$ es una constante o una variable, entonces, puesto que $\bar{t}$ es un símbolo de aridad nula, $\bar{s} = s = \bar{t}$ es la forma μ-normal deseada.

Sea $\bar{t} = f(\bar{t}_1, \ldots, \bar{t}_k)$ y consideremos $s = f(s_1, \ldots, s_k)$. Tenemos que $s_i \to^* \bar{t}_i$, para $1 \leq i \leq k = ar(f)$. Dado que cada s_i tiene a $\bar{t}_i$ como forma normal, aplicamos

la hipótesis de inducción para concluir que existen formas μ-normales $\overline{s'}_i$ de cada s_i tales que cada $C'_i[\,] = CMR^{\mu}(\overline{s'}_i)$ satisface $C'_i[\widetilde{\Omega}] \leq \bar{t}_i$. Adviértase que, por definición de $\leq$, el contexto $C[\,] = f(C_1[\,], \ldots, C_k[\,])$ definido al tomar

$$C_i[\,] = \begin{cases} C'_i[\,] & \text{si } i \in \mu(f) \\ \Box & \text{si } i \notin \mu(f) \end{cases}$$

para $1 \leq i \leq k$, satisface $C[\widetilde{\Omega}] \leq \bar{t}$. Obsérvese que $C[\,]$ es el contexto reemplazante máximo de $\bar{s} = f(\bar{s}_1, \ldots, \bar{s}_{ar(f)})$, que a su vez se define tomando

$$\bar{s}_i = \begin{cases} \overline{s'}_i & \text{si } i \in \mu(f) \\ s_i & \text{si } i \notin \mu(f) \end{cases}$$

para $1 \leq i \leq k$. Todavía debemos demostrar que $\bar{s}$ es una forma μ-normal de t. A partir de la definición de $\bar{s}$ y empleando la Proposición 3.3.6, es claro que $t \hookrightarrow^* s$. Demostramos que $\bar{s}$ es una forma μ-normal por contradicción. Si no lo es, puesto que cada $\bar{s}_i$ para $i \in \mu(f)$ es una forma μ-normal, $\bar{s}$ debe ser un redex de una regla de reescritura $l \to r \in R$. Esto significa que $\bar{s} = \sigma(l)$ para alguna sustitución σ. Sin embargo, puesto que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$, los subtérminos a las ocurrencias no reemplazantes de l son variables, y dado que $\bar{s}$ y $\bar{t}$ coinciden en el prefijo reemplazante $C[\,]$, se sigue que $\bar{t} = \sigma'(l)$ para alguna sustitución σ'. Por tanto, $\bar{t}$ no es una forma normal, una contradicción. □

El Teorema 12.3.2 tiene interés porque asegura que siempre que un término tiene una forma normal, también tiene una forma μ-normal que 'aproxima' la forma normal. Esto tiene algunas consecuencias inmediatas.

Teorema 12.3.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Si $t \in \mathcal{T}(\Sigma, V)$ es normalizable, entonces t es μ-normalizable.*

DEMOSTRACIÓN. Consecuencia del Teorema 12.3.2. □

Teorema 12.3.4 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Si $\mathcal{R}$ es normalizante, entonces $\mathcal{R}$ es μ-normalizante.*

Observación 12.3.5 *Los Teoremas 12.3.3 y 12.3.4 aseguran que, empleando una estrategia μ-normalizante, los cómputos mediante la* rsc *pueden obtener una forma μ-normal, siempre que se realicen sobre un término* normalizable*, es decir, un término que tiene forma normal. Esto nos permite utilizar las estrategias μ-normalizantes para implementar la normalización de términos si procedemos de forma escalonada, obteniendo prefijos estables (ante reducciones posteriores) de las formas normales buscadas y saltando al interior de dichos prefijos para extenderlos hasta obtener finalmente la forma normal. Formalizaremos este procedimiento en la próxima sección.*

```
Procedure norm(t)
   t := head-eval(t);
   let t = f(t_1, ..., t_k);
   for i := 1, ..., k do
      t_i := norm(t_i)
   t := f(t_1, ..., t_k)
   return t
end procedure norm
```

Figura 12.1: Normalización mediante estabilización

12.4 Algoritmos de normalización basados en reescritura sensible al contexto

Cualquier algoritmo de estabilización puede utilizarse para definir procedimientos de normalización. Consideremos el procedimiento de normalización *norm* de la Figura 12.1. Un procedimiento de estabilización *head-eval* (para SRTs débilmente ortogonales) que puede emplearse en *norm* se muestra en la Figura 12.2 [O'D95]. La definición de *head-eval* está basada en la estabilidad del símbolo que etiqueta la raíz de una forma estable y en la decidibilidad de las formas fuertemente estables (punto (1) del algoritmo *head-eval*). Sin embargo, decidir si un término es fuertemente estable puede involucrar cómputos de coste polinómico [O'D95]. Más aún, *head-eval* adolece de un punto débil que también es digno de mención: en el punto (2), se debe decidir cuál será el siguiente subtérmino inmediato de $f(t_1, \ldots, t_k)$ que se reducirá. La terminación de *head-eval* depende de que dicha selección se realice en tiempo finito. Por tanto, la terminación de *norm* también depende de ello. Tal como se ha discutido con anterioridad, realizar esta selección no es trivial. De hecho, supone afrontar el problema de la decidibilidad de las posiciones de los redexes necesarios para estabilizar un término, puesto que *head-eval* sirve como procedimiento de estabilización.

Al amparo de los resultados de la sección anterior, definimos un procedimiento de normalización $norm_\mu$ basado en la *rsc* bajo una función de reemplazamiento μ que obtiene formas μ-normales en sus fases intermedias en vez de formas estables (Figura 12.3). Asumiremos que el procedimiento auxiliar μ-*norm* empleado verifica $t \hookrightarrow^* \mu\text{-}norm(t)$ para todo término t; es decir, que la transformación que opera sobre el término t puede describirse como μ-reescritura.

A continuación establecemos las condiciones que aseguran la corrección y comple-

```
Procedure head-eval(t)
   while t no es fuertemente estable do                          (1)
       if t es un redex then
          t := right-side(t)
       else
          let t := f(t_1, ..., t_k);
          escoger un i ∈ {1, ..., n} cuya reducción sea segura;  (2)
          t_i := head-eval(t_i);
          t = f(t_1, ..., t_k)
   return t
end procedure head-eval
```

Figura 12.2: Un procedimiento de estabilización

titud de $norm_\mu$. Las propiedades computacionales de $norm_\mu$ dependen a su vez de las propiedades del procedimiento auxiliar μ-*norm* que se esté empleando en $norm_\mu$. En particular, la terminación y carácter determinista de μ-*norm* influirán, eventualmente, en la terminación y carácter determinista de $norm_\mu$. La terminación de μ-*norm* puede garantizarse de dos maneras:

1. Si el SRTE es μ-terminante (Capítulo 5), *cualquier* secuencia de μ-reducción terminará, por lo cual, en lo que respecta a la terminación, μ-*norm* no presenta ningún problema adicional.

2. Si no es posible demostrar la μ-terminación de $\mathcal{R}$, todavía podemos encontrar una *μ-estrategia μ-normalizante* $\mathbb{H}_{\mathcal{R}(\mu)}$ (Parte II) que puede utilizarse para definir μ-*norm*.

El carácter determinista de $norm_\mu$ puede garantizarse exigiendo la confluencia del SRTE.

Teorema 12.4.1 (Corrección de $norm_\mu$) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda, $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ y $t \in \mathcal{T}(\Sigma, V)$. Supongamos que, para todo término $s \in \mathcal{T}(\Sigma, V)$, si μ-norm(s) termina, devuelve una forma μ-normal de s. Si $norm_\mu(t)$ termina devolviendo un término $\bar{t}$, entonces $\bar{t}$ es una forma normal de t.*

DEMOSTRACIÓN. Realizamos la demostración por inducción sobre la estructura del término $\bar{t} = norm_\mu(t)$ devuelto. Si $\bar{t} \in \Sigma \cup V$, entonces $CMR^\mu(t) = t$ y n, el número de huecos de $C[\,] = CMR^\mu(t)$, es cero. Por tanto, $\bar{t} = \mu$-*norm*(t) y, puesto que $\bar{t}$ es

```
Procedure norm_μ(t)
   t := μ-norm(t);
   let t = C[t_1, ..., t_n], donde C[ ] = CMR^μ(t);
   for i := 1, ..., n do
      t_i := norm_μ(t_i)
   t := C[t_1, ..., t_n]
   return t
end procedure norm_μ
```

Figura 12.3: Un procedimiento de normalización basado en la *rsc*

una constante o una variable que, por hipótesis, es una forma μ-normal, también es una forma normal.

Si $\bar{t} \notin \Sigma \cup V$, entonces hacemos $\mu\text{-}norm(t) = C[t_1, \ldots, t_n]$ y $\bar{t} = C[\bar{t}_1, \ldots, \bar{t}_n]$. Puesto que $C[\,] = CMR^{\mu}(t) \neq \Box$, después de que $\mu\text{-}norm$ termine, tenemos que $norm_{\mu}(t)$ termina si y sólo si cada $norm_{\mu}(t_1), \ldots, norm_{\mu}(t_n)$ lo hace produciendo $\bar{t}_1, \ldots, \bar{t}_n$ respectivamente. Entonces podemos aplicar la hipótesis de inducción y llegamos a la conclusión final sin más que emplear la Proposición 4.3.1 y el Lema 2.5.14. $\Box$

Teorema 12.4.2 (Completitud de $norm_{\mu}$) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y confluente, $\mu \in M_{\Sigma}$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$ y $t \in \mathcal{T}(\Sigma, V)$. Supongamos que para todo término normalizable $s \in \mathcal{T}(\Sigma, V)$, μ-norm(s) es una forma μ-normal de s. Si $\bar{t}$ es la forma normal de t, entonces $norm_{\mu}(t)$ termina devolviendo $\bar{t}$.*

DEMOSTRACIÓN. Puesto que t tiene una forma normal $\bar{t}$, t es normalizable. Por el Teorema 12.3.3, t es μ-normalizable y, por hipótesis, $\mu\text{-}norm(t)$ devuelve una forma μ-normal s de t que, puesto que $\hookrightarrow^* \subseteq \rightarrow^*$ y $\mathcal{R}$ es confluente, verifica $t \rightarrow^* s \rightarrow^* \bar{t}$. Por la Proposición 4.3.1, $C[\,] = CMR^{\mu}(s)$ verifica que $\omega(C[\widetilde{\Omega}]) = C[\widetilde{\Omega}]$ y $\Omega < \omega(C[\widetilde{\Omega}])$. Ahora procedemos por inducción sobre la estructura de $\bar{t}$.

Si $\bar{t}$ es una constante o una variable, por la Proposición 4.3.1, $\Omega < \omega(C[\widetilde{\Omega}])$ y $\omega(C[\widetilde{\Omega}]) = C[\widetilde{\Omega}]$. Por la Proposición 4.3.5, $C[\widetilde{\Omega}] \leq s$ y $C[\widetilde{\Omega}] \leq \bar{t}$. Dado que $\Omega < \omega(C[\widetilde{\Omega}]) = C[\widetilde{\Omega}] \leq \bar{t}$ y $\bar{t} \in \Sigma \cup V$, por definición de $\leq$ tenemos que $C[\widetilde{\Omega}] = \bar{t}$; es decir, el contexto $C[\,]$ no contiene huecos. Por tanto, debe tenerse $s = \bar{t}$. Puesto que $C[\,]$ no contiene huecos, el valor de n en la definición de $norm_{\mu}$ es cero. Por tanto, $norm_{\mu}(t) = s = \bar{t}$.

Si $\bar{t}$ no es ni constante ni variable, por la Proposición 4.3.5, $C[\widetilde{\Omega}] \leq \bar{t}$, y, por la Proposición 4.3.1, $\Omega < \omega(C[\widetilde{\Omega}])$, y $\omega(C[\widetilde{\Omega}]) = C[\widetilde{\Omega}]$. Por la Proposición 4.3.11, $s =$

```
Procedure μ-norm(t)
   while t no sea una forma μ-normal do
      t := 𝕀𝕀_{R(μ)}(t)
   return t
end procedure μ-norm
```

Figura 12.4: Un procedimiento de μ-normalización basado en una μ-estrategia $\mathbb{II}_{\mathcal{R}(\mu)}$.

$\mu\text{-}norm(t) = C[s_1, \ldots, s_n]$ y, por el Teorema 12.3.2, podemos escribir $\bar{t} = C[\bar{t}_1, \ldots, \bar{t}_n]$. Por confluencia, cada s_i verifica $s_i \rightarrow^* \bar{t}_i$, para $1 \leq i \leq n$, es decir, los s_i son normalizables. Dado que $\Omega < C[\widetilde{\Omega}]$, podemos aplicar la hipótesis de inducción a cada $\bar{t}_i$, lo cual permite afirmar que cada $\bar{s}_i = norm_\mu(s_i)$ es la forma normal de s_i. Entonces, por la Proposición 4.3.1 y el Lema 2.5.14, $C[\bar{s}_1, \ldots, \bar{s}_n]$ es una forma normal que, por confluencia, coincide con $\bar{t}$. □

Observación 12.4.3 *El requisito de confluencia ha sido impuesto en el Teorema 12.4.2 sólo para asegurar una adecuada 'coordinación' entre la salida del procedimiento μ-norm y el resultado final producido por $norm_\mu$. De hecho, el Teorema 12.3.2 podría utilizarse para justificar que, al considerar SRTEs lineales por la izquierda pero no necesariamente confluentes, cualquier forma normal de t puede ser obtenida mediante una elección adecuada del procedimiento μ-norm.*

Como consecuencia del Teorema 12.3.3, siempre que garanticemos que μ-*norm* es μ-normalizante (es decir, que al ser aplicado sobre un término que posee una *forma μ-normal* termina devolviendo una forma μ-normal de éste), podemos utilizarlo en $norm_\mu$ para normalizar términos que tienen una forma *normal*. Esto se debe a que el Teorema 12.3.3 asegura que la condición "*para todo término normalizable $s \in \mathcal{T}(\Sigma, V)$, μ-norm(s) es una forma μ-normal de s*" del Teorema 12.4.2 se satisface de forma inmediata sin más que suponer que s tiene una forma normal, tal como requiere la hipótesis del teorema.

Es obvio que, si disponemos de una μ-estrategia $\mathbb{II}_{\mathcal{R}(\mu)}$ (Definición 7.1.1), podemos emplearla para definir el procedimiento de μ-normalización μ-*norm* (ver Figura 12.4). Observemos que el procedimiento μ-*norm* de la Figura 12.4 satisface el requisito "*si μ-norm(s) termina, devuelve una forma μ-normal de s*" que encontramos en el Teorema 12.4.1. Esto se debe a que, tal como esperábamos, una μ-estrategia siempre realiza un paso de μ-reescritura al ser aplicada sobre un término que no es una forma μ-normal. Si, además, $\mathbb{II}_{\mathcal{R}(\mu)}$ es μ-normalizante, entonces también se satisface el requisito del Teorema 12.4.2.

12.5 Trabajos relacionados

En [KW95], los resultados ofrecidos sobre completitud de la reescritura perezosa en lo que respecta a la normalización de términos se han establecido respecto a la ζ-igualdad de términos (ver la Sección 6.5). Es decir, se garantiza que, dado un término t, existe una forma normal respecto a la reescritura perezosa que es ζ-igual a una forma normal de t. Por tanto, nuestros resultados sobre completitud en la obtención de valores y formas normales son más precisos. También hay que señalar que los resultados dados en [KW95] se limitan a SRTs lineales por la izquierda y confluentes. Esta última condición no es necesaria para establecer nuestros resultados, al menos en lo que respecta a valores.

Los resultados sobre completitud de la reducción condicional de Maranget [Mar90] también tienen que ver con las formas normales pero han sido establecidos para una clase muy restringida de SRTs (ver las Secciones 3.5 y 6.5).

En [OF97], el problema de la completitud de los cómputos por reescritura que emplean la estrategia E no llega a plantearse.

Con relación a la completitud de los cómputos que llevan a una forma estable mediante las relaciones restringidas, dichos trabajos no ofrecen ningún resultado. Ni siquiera plantean el problema.

Ni en [KW95] ni en [Mar90] encontramos indicaciones acerca de cómo definir la función de reemplazamiento que establece las restricciones de reemplazamiento básicas cuando se desea obtener completitud de los cómputos restringidos, ni se realiza ningún análisis sobre cuánto se acercan dichos cómputos al objetivo final dependiendo de la manera en que se fijan las restricciones de reemplazamiento. Tampoco en [OF97] se aborda esta cuestión.

Capítulo 13

Formas normales infinitas

En el contexto de los lenguajes funcionales perezosos, es habitual admitir la existencia de *valores infinitos* que pueden utilizarse para asignar significado a expresiones iniciales cuyo proceso de evaluación nunca termina [FH88, Rea93]. En ese caso, los valores infinitos se definen como *límites* de secuencias infinitas convergentes de valores *parcialmente definidos* que el proceso de evaluación va refinando cada vez más, incrementando su carga semántica. Por ejemplo, consideremos el programa del Ejemplo 1.1.2. El término `from(0)` no tiene forma normal, puesto que cada aplicación de la regla para la función `from` introduce una nueva llamada a la función `from` que a su vez puede reducirse. Sin embargo, la existencia de la siguiente secuencia de reducción (donde escribimos `0`,`1`,`2`, ... para abreviar `0`,`s(0)`,`s(s(0))`,...):

$$\texttt{from(0)} \to \texttt{0::from(1)} \to \texttt{0::1::from(2)} \to \cdots$$

sugiere que el "valor infinito",

$$\texttt{0::1::2::3::}\cdots$$

correspondiente a la lista infinita de todos los números naturales, podría ser asignado como el *valor* (infinito) de la expresión `from(0)` y obtenido como supremo de la cadena creciente

$$\Omega \sqsubseteq \texttt{0::}\Omega \sqsubseteq \texttt{0::1::}\Omega \sqsubseteq \cdots$$

De forma similar, la idea de *reescritura infinita*, es decir, reescritura donde se contempla la posibilidad de que existan secuencias de reescritura infinitas (que, eventualmente, involucran términos infinitos e incluso SRTEs con reglas constituidas por términos infinitos) y se estudian sus propiedades (en particular, la definición de secuencias de reducción que obtienen *formas normales infinitas*) ha suscitado algunos esfuerzos de investigación [Bou85, DKP91, KKSV95, Mid97].

En la Sección 13.1 demostramos que la μ-terminación de un SRTE implica su carácter infinitamente normalizante cuando μ iguala o excede la función de reempla-

zamiento canónica asociada al SRTE.

En la Sección 13.2 estudiamos cómo emplear estrategias de reescritura sensible al contexto para definir estrategias infinitamente normalizantes.

13.1 Normalización infinita

Nuestra discusión sobre cuestiones relativas a la *normalización infinita* de términos seguirá el enfoque de Middeldorp [Mid97] (ver la Sección 2.3.4). Se trata de una aproximación simple aunque suficiente para discutir el tema con el fin de aplicar los resultados al área de la programación. En particular, y en contraste con [DKP91, KKSV95], sólo consideramos *secuencias de reducción infinita de longitud ω (el primer ordinal límite) cuyo límite sea una forma normal infinita.*

El siguiente teorema demuestra que el carácter μ-terminante de un SRTE implica que es infinitamente normalizante.

Teorema 13.1.1 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu_{\mathcal{R}}^{com} \sqsubseteq \mu$. Si $\mathcal{R}$ es μ-terminante, entonces $\mathcal{R}$ es infinitamente normalizante.*

DEMOSTRACIÓN. Dado un término t, demostramos que es posible utilizar la *rsc* para construir una secuencia de reducción fuertemente convergente como sigue: primero μ-reducimos (de forma arbitraria, dado que el SRTE es μ-terminante) el término t a una forma μ-normal $\bar{t}_1$. Obtenemos $t = \bar{t}_0 \rightarrow^* \bar{t}_1$ (aunque en este caso base podríamos escribir también $t = \bar{t}_0 \hookrightarrow^* \bar{t}_1$). Sea $C_1[\,] = CMR^\mu(\bar{t}_1)$, con $\bar{t}_1 = C_1[t_{11}, \ldots, t_{1n_1}]$. Si $C_1[\,]$ no contiene ningún hueco, se tiene que $\bar{t}_1 = C_1[\,]$ es una forma normal, con lo que hemos obtenido la secuencia deseada. Si $C_1[\,]$ contiene algún hueco, sea d_1 la longitud del camino más corto que lleva desde la raíz de $C_1[\,]$ a alguno de los huecos. Por la Proposición 4.3.1, $C_1[\widetilde{\Omega}]$ es un término rígido fuertemente estable y, por tanto $C_1[\,] \neq \Box$. Luego $d_1 > 0$. Ahora μ-reduciremos cada t_{1i} hasta una forma μ-normal $\bar{s}_{1i}$; obtenemos así contextos reemplazantes máximos $C_{1i}[\,] = CMR^\mu(\bar{s}_{1i})$ que son rígidos para $1 \leq i \leq n_1$. Por el Lema 2.5.13, el contexto $C_2[\,] = C_1[C_{11}[\,], \ldots, C_{1n_1}[\,]]$ también es rígido y hemos obtenido $\bar{t}_1 \rightarrow^* C_2[t_{21}, \ldots, t_{2n_2}] = \bar{t}_2$. Definimos d_2 como $d_2 = d_1 + min(\{d_{1i} \mid 1 \leq i \leq n_1\})$ donde $d_{1i} > 0$ es la longitud del camino más corto que va desde la raíz de $C_{1i}[\,]$ a uno de sus huecos (de nuevo, si todo contexto $C_{1i}[\,]$ careciera de huecos, $C_1[C_{11}[\,], \ldots, C_{1n_1}[\,]]$ sería una forma normal, luego asumiremos que existe un $d_{1i} > 0$). Adviértase que $0 < d_1 < d_2$. Las reducciones que arrancan de $\bar{t}_2$ se llevan a cabo sobre redexes cuya profundidad es mayor o igual a d_2, y los subtérminos cuya profundidad es menor que d_2 son estables, puesto que solapan el contexto rígido $C_2[\,]$. Si repetimos este proceso de forma indefinida es fácil concluir que, o bien obtenemos una forma normal finita del término inicial o construimos una secuencia $t = \bar{t}_0 \rightarrow^* \bar{t}_1 \rightarrow^* \bar{t}_2 \rightarrow^* \cdots$ que resulta ser fuertemente convergente: si

$d \geq 0$, siempre existe un $j \geq 1$ tal que $d_{j-1} \leq d < d_j$ (hacemos $d_0 = 0$). Por la forma de construir la derivación, la profundidad d' de cualquier redex en la subderivación final $\bar{t}_j \rightarrow^* \bar{t}_{j+1} \rightarrow^* \cdots$ satisface $d' \geq d_j$ y, por tanto, $d' > d$, tal como se requiere. Dado que toda secuencia fuertemente convergente tiene un límite (que, en este caso, es una forma normal), obtenemos la conclusión deseada. □

Este resultado sólo tiene sentido si $\mu \sqsubset \mu_\top$ puesto que, en este caso, la terminación y la μ-terminación difieren (si consideramos un SRT terminante, no pueden existir secuencias de reducción infinitas). Por tanto, podemos aplicar el resultado a SRTEs no terminantes cuya μ-terminación pueda demostrarse.

La demostración del Teorema 13.1.1 sugiere la posibilidad de definir estrategias de normalización infinita basadas en la *rsc*.

13.2 Estrategias de normalización infinita basadas en la reescritura sensible al contexto

Bajo ciertas condiciones, las estrategias estabilizantes pueden ser empleadas para obtener la forma normal infinita de un término (ver la Sección 2.4.1). Dado que sabemos cómo definir estrategias sensibles al contexto que realizan reducción de índices y que dichas estrategias son estabilizantes, los Teoremas 7.3.3 y 7.3.12 constituyen la base para la definición de estrategias de estabilización.

La estrategia *parallel outermost*, $\mathbb{F}_{po}$, es infinitamente normalizante ([Mid97], Corolario 7.6). Para demostrar que $\mathbb{F}^{\mu}_{po}$ también lo es, necesitaríamos demostrar que $\mathbb{F}^{\mu}_{po}$ es equitativa. Por desgracia, esto no es así.

Ejemplo 13.2.1 *Consideremos el SRT $\mathcal{R}$ [Mid97]:*

```
a       → f(a,a)
f(b,x) → c
```

Adviértase que `a` *no tiene forma normal, aunque sí tiene una forma normal infinita que podemos expresar como la solución de la ecuación* $t =$ `f(`t`,`t`)`*. Tenemos la siguiente* $\mu^{com}_{\mathcal{R}}$*-derivación correspondiente a la estrategia* $\mathbb{F}^{\mu^{com}_{\mathcal{R}}}_{po}$*.*

$$\underline{\texttt{a}} \hookrightarrow \texttt{f(}\underline{\texttt{a}}\texttt{,a)} \hookrightarrow \texttt{f(f(}\underline{\texttt{a}}\texttt{,a),a)} \hookrightarrow \cdots$$

que no es equitativa, puesto que el redex `a` *en el segundo término* `f(a,a)` *de la derivación nunca resulta seleccionado para realizar un paso de reducción.*

Sin embargo, esta 'deficiencia' es lógica; de la estrategia $\mathbb{F}^{\mu}_{po}$ sólo podríamos esperar que obtuviera una forma μ-normal infinita. Como quiera que no estamos interesados en obtener este tipo de términos, exploramos la posibilidad de definir estrategias de normalización infinita basadas en la *rsc*. En primer lugar, tenemos el siguiente resultado.

Teorema 13.2.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT lineal por la izquierda y $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Si $\mathcal{R}$ es μ-normalizante y existe una μ-estrategia μ-normalizante $\mathbb{H}_{\mathcal{R}(\mu)}$ para $\mathcal{R}$, entonces $\mathcal{R}$ es infinitamente normalizante.*

DEMOSTRACIÓN. Similar al Teorema 13.1.1 pero definiendo las secuencias de μ-reducción intermedias mediante $\mathbb{H}_{\mathcal{R}(\mu)}$. □

El Teorema 13.2.2 demuestra que las μ-estrategias μ-normalizantes exhiben una propiedad que las estrategias normalizantes no poseen: son útiles para obtener formas normales infinitas cuando se puede asegurar que el SRT es μ-normalizante.

Tratándose de normalización infinita, no tiene sentido definir un procedimiento que 'produzca' la forma normal infinita de una expresión puesto que, por definición, dicho proceso no terminará nunca. Sin embargo, apoyándose en el Teorema 13.2.2 y teniendo en cuenta la demostración del Teorema 13.1.1, donde construíamos una secuencia de reducción convergente a una forma normal infinita apoyándonos en subsecuencias de reducción sensible al contexto, es inmediato utilizar una μ-estrategia μ-normalizante $\mathbb{H}_{\mathcal{R}(\mu)}$ para definir una estrategia (paralela) de normalización infinita $\mathbb{F}_{\mathcal{R},\mu}$:

$$\mathbb{F}_{\mathcal{R},\mu}(t) = \begin{cases} \mathbb{H}_{\mathcal{R}(\mu)}(t) & \text{si } t \text{ no es una forma } \mu\text{-normal} \\ C[s_1, \ldots, s_n] & \text{en otro caso, donde:} \\ & \quad C[\,] = CMR^{\mu}(t) \\ & \quad t = C[t_1, \ldots, t_n] \text{ y} \\ & \quad s_i := \mathbb{F}_{\mathcal{R},\mu}(t_i) \text{ para } 1, \ldots, n \end{cases}$$

Teorema 13.2.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT lineal por la izquierda y sea $\mu \in M_\Sigma$ tal que $\mu^{com}_{\mathcal{R}} \sqsubseteq \mu$. Si $\mathbb{H}_{\mathcal{R}(\mu)}$ es una μ-estrategia μ-normalizante para $\mathcal{R}$, entonces $\mathbb{F}_{\mathcal{R},\mu}$ es una estrategia infinitamente normalizante para $\mathcal{R}$.*

13.3 Trabajos relacionados

El Teorema 13.1.1 complementa los resultados sobre estrategias de normalización infinita dados en [Mid97, KKSV95] puesto que dichos resultados no pueden aplicarse a SRTs lineales por la izquierda sino sólo a SRTs ortogonales. Como contrapartida, nuestros resultados dependen de la μ-terminación del SRT o de que seamos capaces de definir una estrategia μ-normalizante para el SRT.

Parte IV

Extensiones. Restricciones de reemplazamiento generales

Capítulo 14

Extensiones

En este capítulo discutimos algunas extensiones de la reescritura sensible al contexto.

En la Sección 14.1 analizamos la utilización de las restricciones de reemplazamiento sensibles al contexto para limitar los cómputos por estrechamiento. Demostramos que las mejoras derivadas de considerar las restricciones de reemplazamiento en la reescritura pueden obtenerse de forma análoga para el estrechamiento.

En la Sección 14.2 estudiamos la utilización de funciones de reemplazamiento para limitar la relación de reescritura inducida por un SRT condicional (SRTC). Demostramos que, en el caso de los SRTCs *normales*, las buenas propiedades de la *rsc* para el caso incondicional pueden ser aprovechadas también para mejorar los cómputos mediante reescritura condicional.

14.1 Estrechamiento

Los lenguajes lógico-funcionales integran en un solo entorno computacional las características más interesantes tanto de los lenguajes funcionales puros como de los lenguajes lógicos puros. La posibilidad de emplear variables lógicas, estructuras de datos parciales y búsqueda de soluciones constituye la aportación fundamental de la parte lógica de estos lenguajes. Por otro lado, es posible definir expresiones anidadas, funciones de orden superior y explotar la naturaleza determinista de éstas como consecuencia del componente funcional del lenguaje [AFV98, AFV98b, Han95c, Han94, Red95, Red85, Rod95].

Para dar soporte efectivo al cómputo con variables lógicas, es preciso proporcionar un nuevo mecanismo operacional, diferente de la reescritura, que sea capaz de instanciar éstas. En la mayoría de los casos, particularmente cuando se busca un principio computacional completo, dicho mecanismo es el estrechamiento.

Definición 14.1.1 (Estrechamiento [Hul80, Sla74]**)** *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $t, s \in \mathcal{T}(\Sigma, V)$. Un término t se estrecha a s, escrito $t \leadsto_{[u,\alpha,\sigma]} s$ (o simplemente $t \leadsto_\sigma s$) si existe $u \in O_\Sigma(t)$ y un renombramiento de una regla $\alpha : l \to r \in R$ tales que $t|_u$ y l unifican con* umg *idempotente σ y $s = \sigma(t[r]_u)$.*

Una derivación por estrechamiento (o $\leadsto$-derivación) $t \leadsto^*_\theta s$ es tal que, o bien $t = s$ y $\theta = \epsilon$ o bien $t \leadsto_{\sigma_0} t_1 \leadsto_{\sigma_1} \cdots t_{n-1} \leadsto_{\sigma_{n-1}} s$ y $\theta = \sigma_{n-1} \circ \cdots \circ \sigma_1 \circ \sigma_0$.

Observación 14.1.2 *Con relación a una posible noción de estrechamiento para SRTEs, observemos que la propia definición de estrechamiento obliga a que la sustitución empleada para efectuar un paso elemental de estrechamiento sea un unificador más general. Por tanto, las variables extra existentes en la regla empleada para realizar el paso elemental son tratadas, en realidad, como constantes (en ese paso) pues no pueden ser instanciadas por el* umg *computado. Existen otras nociones de estrechamiento que no utilizan* umg*'s para realizar los pasos elementales (ver, por ejemplo, [AEH94]). Sin embargo, en esta sección emplearemos la definición estándar de estrechamiento (Definición 14.1.1) para SRTs y nos ceñiremos, en consecuencia, a éstos.*

14.1.1 Estrechamiento sensible al contexto

Con el fin de transferir al campo lógico-funcional las ventajas que ofrece la *rsc* en el contexto de la implementación de la programación funcional mediante técnicas de reescritura, consideremos en lo que sigue la idea de limitar las secuencias de derivación por estrechamiento . Restringiremos (mediante una función de reemplazamiento) las ocurrencias de los subtérminos que pueden considerarse para realizar el estrechamiento.

Definición 14.1.3 (Estrechamiento sensible al contexto) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\mu \in M_\Sigma$. Un término t es μ-estrechado a s, y se escribe $t \overset{\mu}{\leadsto}_{[u,\alpha,\sigma]} s$ (o simplemente $t \overset{\mu}{\leadsto}_\sigma s$) si y sólo si $t \leadsto_{[u,\alpha,\sigma]} s$ en $\mathcal{R}$ y $u \in O^\mu(t)$.*

Análogamente al caso de la *rsc*, tenemos que $\overset{\mu}{\leadsto} \subseteq \leadsto$. Por tanto, el estrechamiento sensible al contexto (*esc*) considera un espacio de búsqueda menor que el del estrechamiento no restringido. Veamos un ejemplo.

Ejemplo 14.1.4 *Consideremos el SRT y la función de reemplazamiento del Ejemplo 3.3.3. El paso de estrechamiento (innecesario)*

`if(and(x,false),y+s(0),0)` $\leadsto_{\{\mathtt{y} \mapsto 0\}}$ `if(and(x,false),s(0),0)`

no tiene lugar usando el esc*, puesto que la ocurrencia 2 del objetivo inicial no es reemplazante.*

Establecemos, para el *esc*, un resultado similar al Teorema 11.1.4 para la *rsc*. Para ello nos apoyamos en el siguiente teorema, que expresa la correspondencia entre la *rsc* y el *esc* de forma análoga al resultado estándar de Hullot que asocia una derivación de estrechamiento a cada derivación de reescritura y viceversa [Hul80].

En el enunciado del siguiente teorema, entendemos por sustitución normalizada una sustitución σ tal que $\sigma(x)$ es una forma normal para todo $x \in V$.

Teorema 14.1.5 *Sea $\mathcal{R}$ un SRT y $\mu \in M_\Sigma$. Sea $t \in \mathcal{T}(\Sigma, V)$ y W un conjunto finito de variables que contiene $Var(t)$. Sea η una sustitución normalizada con $\mathcal{D}om(\eta) \subseteq Var(t)$. Consideremos la siguiente derivación de μ-reescritura que arranca en $\eta(t)$:*

$$\eta(t) = s_0 \overset{[u_0,\alpha_0]}{\hookrightarrow} s_1 \overset{[u_1,\alpha_1]}{\hookrightarrow} s_2 \hookrightarrow \cdots \hookrightarrow s_{n-1} \overset{[u_{n-1},\alpha_{n-1}]}{\hookrightarrow} s_n \qquad (1)$$

Existe una derivación de μ-estrechamiento asociada que parte de t:

$$t = t_0 \overset{\mu}{\leadsto}_{[u_0,\alpha_0,\sigma_0]} t_1 \overset{\mu}{\leadsto}_{[u_1,\alpha_1,\sigma_1]} t_2 \overset{\mu}{\leadsto} \cdots \overset{\mu}{\leadsto} t_{n-1} \overset{\mu}{\leadsto}_{[u_{n-1},\alpha_{n-1},\sigma_{n-1}]} t_n, \qquad (2)$$

y para todo i, $0 \leq i \leq n$, existe una sustitución η_i y un conjunto finito de variables W_i tal que:

1. *$\mathcal{D}om(\eta_i) \subseteq W_i$,*
2. *η_i es una sustitución normalizada*[1]*,*
3. *$\eta_{|W} = (\eta_i \circ \theta_i)_{|W}$ y*
4. *$\eta_i(t_i) = s_i$*

donde $\theta_0 = \epsilon$ y $\theta_{i+1} = \sigma_{i+1} \circ \theta_i$. También al contrario, para cada derivación de μ-estrechamiento (2) y sustitución η tal que $\theta_n \leq \eta\, [\![W]\!]$, existe una derivación de μ-reescritura (1) asociada.

DEMOSTRACIÓN. La demostración es básicamente la misma que el Teorema 1 de Hullot [Hul80], puesto que cada paso de reducción utiliza exactamente la misma ocurrencia que el correspondiente paso de estrechamiento y viceversa. Puesto que al considerar derivaciones de estrechamiento las restricciones de reemplazamiento son las mismas que en el caso de la reescritura y éstas sólo dependen de la ocurrencia considerada, la demostración de Hullot es válida incluso cuando se consideran las restricciones de reemplazamiento. □

Cuando $\mu = \mu_\top$, el Teorema 14.1.5 expresa exactamente lo mismo que el resultado de Hullot (ver [Hul80]). En el contexto de dicho teorema, las sustituciones normalizadas son importantes. Para asegurar que las sustituciones computadas son normalizadas,

[1] En el resultado original de Hullot, el requisito de que cada sustitución η_i esté normalizada se debe a la necesidad de asegurar que la instanciación no introduce nuevos redexes sin mediar la participación del término instanciado. Por este motivo, no podemos relajar aquí esta condición (exigiendo, por ejemplo, que η_i sea μ-normalizada), porque, en ese caso no queda asegurada la propiedad anterior

consideraremos sistemas de reescritura basados en constructores (SRT BCs). El siguiente resultado es auxiliar.

Lema 14.1.6 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT BC lineal por la izquierda. Sean $t, s \in \mathcal{T}(\Sigma, V)$, tales que $t \leadsto^*_\theta s$. Entonces $\theta_{|Var(t)}$ es una sustitución constructora.*

Por tanto, al trabajar con SRT BCs, las sustituciones computadas en cada paso y la composición de éstas para formar la sustitución computada asociada a una derivación de estrechamiento, son sustituciones normalizadas. Ahora presentamos el resultado principal de esta sección.

Teorema 14.1.7 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRT BC lineal por la izquierda, $\mathcal{B} \subseteq \mathcal{C}$ y $\mu \in M_\Sigma$ tal que $\mu^{\mathcal{B}}_{\mathcal{R}} \sqsubseteq \mu$. Sea $t \in \mathcal{T}(\Sigma, V)$ y $\delta \in \mathcal{T}(\mathcal{B}, V)$. Entonces, $t \leadsto^*_{\theta'} \delta$ si y sólo si $t \overset{\mu}{\leadsto}{}^*_\theta \delta$ y $\theta \leq \theta'[\![Var(t)]\!]$.*

DEMOSTRACIÓN. La parte *si* es inmediata puesto que $\overset{\mu}{\leadsto} \subseteq \leadsto$ y se tiene $\theta = \theta'$. Para la parte *sólo si*, por el Lema 14.1.6 y el Teorema 14.1.5 (tomando $\eta = \theta'_{|Var(t)}$, $\mu = \mu_\top$ y considerando que, debido a que $\eta = \theta'_{|Var(t)}$, tenemos $\eta_n = \epsilon$ y por tanto $\eta_n(\delta) = \delta$), obtenemos $t \leadsto^*_{\theta'} \delta \Rightarrow \theta'(t) \rightarrow^* \delta$. Por el Teorema 11.1.4, $\theta'(t) \rightarrow^* \delta \Leftrightarrow \theta'(t) \hookrightarrow^* \delta$. Finalmente, por el Teorema 14.1.5, $\theta'(t) \hookrightarrow^* \delta \Rightarrow t \overset{\mu}{\leadsto}{}^*_\theta \delta$ y $\theta \leq \theta'[\![Var(t)]\!]$. □

Como consecuencia del Teorema 14.1.7, el *esc* obtiene soluciones más generales que el estrechamiento no restringido, sin pérdida de completitud (cuando consideramos derivaciones de éxito que llevan a términos constructores).

Ejemplo 14.1.8 *Una derivación de* esc *que concluye con éxito la evealuación del objetivo mostrado en el Ejemplo 14.1.4 es:*

$$\texttt{if(and(x,false),y+s(0),0)} \overset{\mu}{\leadsto}{}^*_{\{\texttt{x}\mapsto\texttt{true}\}} \texttt{0}$$

La derivación de estrechamiento no restringida

$$\texttt{if(and(x,false),y+s(0),0)} \leadsto^*_{\{\texttt{x}\mapsto\texttt{true},\texttt{y}\mapsto\texttt{0}\}} \texttt{0}$$

de mayor longitud (y que obtiene una respuesta menos general) no se produce.

14.1.2 Resolución de ecuaciones

Al permitir instanciar las variables de una expresión, el mecanismo de estrechamiento puede emplearse para resolver ecuaciones si podemos conseguir que dicha ecuación sea evaluada (tras instanciar las variables convenientemente) al valor `true`. Una *ecuación* es una expresión $t \approx s$ donde t y s son términos. Definiendo el símbolo $\approx$ como un símbolo de operación binario (una función booleana) las ecuaciones pueden interpretarse también (y tratarse) como términos. Por tanto, todas las nociones sobre términos tales como la de ocurrencia, subtérmino, sustitución, estrechamiento, etc. serán empleadas también con ecuaciones.

Si no nos limitamos a SRTs terminantes (la situación habitual al trabajar con lenguajes lógico-funcionales con semántica operacional perezosa) es posible que no existan formas normales para algunos términos. Por tanto, la validez de una ecuación se define respecto a la *igualdad estricta* sobre términos. Esto es así, por ejemplo, en lenguajes lógico-funcionales perezosos como *BABEL* [MR92], *K-LEAF* [GLMP91], etc., y constituye la elección más adecuada en presencia de estructuras de datos y computaciones infinitas. Por tanto, una sustitución σ es una *solución* para una ecuación $t \approx s$ si y sólo si $\sigma(t)$ y $\sigma(s)$ son reducibles al mismo término constructor básico. Dado un SRT BC $\mathcal{R} = (\mathcal{C} \uplus \mathcal{F}, R)$, la operación $\approx$ viene definida por el SRT $\mathcal{S}_{\approx} = (\mathcal{C} \uplus \{\texttt{true}\} \uplus \{\approx, \wedge\}, S_{\approx})$, donde $S_{\approx}$ es:

$$\begin{array}{rcl} S_{\approx} & = & \{c \approx c \rightarrow \texttt{true} \mid c \in \mathcal{C} \text{ and } ar(c) = 0\} \cup \\ & & \{c(x_1, \ldots, x_k) \approx c(y_1, \ldots, y_k) \rightarrow x_1 \approx y_1 \wedge \cdots \wedge x_k \approx y_k \mid c \in \mathcal{C}, k > 0\} \cup \\ & & \{\texttt{true} \wedge x \rightarrow x\} \end{array}$$

donde $\wedge$ se utiliza como un símbolo asociativo por la derecha. Estas son las reglas que definen la *igualdad estricta* para la signatura. Añadiendo estas reglas al SRT, podemos resolver ecuaciones estrechando éstas hasta alcanzar el valor `true`.

El SRT $\mathcal{S}_{\approx}$ es lineal por la izquierda y $\mu^{com}_{\mathcal{S}_{\approx}}$ es:

$\mu^{com}_{\mathcal{S}_{\approx}}(c) = \emptyset$ para todo $c \in \mathcal{C}\}$,

$\mu^{com}_{\mathcal{S}_{\approx}}(\wedge) = \{1\}$ y $\mu^{com}_{\mathcal{S}_{\approx}}(\approx) = \{1, 2\}$

Por tanto, dado un SRT BC lineal por la izquierda $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$, un subconjunto de símbolos constructores $\mathcal{B} \subseteq \mathcal{C}$, y una función de reemplazamiento μ tal que $\mu^{\mathcal{B}}_{\mathcal{R}} \sqsubseteq \mu$, es obvio que, siendo $\mathcal{R}_{\approx} = (\Sigma \cup \{\wedge, \approx\}, R \cup S_{\approx})$, también tenemos que

$$\mu^{com}_{\mathcal{R}} \sqcup \mu^{com}_{\mathcal{S}_{\approx}} \sqcup \mu_{\mathcal{B}} = \mu^{\mathcal{B}}_{\mathcal{R}} \sqcup \mu^{com}_{\mathcal{S}_{\approx}} = \mu^{\mathcal{B}}_{\mathcal{R}_{\approx}} \sqsubseteq \mu \sqcup \mu^{com}_{\mathcal{S}_{\approx}}$$

Por tanto, simplemente necesitamos añadir $\mu^{com}_{\mathcal{S}_{\approx}}$ a μ para garantizar que μ puede emplearse para resolver ecuaciones mediante el *esc*.

Es interesante observar que, en lo que respecta a las tareas de resolución de ecuaciones, puesto que cualquier derivación de estrechamiento de un objetivo ecuacional e que culmine con la obtención de un valor, sólo puede computar el valor `true`, es decir, $\mathcal{B} = \{\texttt{true}\}$, trivialmente tenemos que $\mu_{\mathcal{B}} = \mu_{\perp}$ y, por tanto, $\mu^{\mathcal{B}}_{\mathcal{R}_{\approx}} = \mu^{com}_{\mathcal{R}_{\approx}}$, es decir, es suficiente con emplear la función de reemplazamiento canónica de $\mathcal{R}_{\approx}$.

Ejemplo 14.1.9 *Consideremos el SRT del Ejemplo 14.1.4 y la ecuación*

```
if(and(x,false),y+s(0),0) ≈ 0
```

Obtenemos

```
if(and(x,false),y+s(0),0) ≈ 0
```

$$\overset{\mu}{\rightsquigarrow}_{\{\mathtt{x}\mapsto\mathtt{true}\}} \mathtt{if(false,y+s(0),0)} \approx \mathtt{0}$$
$$\overset{\mu}{\rightsquigarrow}_{\epsilon} \mathtt{0} \approx \mathtt{0}$$
$$\overset{\mu}{\rightsquigarrow}_{\epsilon} \mathtt{true}$$

y también

$$\mathtt{if(and(x,false),y+s(0),0)} \approx \mathtt{0}$$
$$\overset{\mu}{\rightsquigarrow}_{\{\mathtt{x}\mapsto\mathtt{false}\}} \mathtt{if(false,y+s(0),0)} \approx \mathtt{0}$$
$$\overset{\mu}{\rightsquigarrow}_{\epsilon} \mathtt{0} \approx \mathtt{0}$$
$$\overset{\mu}{\rightsquigarrow}_{\epsilon} \mathtt{true}$$

Adviértase que $\{\mathtt{x}\mapsto\mathtt{true}\}$ *y* $\{\mathtt{x}\mapsto\mathtt{false}\}$ *son las soluciones más generales de la ecuación inicial. El estrechamiento no restringido iniciaría el cómputo de un número infinito de soluciones correspondientes a instanciaciones inútiles de la variable* `y`.

14.2 SRTs condicionales

En los lenguajes de programación cuyos programas constan de conjuntos de ecuaciones (y son teorías ecuacionales desde el punto de vista lógico), resulta de interés restringir la satisfacción de cada ecuación haciéndola depender de ciertas *condiciones*. Dichas condiciones pueden expresarse mediante predicados arbitrarios (ver [DO90]) pero es más común emplear (conjunciones de) otras ecuaciones para ello. Esto da lugar a *teorías ecuacionales condicionales*, constituidas por fórmulas condicionales de la forma:

$$t_1 = s_1 \wedge \cdots \wedge t_n = s_n \Rightarrow l = r$$

y donde se asume que las variables participantes en los términos t_i, s_i para $1 \leq i \leq n$ y l, r están cuantificadas universalmente [BK86, Kap84, Klo92].

De forma similar a como los SRTs sirven para implementar los cómputos con teorías ecuacionales, la implementación de teorías ecuacionales condicionales puede efectuarse mediante *sistemas de reescritura condicionales* (SRTCs). De forma análoga a como la reescritura o el estrechamiento se emplean como mecanismos operacionales en tareas de deducción automática y resolución de ecuaciones con SRTs, respectivamente, tanto la reescritura como el estrechamiento han sido extendidos al caso condicional y empleados en programación con SRTCs [DOS88, DO90, DOS88, DP88, DS87, Höl89, Kap84]

El estudio de los SRTCs resulta ser sustancialmente más complejo que en el caso incondicional (SRTs). Como ejemplo, baste señalar que ni siquiera es decidible, en el caso general, si un término se encuentra o no en forma normal. Esto se debe a que, a diferencia del caso incondicional, la presencia de un redex en un término no implica que éste sea reducible dado que esto último dependerá, tal como veremos en seguida,

de que se satisfagan o no las condiciones asociadas a la regla que determina el redex [DOS88, Kap84].

Una regla de reescritura condicional es de la forma[2] $l \to r \Leftarrow c$ donde c se denomina la *parte condicional* de la regla. Con más detalle podemos escribir:

$$l \to r \Leftarrow t_1 = s_1 \wedge \cdots \wedge t_n = s_n$$

para $n \geq 0$. Debe cumplirse que $l \notin V$. Cuando $n = 0$ escribiremos simplemente $l \to r$. Un SRTC $\mathcal{R} = (\Sigma, R)$ consta de un conjunto de reglas condicionales R. A cada SRTC $\mathcal{R}$ le asociamos el SRT *subyacente* $\mathcal{R}_u$ que se obtiene a partir de las reglas en R eliminando la parte condicional de éstas.

Observación 14.2.1 (Variables extra en los SRTCs) *Con relación a las restricciones habituales acerca de la ocurrencia de variables extra en el componente $l \to r$ de una regla de reescritura, y dado que ahora hay que tener en cuenta la parte condicional, se ha estudiado el comportamiento de los cómputos tanto de la reescritura como del estrechamiento realizados respecto a SRTCs sujetos a distintas formas de limitar la presencia de variables extra [BK86, DO90, Han95b, MH94, SMI95]. En el contexto de los SRTCs llamamos variables extra de una regla condicional $l \to r \Leftarrow c$ a aquéllas que no se encuentran en $Var(l)$. De acuerdo con [MH94, SMI95], se distinguen cuatro clases de SRTCs: los de tipo 1 donde no se admiten variables extra en absoluto, los de tipo 2 donde se admiten variables extra en la parte condicional, los de tipo 3 donde se acepta que r contenga variables extra siempre que ocurran también en la parte condicional y los de tipo 4 donde no hay limitación alguna. La mayor parte de los trabajos sobre SRTCs se refieren a los tipos 1 y 2, aunque recientemente han aparecido algunos trabajos importantes que estudian propiedades de los SRTCs de tipo 3 [Han95b, SMI95].*

Para definir la relación de reescritura asociada a un SRTC $\mathcal{R}$ hay que decidirse por una de las muchas nociones existentes de *satisfacción* de una restricción. Tal como apunta [DOS88], una vez decidido qué se entiende por satisfacción de la parte condicional c de una regla $l \to r \Leftarrow c$, se define la relación de reescritura condicional en un paso $\to$ como sigue.

Definición 14.2.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC y $t, s \in \mathcal{T}(\Sigma, V)$. Decimos que t se reescribe a s (a la ocurrencia u), escrito $t \to s$, si existe una sustitución $\sigma \in Subst(\Sigma, V)$ y una regla $l \to r \Leftarrow c$ tales que $t|_u = \sigma(l)$, $s = t[\sigma(r)]$ y $\sigma(c)$* cumple el criterio de satisfacción establecido.

En [DOS88b] se denomina *sustitución factible* a toda sustitución σ que determina un redex de una regla condicional y también satisface las condiciones de dicha regla.

[2] Esta notación con la conectiva '$\Rightarrow$' invertida es la más utilizada en la actualidad [Klo92].

De acuerdo con el criterio de satisfacción de la parte condicional de las reglas del SRTC, se puede establecer una nueva taxonomía de SRTCs que incluso determina una notación distinta para expresar las reglas del SRTC (ver [BK86, Kap84] y sobre todo [DOS88, DO90]).

Observación 14.2.3 (Criterios de satisfacción condicional para SRTCs) *Nos interesan aquí los siguientes tipos de SRTCs, según el criterio adoptado para satisfacer las condiciones [DOS88]:*

1. SRTCs semi ecuacionales*: Para establecer si se satisface una condición elemental $t_i = s_i$ (para una determinada sustitución σ), se comprueba si $\sigma(t_i) \leftrightarrow^* \sigma(s_i)$. La notación empleada para las reglas de estos SRTCs es la básica:*

   $$l \to r \Leftarrow t_1 = s_1 \wedge \cdots \wedge t_n = s_n$$

2. SRTCs estándar *(*standard *o* join CTRS's*): La semántica de las ecuaciones $t_i = s_i$ presentes en las condiciones se establece como convergencia (*joinability*) de las instancias correspondientes $\sigma(t_i)$ y $\sigma(s_i)$ a un reducto común, es decir: $\sigma(t_i) \downarrow \sigma(s_i)$. La notación empleada es:*

   $$l \to r \Leftarrow t_1 \downarrow s_1 \wedge \cdots \wedge t_n \downarrow s_n$$

3. SRTCs normales*: Son un caso especial de SRTCs estándar donde todas las condiciones $t_i = s_i$ para i, $1 \leq i \leq n$ verifican que s_i es una forma normal básica (respecto al SRT subyacente $\mathcal{R}_u$). Por tanto, la semántica para las instancias de las condiciones dada por convergencia de instancias se convierte simplemente en reducción a la forma normal correspondiente, es decir: $\sigma(t_i) \to^* \sigma(s_i) = s_i$. La notación empleada es:*

   $$l \to r \Leftarrow t_1 \to^! s_1 \wedge \cdots \wedge t_n \to^! s_n$$

   donde $t \to^! s$ indica la reducción de t hasta alcanzar una forma normal básica s (respecto a $\mathcal{R}_u$).

Después de examinar algunos de los criterios usuales para determinar la satisfacción de la parte condicional de una regla de reescritura (Observación 14.2.3), podemos observar que la Definición 14.2.2 encubre una forma de descripción 'circular' de la relación de reescritura condicional puesto que emplea la propia relación que se está definiendo. Por este motivo se emplea una definición inductiva equivalente de la relación. Dado que nuestro interés se centra en los SRTCs normales, adecuaremos la definición a este criterio concreto de satisfacción condicional; para otros criterios tendremos definiciones similares.

Definición 14.2.4 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC normal. Definimos inductivamente los SRTs $\mathcal{R}_n = (\Sigma, R_n)$ para $n \geq 0$ como sigue:*

$$\begin{array}{lcl} R_0 & = & \emptyset \\ R_{n+1} & = & \{\sigma(l) \to \sigma(r) \mid l \to r \Leftarrow c \in R \text{ y } \sigma(t) \to^*_{\mathcal{R}_n} s \text{ para toda } t \to^! s \text{ en } c\} \end{array}$$

La relación existente entre las Definiciones 14.2.2 y 14.2.4 es la siguiente (ver [Kap84, BK86])

Teorema 14.2.5 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC normal y $t, s \in \mathcal{T}(\Sigma, V)$. Entonces, $t \to_{\mathcal{R}} s$ si y sólo si existe un n tal que $t \to_{\mathcal{R}_n} s$.*

De acuerdo con [DOS88], definimos la *profundidad* de un paso de reducción $t \to_{\mathcal{R}} s$ como sigue: la profundidad de un paso incondicional es 0; la profundidad de un paso que emplea una regla $l \to r \Leftarrow t_1 \to^! s_1 \wedge \cdots \wedge t_n \to^! s_n$ es $1 + max(\{m_i \mid 1 \leq i \leq n\})$, donde m_i es la profundidad de la derivación que satisface la restricción $t_i \to^! s_i$ para $1 \leq i \leq n$. La profundidad de una derivación $t \to^*_{\mathcal{R}} s$ es el máximo de las profundidades de sus derivaciones en un paso.

Observación 14.2.6 *Aunque pudiera pensarse que la restricción a SRTCs normales es excesiva, Dershowitz y Okada demostraron que cualquier SRTC estándar puede ser simulado por un SRTC normal [DO90] (aunque con pérdida de la linealidad por la izquierda).*

14.2.1 Reescritura condicional sensible al contexto

Ahora estamos en condiciones de aplicar los conocimientos adquiridos sobre *rsc* para extender ésta a SRTCs. La definición de la reescritura *condicional* sensible al contexto (*rcsc*) es la obvia, empleando $\hookrightarrow$ en lugar de $\to$. También particularizamos la definición a los SRTCs normales.

Definición 14.2.7 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC, $\mu \in M_\Sigma$ y $t \in \mathcal{T}(\Sigma, V)$. Decimos que t se μ-reescribe a s (a la ocurrencia u), escrito $t \hookrightarrow s$, si existe una sustitución $\sigma \in Subst(\Sigma, V)$ y una regla*

$$l \to r \Leftarrow t_1 \to^! s_1 \wedge \cdots \wedge t_n \to^! s_n$$

tales que $t|_u = \sigma(l)$, $u \in O^\mu(t)$, $s = t[\sigma(r)]$ y $\sigma(t_i) \hookrightarrow^* s_i$ para todo i, $1 \leq i \leq n$.

Y tenemos una definición inductiva análoga a la Definición 14.2.4 para la *rcsc* (empleando $\hookrightarrow$ en lugar de $\to$). La definición de profundidad de un paso de reducción $t \hookrightarrow s$ y de una derivación $t \hookrightarrow^* s$ es análoga al caso no restringido. Para la *rcsc* tenemos un resultado análogo al Teorema 14.2.5.

Teorema 14.2.8 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC normal, $\mu \in M_\Sigma$ y $t, s \in \mathcal{T}(\Sigma, V)$. Entonces, $t \hookrightarrow_{\mathcal{R}(\mu)} s$ si y sólo si existe un n tal que $t \hookrightarrow_{\mathcal{R}_n(\mu)} s$.*

DEMOSTRACIÓN. Por inducción en la profundidad m de $t \hookrightarrow_{\mathcal{R}(\mu)} s$. Si $m = 0$ es inmediato, puesto que $t \hookrightarrow_{\mathcal{R}(\mu)} s$ incondicionalmente, luego $t \hookrightarrow_{\mathcal{R}_1(\mu)} s$. Si $m > 0$, entonces $t \hookrightarrow_{\mathcal{R}(\mu)} s$ si y sólo si existe una regla condicional $l \to r \Leftarrow t_1 \to^! s_1 \wedge \cdots \wedge t_n \to^! s_n$ y una sustitución σ tales que $t|_u = \sigma(l)$ para $u \in O^\mu(t)$ y existen derivaciones $\sigma(t_i) \hookrightarrow^*_{\mathcal{R}(\mu)} s_i$ de profundidad m_i para cada i, $1 \leq i \leq n$, de forma que $m = 1 + max(\{m_i \mid 1 \leq i \leq n\})$. Siendo $m > m_i$ para todo i, $1 \leq i \leq n$, por la hipótesis de inducción, existen n_i tales que $t_i \hookrightarrow^*_{\mathcal{R}_{n_i}} s_i$ para $1 \leq i \leq n$. Puesto que, para todo p, $\hookrightarrow_{\mathcal{R}_p(\mu)} \subseteq \hookrightarrow_{\mathcal{R}_{p+1}(\mu)}$, tenemos que $t \hookrightarrow_{\mathcal{R}_{1+max(\{n_i \mid 1 \leq i \leq n\})}} s$. □

En el caso de los SRTCs normales, el Teorema 12.1.4 nos permite establecer condiciones para emplear exclusivamente la *rsc* manteniendo un mínimo grado de completitud en los cómputos. En lugar de demostrarlo directamente para cada caso (formas estables, valores, etc.) emplearemos las funciones de descomposición (*unraveling*) de Marchiori [Mar96]. Mediante estas funciones podemos asociar un SRT a un SRTC determinado de modo que es posible reflejar ciertas propiedades de los SRTs en los SRTCs correspondientes. Una propiedad P de un SRT se traslada mediante una función de descomposición a una *ultrapropiedad ultra-P* del SRTC cuya transformación genera el SRT en cuestión.

Definición 14.2.9 (Función de descomposición [Mar96]) *Una función de descomposición* U *es una función computable de SRTCs a SRTs tal que*

1. $\downarrow_{\mathcal{R}} \subseteq \downarrow_{\mathsf{U}(\mathcal{R})}$
2. $\mathsf{U}(\mathcal{R} \cup \mathcal{S}) = \mathcal{R} \cup \mathsf{U}(\mathcal{S})$ *si* $\mathcal{R}$ *es un SRT.*

En el caso de los SRTCs normales, existe una función de descomposición con propiedades muy interesantes.

Definición 14.2.10 (Función de descomposición U_n [Mar96]) *Dada una regla* α : $l \to r \Leftarrow t_1 \to^! s_1 \wedge \cdots \wedge t_n \to^! s_n$ *con* $n \geq 1$, $Var(l) = \{x_1, \ldots, x_p\}$ *y* $\mathtt{u}_\alpha$ *un símbolo nuevo asociado a la regla* α *y cuya aridad es* $ar(\mathtt{u}_\alpha) = n + p$, *el resultado de* $\mathsf{U}_\mathsf{n}(\alpha)$ *es:*

$$l \to \mathtt{u}_\alpha(t_1, \ldots, t_n, x_1, \ldots, x_p)$$
$$\mathtt{u}_\alpha(s_1, \ldots, s_n, x_1, \ldots, x_p) \to r$$

Al aplicar U_n sobre un SRTC $\mathcal{R} = (\Sigma, R)$, obtenemos un SRT $\mathsf{U}_\mathsf{n}(\mathcal{R}) = (\Sigma \cup \Upsilon, \mathsf{U}_\mathsf{n}(R))$ donde

$$\Upsilon = \{\mathtt{u}_\alpha \mid \alpha \in R\}$$

recoge los nuevos símbolos introducidos para cada regla condicional $\alpha \in R$.

La función $\mathsf{U_n}$ preserva la linealidad por la izquierda del SRTC, es decir, si $\mathcal{R}$ es lineal por la izquierda, $\mathsf{U_n}(\mathcal{R})$ también lo es [Mar96]. La otra propiedad relevante de esta función de descomposición viene expresada por el siguiente teorema.

Teorema 14.2.11 [Mar96] *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC normal, de tipo 1 y lineal por la izquierda. Sean $t, s \in \mathcal{T}(\Sigma, V)$. Entonces, $t \to^*_{\mathcal{R}} s$ si y sólo si $t \to^*_{\mathsf{U_n}(\mathcal{R})} s$.*

Empleando este teorema podemos demostrar con facilidad varios resultados sobre la completitud de la *rcsc*. Previamente necesitamos las siguientes definiciones.

Definición 14.2.12 (Función de reemplazamiento normal) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC normal. Dada una regla $\alpha : l \to r \Leftarrow t_1 \to^! s_1 \wedge \cdots \wedge t_n \to^! s_n$, se define*

$$\mu_\alpha = \sqcup_{1 \leq i \leq n} \mu_{s_i}$$

y, asociada al SRTC, la función de reemplazamiento normal

$$\mu^{nor}_{\mathcal{R}} = \sqcup_{\alpha \in R} \mu_\alpha$$

Definición 14.2.13 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC. Dada una regla $\alpha \in R$, se define la función de reemplazamiento μ_Υ como sigue:*

$$\mu_\Upsilon(\mathtt{u}_\alpha) = \mathbb{N}^+_{n(\alpha)}$$

siendo $\mathtt{u}_\alpha$ como en la Definición 14.2.9 y $n(\alpha)$ el número de ecuaciones de que consta la parte condicional de α.

Proposición 14.2.14 *Si $\mathcal{R} = (\Sigma, R)$ es un SRTC normal, entonces*

$$\mu^{com}_{\mathsf{U_n}(\mathcal{R})} = \mu^{com}_{\mathcal{R}} \sqcup \mu_\Upsilon \sqcup \mu^{nor}_{\mathcal{R}}$$

DEMOSTRACIÓN. Inmediata empleando las definiciones de $\mu^{com}_{\mathcal{R}}$, μ_Υ y $\mu^{nor}_{\mathcal{R}}$ y la transformación $\mathsf{U_n}$. □

Teorema 14.2.15 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTC normal, de tipo 1 y lineal por la izquierda y $\mathcal{R}' = \mathsf{U_n}(\mathcal{R})$. Sea $\mu \in M_\Sigma$ y $t, s \in \mathcal{T}(\Sigma, V)$. Entonces, $t \hookrightarrow^*_{\mathcal{R}(\mu)} s$ si y sólo si $t \hookrightarrow^*_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} s$.*

DEMOSTRACIÓN. Demostramos la parte *solo si* de la doble implicación por inducción en la longitud q de la derivación $t \hookrightarrow^*_{\mathcal{R}(\mu)} s$. Si $q = 0$, es inmediato. Si $q > 0$, sea $t \hookrightarrow_{\mathcal{R}(\mu)} t' \hookrightarrow^*_{\mathcal{R}(\mu)} s$. Demostramos que si $t \hookrightarrow_{\mathcal{R}(\mu)} t'$ entonces $t \hookrightarrow^*_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} s$ por inducción en la profundidad m de un paso $t \hookrightarrow_{\mathcal{R}(\mu)} t'$. Si $m = 0$, entonces existe una ocurrencia $u \in O^\mu(t)$, y una regla incondicional $l \to r \in R$ tal que $t|_u = \sigma(l)$

y $s = t[\sigma(r)]_u$. Por la Definición 14.2.9, $l \to r$ es una regla de $\mathcal{R}' = \mathsf{U}_{\mathsf{n}}(\mathcal{R})$, luego $t \hookrightarrow_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} s$

Si $m > 0$, entonces, para toda ecuación $t_i \to^! s_i$ de la parte condicional c de la regla $\alpha : l \to r \Leftarrow t_1 \to^! s_1 \wedge \cdots \wedge t_n \to^! s_n$ aplicada, tenemos que $\sigma(t_i) \hookrightarrow^*_{\mathcal{R}(\mu)} \sigma(s_i)$ es una derivación de profundidad menor que m para todo $1 \leq i \leq n$. Por la hipótesis de inducción, $\sigma(t_i) \hookrightarrow^*_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} \sigma(s_i)$ para todo $1 \leq i \leq n$. Por la Definición 14.2.10,

$$\sigma(l) \hookrightarrow_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} \sigma(\mathtt{u}_\alpha(t_1, \ldots, t_n, x_1, \ldots, x_p)),$$

es decir,

$$\sigma(l) \hookrightarrow_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} \mathtt{u}_\alpha(\sigma(t_1), \ldots, \sigma(t_n), \sigma(x_1), \ldots, \sigma(x_p)).$$

Dado que $\mu_\Upsilon(\mathtt{u}_\alpha) = \mathbb{N}^+_n$, por la Proposición 3.3.6, tenemos que

$$\mathtt{u}_\alpha(\sigma(t_1), \ldots, \sigma(t_n), \sigma(x_1), \ldots, \sigma(x_p)) \hookrightarrow^*_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} \mathtt{u}_\alpha(\sigma(s_1), \ldots, \sigma(s_n), \sigma(x_1), \ldots, \sigma(x_p)),$$

es decir,

$$\mathtt{u}_\alpha(\sigma(t_1), \ldots, \sigma(t_n), \sigma(x_1), \ldots, \sigma(x_p)) \hookrightarrow^*_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} \sigma(\mathtt{u}_\alpha(s_1, \ldots, s_n, x_1, \ldots, x_p))$$

y, de nuevo por la Definición 14.2.9,

$$\sigma(\mathtt{u}_\alpha(s_1, \ldots, s_n, x_1, \ldots, x_p)) \hookrightarrow_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} \sigma(r).$$

Dado que $u \in O^\mu(t)$, y $\mu \sqsubseteq \mu \sqcup \mu_\Upsilon$, tenemos que $u \in O^{\mu \sqcup \mu_\Upsilon}(t)$ y, empleando la Proposición 3.3.6, concluimos que $t \hookrightarrow^*_{\mathcal{R}'(\mu \sqcup \mu_\Upsilon)} s$.

La demostración en el sentido contrario seguiría un razonamiento similar. □

El siguiente teorema permite extender los resultados sobre completitud para reescritura incondicional obtenidos en los capítulos anteriores al caso de la reescritura condicional.

Teorema 14.2.16 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTC normal, de tipo 1 y lineal por la izquierda. Sea $\mathcal{B} \subseteq \mathcal{C}$, $\mu \in M_\Sigma$, $t, s \in \mathcal{T}(\Sigma, V)$ y $\delta \in \mathcal{T}(\mathcal{B}, V)$.*

1. *Si $\mu^{com}_{\mathcal{R}} \sqcup \mu^{nor}_{\mathcal{R}} \sqsubseteq \mu$ y $s = c(\tilde{s})$ para algún $c \in \mathcal{C}$ y $t \to^*_{\mathcal{R}} s$, entonces existe un s' tal que $root(s') = c$ y $t \hookrightarrow^*_{\mathcal{R}(\mu)} s'$.*
2. *Si $\mu^{\mathcal{B}}_{\mathcal{R}} \sqcup \mu^{nor}_{\mathcal{R}} \sqsubseteq \mu$, entonces $t \to^*_{\mathcal{R}} \delta$ si y sólo si $t \hookrightarrow^*_{\mathcal{R}(\mu)} \delta$.*

DEMOSTRACIÓN. Sea $\mathcal{R}' = \mathsf{U}_{\mathsf{n}}(\mathcal{R})$.

1. Por el Teorema 14.2.11, $t \to^*_{\mathcal{R}} s$ si y sólo si $t \to^*_{\mathcal{R}'} s$. Por el Teorema 11.1.2, $t \to^*_{\mathcal{R}'} s$ si y sólo si $t \hookrightarrow^*_{\mathcal{R}'(\mu^{com}_{\mathcal{R}'})} s'$. Por la Proposición 14.2.14 y el Teorema 14.2.15 concluimos el resultado deseado.
2. Similar a la anterior empleando el Teorema 11.1.4.

□

14.3 Trabajos relacionados

La extensión de la *rsc* al marco de la reescritura módulo una teoría cuya signatura incluye algunos símbolos de función que poseen la propiedad asociativa o conmutativa (teorías AC) ha sido realizada por Ferreira y Ribeiro [FR98]. Además de plantear la citada extensión (*context-sensitive AC-rewriting*), en [FR98] se analiza la terminación de la relación de reescritura sensible al contexto módulo una teoría AC. La técnica empleada es, a su vez, una extensión del trabajo de Zantema sobre terminación de la *rsc* [Zan97].

Capítulo 15

Un marco para el análisis de restricciones de reemplazamiento sintácticas

La restricción en los reemplazamientos expresada por las restricciones sensibles al contexto constituyen un caso particular de lo que venimos llamando restricción de reemplazamiento sintáctica.

En la Sección 15.1 proponemos un marco para expresar restricciones de reemplazamiento sintácticas generales. Desligamos la definición de la restricción de reemplazamiento de los símbolos particulares que aparecen en los términos y centramos nuestra atención en los términos en sí, sin atender a su estructura interna.

Con el fin de analizar las propiedades de una restricción de reemplazamiento, proponemos un marco de análisis que consta de dos componentes fundamentales: lógico y algebraico.

En la Sección 15.2 presentamos el componente *lógico* o *descriptivo* en el que proponemos un conjunto de *propiedades* que una restricción de reemplazamiento (o un conjunto de ellas) puede o no cumplir. Dichas propiedades permiten caracterizar *clases* de restricciones de reemplazamiento (en lugar de restricciones individuales).

En la Sección 15.3 describimos el componente *algebraico*, que nos permite relacionar, comparar y combinar restricciones de reemplazamiento entre sí. Este componente permite aprovechar los resultados establecidos para la *rsc* puesto que ésta no es más que un caso particular del marco aquí estudiado.

En la Sección 15.4 empleamos el marco de análisis esbozado para establecer condiciones que aseguren la terminación, confluencia y completitud de la reescritura restringida.

15.1 Restricciones de reemplazamiento en reescritura

A continuación, introducimos lo que entendemos por *restricción de reemplazamiento* sintáctica.

Definición 15.1.1 (Restricción de reemplazamiento sintáctica) *Una función* $\gamma : \mathcal{T}(\Sigma, V) \to \mathcal{P}(Occ)$ *cuya aplicación a un término* t *denotamos como* $O^\gamma(t)$ *es una* restricción de reemplazamiento sintáctica *(o simplemente una* restricción de reemplazamiento*) si, para todo* $t \in \mathcal{T}(\Sigma, V)$, $O^\gamma(t) \subseteq O(t)$.

Al conjunto $O^\gamma(t)$, resultado de la aplicación de una restricción de reemplazamiento γ sobre un término t lo denominamos "conjunto de *ocurrencias* γ*-reemplazantes* de t". De forma similar, $\overline{O^\gamma}(t) = O(t) \backslash O^\gamma(t)$ es el conjunto de ocurrencias no γ-reemplazantes de t.

El conjunto de restricciones de reemplazamiento definibles para una determinada signatura Σ es

$$\Gamma(\Sigma) = \{\gamma \in \mathcal{T}(\Sigma, V) \to \mathcal{P}(Occ) \mid O^\gamma(t) \subseteq O(t)\}$$

o simplemente Γ, cuando no haya confusión respecto a la signatura subyacente.

Dado un SRTE $\mathcal{R} = (\Sigma, R)$, podemos emplear una restricción de reemplazamiento $\gamma \in \Gamma(\Sigma)$ para restringir la relación de reescritura $\to_\mathcal{R}$.

Definición 15.1.2 (Relación de reescritura restringida) *Sea* $\mathcal{R} = (\Sigma, R)$ *un SRTE y* $\gamma \in \Gamma(\Sigma)$. *Un término* $t \in \mathcal{T}(\Sigma, V)$ *es* γ*-reescrito al término* s *y se denota como* $t \overset{u}{\hookrightarrow}_{\mathcal{R}(\gamma)} s$ *si* $t \overset{u}{\to}_\mathcal{R} s$ *y* $u \in O^\gamma(t)$.

Como es habitual, abreviaremos $t \overset{u}{\hookrightarrow}_{\mathcal{R}(\gamma)} s$ como $t \hookrightarrow_{\mathcal{R}(\gamma)} s$, $\hookrightarrow_\gamma$ e incluso $\hookrightarrow$, si esto no da lugar a confusión.

Observación 15.1.3 *Al considerar una restricción de reemplazamiento* γ *el concepto de* redex *no resulta ser tan 'operacional' como en el caso de la reescritura no restringida. La caracterización de un subtérmino como un redex (respecto a un SRTE) depende exclusivamente de las partes izquierdas del SRTE en cuestión. Ahora bien, la propiedad más interesante de un subtérmino que calificamos como un redex es que puede ser reducido mediante la(s) correspondiente(s) regla(s) del SRTE.*

Al considerar una restricción de reemplazamiento, esta propiedad deja de verificarse y pasa a depender de que la ocurrencia donde el redex está situado pertenezca al conjunto de ocurrencias reemplazantes del término donde el redex se encuentra.

Dada una restricción de reemplazamiento γ y un SRTE $\mathcal{R}$, llamaremos γ-redex a todo redex γ-reemplazante. El conjunto de ocurrencias de γ-redexes de un término

es $O^{\gamma}_{\mathcal{R}}(t) = O_{\mathcal{R}}(t) \cap O^{\gamma}(t)$. Un término se encuentra en forma γ-normal si $O^{\gamma}_{\mathcal{R}}(t) = \emptyset$ y es γ-reducible si $O^{\gamma}_{\mathcal{R}}(t) \neq \emptyset$.

15.2 Descripción de restricciones de reemplazamiento

Para poder realizar un estudio descriptivo de las restricciones de reemplazamiento, debemos plantearnos cómo identificar éstas a distintos niveles de abstracción. Tratar de describir las restricciones a un nivel de abstracción elevado (conociendo pocas cosas acerca de su definición concreta) permite aislar las propiedades abstractas que condicionan los resultados teóricos y prácticos de interés, y ofrece la posibilidad de establecer dichos resultados de forma más general.

Con este fin, caracterizamos *clases* de restricciones de reemplazamiento que comparten *propiedades* comunes.

Podemos identificar una propiedad PROP acerca de restricciones de reeplazamiento como un subconjunto Γ^{PROP} de Γ, es decir, $\Gamma^{\mathsf{PROP}} \subseteq \Gamma$. Diremos que γ tiene la propiedad PROP si $\gamma \in \Gamma^{\mathsf{PROP}}$.

Notación 15.2.1 *Dadas las propiedades* PROP1, ..., PROPn, *el conjunto*

$$\Gamma^{\mathsf{PROP1}+\cdots+\mathsf{PROPn}} = \Gamma^{\mathsf{PROP1}} \cap \ldots \cap \Gamma^{\mathsf{PROPn}}$$

reúne todas las restricciones que satisfacen las propiedades PROP1, ..., PROPn *simultáneamente.*

En la Figura 15.1, proponemos una lista de propiedades básicas que empleamos para caracterizar conjuntos de restricciones de reemplazamiento.

15.2.1 Justificación de las propiedades básicas escogidas

Para motivar las propiedades escogidas, nos apoyaremos tanto en la carga intuitiva propia de algunas de ellas como en ciertas propiedades de los índices fuertemente secuenciales. Dado un SRTE cualquiera, podemos definir una restricción de reemplazamiento γ_{Ω} basada en el cálculo de las ocurrencias índice de cada término.

$$\text{Para todo } t \in \mathcal{T}(\Sigma, V),\ u \in O^{\gamma_{\Omega}}(t) \text{ si y sólo si } u \in \mathcal{I}(t[\Omega]_u).$$

Utilizaremos γ_{Ω} para motivar el conjunto de propiedades escogido.

La propiedad ROOT se satisface de forma inmediata por la definición dada para γ_{Ω}. Por tanto, $\gamma_{\Omega} \in \Gamma^{\mathsf{ROOT}}$.

Sean $C, t, s \in \mathcal{T}(\Sigma, V)$ y $\sigma \in Subst(\Sigma, V)$ cuantificados universalmente:

ROOT	$\epsilon \in O^\gamma(t)$
DOWN	$u.v \in O^\gamma(C[t]_u) \Rightarrow u \in O^\gamma(C[t]_u)$
SUBTERM	$u.v \in O^\gamma(C[t]_u) \Rightarrow v \in O^\gamma(t)$
DECOMP	$u.v \in O^\gamma(C[t]_u)) \Rightarrow (u \in O^\gamma(C[t]_u) \wedge v \in O^\gamma(t))$, es decir DECOMP=DOWN+SUBTERM,
COMP	$(u \in O^\gamma(C[t]_u) \wedge v \in O^\gamma(t)) \Rightarrow u.v \in O^\gamma(C[t]_u)$
CONTEXT	$\forall u \in O^\gamma(C), v \in O(C)\ \ (u \parallel v \Rightarrow u \in O^\gamma(C[t]_v))$
REPL	$u \in O^\gamma(C[t]_u) \Rightarrow u \in O^\gamma(C[s]_u)$
SUBST	$u \in O^\gamma(t) \Rightarrow u \in O^\gamma(\sigma(t))$
$\overline{\text{SUBST}}$	$u \in \overline{O^\gamma}(t) \Rightarrow u \in \overline{O^\gamma}(\sigma(t))$

Figura 15.1: Un conjunto de propiedades básicas

Las definiciones de DOWN y SUBTERM se inspiran en la Proposición 2.5.17. Ambas propiedades son una descripción detallada de DECOMP, que es la propiedad de los índices fuertemente secuenciales expresada por la Proposición 2.5.17(1). Por tanto, $\gamma_\Omega \in \Gamma^{\text{DECOMP}}$.

Observación 15.2.2 *Es interesante notar que si* $\gamma \in \Gamma^{\text{DOWN}}$*, esto implica que* $\overline{O^\gamma}(t)$ *es un* cono invertido *en el orden* $(O(t), \leq)$*. Esta propiedad resulta útil a la hora de implementar sistemas que incorporen este tipo de restricciones: al inspeccionar un término* t *a la búsqueda de un redex (reemplazante), si encontramos una ocurrencia no reemplazante* $u \in \overline{O^\gamma}(t)$*, podemos detener la búsqueda de otros redexes situados por debajo de éste, puesto que no serán reemplazantes.*

La propiedad COMP está relacionada con la *transitividad* de los índices [Klo92, KM91, TSEP93]. Un índice v en un término s se denomina *transitivo* si, para cualquier otro índice u en un término t, tenemos que $u.v$ es un índice en $t[s]_u$ [TSEP93]. La transitividad parece esencial para implementar estrategias de reducción (de índices) de forma eficiente. Tras efectuar el reemplazamiento de un redex situado sobre un índice, la transitividad posibilita reanudar la búsqueda del próximo redex a reducir *localmente*, es decir, sin volver de nuevo a la raíz del término maximal [Str89, TSEP93]. La restricción de reemplazamiento γ_Ω *no* satisface COMP [Klo92, KM91]. Si definimos γ_τ como

$$u \in O^{\gamma_\tau}(t) \text{ si y sólo si } u \text{ es un índice transitivo en } t[\Omega]_u$$

entonces γ_τ satisface COMP (ver el Lema 4.2 de [TSEP93]), es decir, $\gamma_\tau \in \Gamma^{\text{COMP}}$. En [TSEP93] se define la clase de SRTs donde tales índices existen. Por desgracia,

γ_τ no hereda todas las propiedades de γ_Ω. Por ejemplo, es fácil comprobar que γ_τ satisface DOWN pero no SUBTERM.

La propiedad CONTEXT significa que el carácter reemplazante de una ocurrencia de un determinado término no depende del contexto que *circunda* dicho término. Se inspira en la propiedad de los índices expresada por la Proposición 2.5.17(2).

La propiedad REPL aplicada a ocurrencias de índices significa que la cualidad de ser un índice de un término t sólo depende de la posición de dicho índice en el término y no del subtérmino situado sobre la ocurrencia de interés. Por tanto, tenemos que $\gamma_\Omega \in \Gamma^{\mathsf{REPL}}$.

Observación 15.2.3 *Las propiedades* CONTEXT *y* REPL *están relacionadas con la persistencia de las restricciones de reemplazamiento durante sucesivas operaciones de reemplazamiento. Establecen que el* status *de la restricción de reemplazamiento para una ocurrencia determinada de un término no depende del término localizado en esa ocurrencia ni del contexto que circunda a dicho término sino sólo de los símbolos situados inmediatamente por encima de dicha ocurrencia. Esto es importante para implementar el proceso de reescritura restringida.*

La siguiente proposición formaliza la interpretación sintáctica de estas propiedades.

Proposición 15.2.4 *Sean $t, s \in \mathcal{T}(\Sigma, V)$. Entonces, $\gamma \in \Gamma^{\mathsf{CONTEXT+REPL}}$ si y sólo si para toda $u \in O(t) \cap O(s)$ tal que $sprefix_t(u) = sprefix_s(u)$, tenemos que $u \in O^\gamma(t) \Leftrightarrow u \in O^\gamma(s)$.*

DEMOSTRACIÓN. En primer lugar, demostramos la parte *sólo si* de la doble implicación. Sea $u \in O(t) \cap O(s)$ tal que $sprefix_t(u) = sprefix_s(u)$. Empleando CONTEXT y REPL, podemos transformar el término $t = C[t']_u$ en $s = D[s']_u$ sin alterar el carácter reemplazante de la ocurrencia u como mostramos a continuación: Dada $u = u'.i$, sea $s = D[s']_{u'.i} = D'[f(\tilde{s})]_{u'}$ y $t = C[t']_{u'.i} = C'[f(\tilde{t})]_{u'}$, donde $s_i = s'$ and $t_i = t'$. Entonces, por REPL, $u \in O^\gamma(C[t']_u) \Rightarrow u \in O^\gamma(C[s']_u)$. Por CONTEXT, $u \in O^\gamma(C[s']_u) \Rightarrow u \in O^\gamma(C'[f(\tilde{s})]_{u'})$. Continuamos este proceso, modificando $C'[\,]$ hasta obtener $D'[\,]$ y, por tanto, hasta obtener s manteniendo el mismo prefijo estricto. Observemos que si hubiéramos empezado con una ocurrencia no reemplazante llegaríamos a una conclusión análoga (donde la ocurrencia seguiría siendo, en este caso, no reemplazante), puesto que, de no ser así, al aplicar las transformaciones en sentido contrario llegaríamos a una contradicción. El proceso de transformación y el resultado obtenido es completamente análogo si lo inicamos con el término s para obtener t. Por tanto, tenemos la equivalencia deseada: $O^\gamma(t) \Leftrightarrow O^\gamma(s)$.

Para demostrar la parte *si*, hacemos $t = C[t']_u$. Si $u \in O^\gamma(t)$, cualquier modificación de t por debajo de $t|_u$ o en ocurrencias v no comparables con u para obtener un término s deja intacto su prefijo estricto, es decir, $sprefix_t(u) = sprefix_s(u)$. Luego, por hipótesis, $u \in O^\gamma(s)$, es decir, se verifican REPL y CONTEXT. □

SUBST asegura que la aplicación de una sustitución no cambia el carácter de una ocurrencia reemplazante en el término original. $\overline{\mathsf{SUBST}}$ representa la propiedad análoga para ocurrencias no reemplazantes. Esto es importante si queremos describir las restricciones de reemplazamiento especificándolas primero para ciertos términos, por ejemplo las partes izquierdas de las reglas de reescritura de un SRTE, y luego extendiendolas a otros términos.

No es difícil comprobar que, en general, ni γ_Ω ni γ_τ satisfacen CONTEXT o SUBST. Por tanto, resumiendo:

$$\gamma_\Omega \in \Gamma^{\mathsf{ROOT+DECOMP+REPL}}$$
$$\gamma_\tau \in \Gamma^{\mathsf{ROOT+DOWN+COMP+REPL}}$$

En la Sección 15.3 comparamos estas restricciones de reemplazamiento con las restricciones sensibles al contexto.

15.2.2 Relaciones fundamentales entre las propiedades básicas

El primer resultado de interés establece que CONTEXT y REPL dependen de ROOT, COMP y DECOMP.

Proposición 15.2.5 $\mathsf{ROOT} + \mathsf{COMP} + \mathsf{DECOMP} \Rightarrow \mathsf{CONTEXT} + \mathsf{REPL}$

DEMOSTRACIÓN. Demostramos en primer lugar que $\mathsf{ROOT} + \mathsf{COMP} + \mathsf{DECOMP}$ implica REPL. Lo hacemos por inducción sobre una ocurrencia $u \in O^\gamma(C[t]_u)$ cualquiera. Si $u = \epsilon$, es inmediato. Si $u = v.w$ con $|v| < |u|$, escribimos $C[\ ]_u = C_1[C_2[\ ]_w]_v$. Por DOWN, tenemos que $v \in O^\gamma(C_1[C_2[t]_w]_v)$ y, por la hipótesis de inducción, $v \in O^\gamma(C_1[C_2[s]_w]_v)$. Por SUBTERM, $w \in O^\gamma(C_2[t]_w)$ y, por la hipótesis de inducción, de nuevo, $w \in O^\gamma(C_2[s]_w)$. Por COMP, $u = v.w \in O^\gamma(C_1[C_2[s]_w]_v) = O^\gamma(C[s]_u)$.

Para demostrar que $\mathsf{ROOT} + \mathsf{COMP} + \mathsf{DECOMP}$ implica CONTEXT procedemos por inducción sobre la ocurrencia $u \in O^\gamma(C)$. Escribimos $u = u_1.u_2$, $C[\ ]_u = C_1[C_2[\ ]_{u_2}]_{u_1}$ y, asumiendo que $u \parallel v$, distinguimos dos casos:

1. Si $v \parallel u_1$, entonces, por DOWN, $u_1 \in O^\gamma(C_1)$. Por SUBTERM, $u_2 \in O^\gamma(C_2)$ y, por la hipótesis de inducción, $u_1 \in O^\gamma(C_1[t]_v)$. Por REPL (demostrado en el apartado anterior), $u_1 \in O^\gamma(C[t]_v)$. Por COMP, $u_1.u_2 \in O^\gamma(C[t]_)$.

2. Si $u_1 \leq v$, entonces, por DOWN, $u_1 \in O^\gamma(C_1)$. Por la hipótesis de inducción, $u_2 \in O^\gamma(C_2[t]_v)$ y, por COMP, $u_1.u_2 \in O^\gamma(C_1[C_2[t]_v]_{u_1}) = O^\gamma(C[t]_v)$

Con esto concluimos la demostración. □

Como consecuencia inmediata de la Proposición 15.2.4, tenemos el siguiente resultado.

Proposición 15.2.6 $\mathsf{CONTEXT} + \mathsf{REPL} \Rightarrow \mathsf{SUBST}$. $\mathsf{CONTEXT} + \mathsf{REPL} \Rightarrow \overline{\mathsf{SUBST}}$

Otras conexiones interesantes entre las propiedades básicas enumeradas son las siguientes.

Proposición 15.2.7 *Si* $\gamma \in \Gamma^{\mathsf{DOWN}}$ *y para todo término* t *se verifica* $O^\gamma(t) \neq \emptyset$, *entonces* $\gamma \in \Gamma^{\mathsf{ROOT}}$.

DEMOSTRACIÓN. Dado que para todo término t, existe una ocurrencia $u \in O^\gamma(t)$, por DOWN podemos decir que $\epsilon \in O^\gamma(t)$, es decir, $\gamma \in \Gamma^{\mathsf{ROOT}}$. □

Proposición 15.2.8 *Si* $\gamma \in \Gamma^{\mathsf{DOWN+REPL}}$ *y existe un término* t *tal que* $O^\gamma(t) \neq \emptyset$, *entonces* $\gamma \in \Gamma^{\mathsf{ROOT}}$.

DEMOSTRACIÓN. Si $u \in O^\gamma(t)$, entonces, por DOWN, $\epsilon \in O^\gamma(t)$ y, por REPL, para todo s, $\epsilon \in O^\gamma(s)$, luego $\gamma \in \Gamma^{\mathsf{ROOT}}$. □

La siguiente proposición establece que la diferencia entre los conjuntos de restricciones $\Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}}$ y $\Gamma^{\mathsf{ROOT+COMP+DECOMP}}$ es simplemente $\gamma_\perp$.

Proposición 15.2.9 $\Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}} = \Gamma^{\mathsf{ROOT+COMP+DECOMP}} \cup \{\gamma_\perp\}$.

DEMOSTRACIÓN. La inclusión

$$\Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}} \supseteq \Gamma^{\mathsf{ROOT+COMP+DECOMP}} \cup \{\gamma_\perp\}$$

es inmediata pues, por un lado, $\gamma_\perp$ verifica (trivialmente) las propiedades COMP, DECOMP, CONTEXT y REPL, es decir, $\gamma_\perp \in \Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}}$. Por otro lado, por la Proposición 15.2.5, tenemos que $\Gamma^{\mathsf{ROOT+COMP+DECOMP}} \subseteq \Gamma^{\mathsf{CONTEXT+REPL}}$ y, por definición, $\Gamma^{\mathsf{ROOT+COMP+DECOMP}} \subseteq \Gamma^{\mathsf{COMP+DECOMP}}$, es decir,

$$\begin{aligned} \Gamma^{\mathsf{ROOT+COMP+DECOMP}} &\subseteq \Gamma^{\mathsf{COMP+DECOMP}} \cap \Gamma^{\mathsf{CONTEXT+REPL}} \\ &= \Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}}. \end{aligned}$$

La inclusión

$$\Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}} \subseteq \Gamma^{\mathsf{ROOT+COMP+DECOMP}} \cup \{\gamma_\perp\}$$

se demuestra como sigue: supongamos que $\gamma \in \Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}}$ es tal que $\gamma \neq \gamma_\perp$. Entonces, existe un término t tal que $O^\gamma(t) \neq \emptyset$. Por la Proposición 15.2.8, $\gamma \in \Gamma^{\mathsf{ROOT}}$, es decir, $\gamma \in \Gamma^{\mathsf{ROOT+COMP+DECOMP+CONTEXT+REPL}} \subseteq \Gamma^{\mathsf{ROOT+COMP+DECOMP}}$. □

15.3 Un marco algebraico para el análisis de restricciones de reemplazamiento

El orden $\subseteq$ sobre $\mathcal{P}(Occ)$ puede extenderse de la forma usual a un orden (punto a punto) $\sqsubseteq$ sobre restricciones de reemplazamiento.

Definición 15.3.1 (Orden entre restricciones de reemplazamiento) *Sea Σ una signatura. Se define el orden parcial $\sqsubseteq$ sobre $\Gamma(\Sigma)$ como sigue. Para todo $\gamma, \gamma' \in \Gamma(\Sigma)$ y término $t \in \mathcal{T}(\Sigma, V)$:*

$$\gamma \sqsubseteq \gamma' \Leftrightarrow O^{\gamma}(t) \subseteq O^{\gamma'}(t).$$

Esto nos permite disponer de una estructura algebraica $(\Gamma, \sqsubseteq, \gamma_\bot, \gamma_\top, \sqcup, \sqcap)$ que es un retículo completo distributivo: para todo término t, tenemos $O^{\gamma_\top}(t) = O(t)$ ($\gamma_\top$ es el elemento máximo), $O^{\gamma_\bot}(t) = \emptyset$ ($\gamma_\bot$ es el elemento mínimo), $O^{\gamma \sqcup \gamma'}(t) = O^{\gamma}(t) \cup O^{\gamma'}(t)$ ($\sqcup$ es la operación supremo del retículo) y $O^{\gamma \sqcap \gamma'}(t) = O^{\gamma}(t) \cap O^{\gamma'}(t)$ ($\sqcap$ es la operación ínfimo del retículo). Las operaciones $\sqcup$ y $\sqcap$ son idempotentes y distributivas (cada una respecto a la otra): para todo $\gamma, \gamma', \gamma'' \in \Gamma$, tenemos $\gamma \sqcup \gamma = \gamma$, $\gamma \sqcap \gamma = \gamma$, $\gamma \sqcup (\gamma' \sqcap \gamma'') = (\gamma \sqcup \gamma') \sqcap (\gamma \sqcup \gamma'')$ y $\gamma \sqcap (\gamma' \sqcup \gamma'') = (\gamma \sqcap \gamma') \sqcup (\gamma \sqcap \gamma'')$.

La propiedad más general que podemos enunciar acerca de dos restricciones de reemplazamiento comparables es la siguiente.

Proposición 15.3.2 (Monotonía de $\hookrightarrow$ respecto a $\sqsubseteq$) *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\gamma, \gamma' \in \Gamma$. Si $\gamma \sqsubseteq \gamma'$, entonces $\hookrightarrow_{\mathcal{R}(\gamma)} \subseteq \hookrightarrow_{\mathcal{R}(\gamma')}$.*

DEMOSTRACIÓN. Inmediata. □

15.3.1 Preservación de las propiedades básicas

Es fácil demostrar que los conjuntos Γ^{ROOT} y Γ^{DECOMP} son cerrados bajo las operaciones $\sqcup$ y $\sqcap$ del retículo de restricciones.

Proposición 15.3.3 *Si $\gamma, \gamma' \in \Gamma^{\mathsf{ROOT}}$, entonces $\gamma \sqcup \gamma' \in \Gamma^{\mathsf{ROOT}}$ y $\gamma \sqcap \gamma' \in \Gamma^{\mathsf{ROOT}}$. Si $\gamma, \gamma' \in \Gamma^{\mathsf{DECOMP}}$, entonces $\gamma \sqcup \gamma' \in \Gamma^{\mathsf{DECOMP}}$ y $\gamma \sqcap \gamma' \in \Gamma^{\mathsf{DECOMP}}$.*

DEMOSTRACIÓN. Respecto a ROOT, tenemos que, si $\gamma, \gamma' \in \Gamma^{\mathsf{ROOT}}$ entonces, para todo término t, $\epsilon \in O^{\gamma}(t)$ y $\epsilon \in O^{\gamma'}(t)$. Es decir, $\epsilon \in O^{\gamma}(t) \cup O^{\gamma'}(t)$ y $\epsilon \in O^{\gamma}(t) \cap O^{\gamma'}(t)$. Por tanto, $\epsilon \in O^{\gamma \sqcup \gamma'}(t)$, luego $\gamma \sqcup \gamma' \in \Gamma^{\mathsf{ROOT}}$ y $\epsilon \in O^{\gamma \sqcap \gamma'}(t)$, luego $\gamma \sqcap \gamma' \in \Gamma^{\mathsf{ROOT}}$.

Respecto a DECOMP seguiríamos un razonamiento parecido. □

El conjunto Γ^{COMP} es cerrado bajo la operación que computa el ínfimo del retículo.

Proposición 15.3.4 *Si $\gamma, \gamma' \in \Gamma^{\mathsf{COMP}}$, entonces $\gamma \sqcap \gamma' \in \Gamma^{\mathsf{COMP}}$.*

DEMOSTRACIÓN. Sea $v \in O(t)$. Si $u \in O^{\gamma \sqcap \gamma'}(C[t]_u)$, entonces $u \in O^{\gamma}(C[t]_u)$ y $u \in O^{\gamma'}(C[t]_u)$. Por COMP, $u.v \in O^{\gamma}(C[t]_u)$ y $u.v \in O^{\gamma'}(C[t]_u)$, es decir, $u.v \in O^{\gamma \sqcap \gamma'}(C[t]_u)$. □

Por desgracia, como mostraremos más adelante, Γ^{COMP} *no* es cerrado bajo la operación supremo del retículo. No obstante, podemos formular una propiedad de cierre más débil.

Proposición 15.3.5 *Sean* $\gamma, \gamma' \in \Gamma^{\mathsf{COMP}}$, $C \in \mathcal{T}(\Sigma, V)$ *y* $t \in \mathcal{T}(\Sigma, V)$. *Si* $u \in O^{\gamma \sqcap \gamma'}(C) \wedge v \in O^{\gamma \sqcup \gamma'}(t)$, *entonces* $u.v \in O^{\gamma \sqcup \gamma'}(C[t]_u)$.

DEMOSTRACIÓN. Si $u \in O^{\gamma \sqcap \gamma'}(C)$ y $v \in O^{\gamma \sqcup \gamma'}(t)$, entonces, aplicando la propiedad COMP tanto de γ como de γ', tenemos $u.v \in O^{\gamma}(C[t]_u)$ o bien $u.v \in O^{\gamma'}(C[t]_u)$. Luego $u.v \in O^{\gamma \sqcup \gamma'}(C[t]_u)$. □

Tanto CONTEXT como REPL son preservadas por las operaciones del retículo.

Proposición 15.3.6 *Si* $\gamma, \gamma' \in \Gamma^{\mathsf{CONTEXT}}$, *entonces* $\gamma \sqcup \gamma' \in \Gamma^{\mathsf{CONTEXT}}$ *y* $\gamma \sqcap \gamma' \in \Gamma^{\mathsf{CONTEXT}}$. *Si* $\gamma, \gamma' \in \Gamma^{\mathsf{REPL}}$, *entonces* $\gamma \sqcup \gamma' \in \Gamma^{\mathsf{REPL}}$ *y* $\gamma \sqcap \gamma' \in \Gamma^{\mathsf{REPL}}$.

15.3.2 Aproximación de restricciones de reemplazamiento

El análisis de las propiedades computacionales de la relación de reescritura restringida $\hookrightarrow_{\mathcal{R}(\gamma)}$ inducida (para un SRTE $\mathcal{R}$ dado) por una restricción de reemplazamiento γ descrita mediante las propiedades básicas de la Sección 15.2 puede ser dificultoso. Una posibilidad es *aproximar* Γ por otro conjunto de restricciones de reemplazamiento cuyas propiedades ya sean conocidas. Con ello, a cada restricción de reemplazamiento $\gamma \in \Gamma$, le corresponderá una restricción *aproximada* cuyas propiedades podremos relacionar con la restricción original.

Una aproximación trivial del conjunto Γ es $\{\gamma_\bot, \gamma_\top\}$: toda restricción $\gamma \in \Gamma$ viene acotada superiormente por $\gamma_\top$ e inferiormente por $\gamma_\bot$. De forma análoga, puesto que $\hookrightarrow_{\mathcal{R}(\gamma)}$ es una restricción de la reescritura, siempre puede ser aproximada superiormente por $\rightarrow_{\mathcal{R}} = \hookrightarrow_{\mathcal{R}(\gamma_\top)}$. Algunas propiedades computacionales de $\rightarrow_{\mathcal{R}}$ (por ejemplo, la terminación) serían 'heredadas' por $\hookrightarrow_{\mathcal{R}(\gamma)}$. Sin embargo, esto no resulta satisfactorio puesto que las mejoras en el comportamiento de terminación introducidas por una restricción de reemplazamiento γ no pueden ser capturadas mediante esta aproximación.

Un dominio de aproximación más adecuado es el conjunto de restricciones sensibles al contexto. Dada una función de reemplazamiento μ, definimos una restricción de reemplazamiento γ_μ como sigue:

Para todo $t \in \mathcal{T}(\Sigma, V)$, $O^{\gamma_\mu}(t) = O^{\mu}(t)$.

Ahora podemos definir el conjunto de restricciones de reemplazamiento sensibles al contexto Γ_{rsc} como sigue:

$$\Gamma_{rsc} = \{\gamma_\mu \mid \mu \in M_\Sigma\}$$

El retículo $(M_\Sigma, \sqsubseteq, \mu_\bot, \mu_\top, \sqcup, \sqcap)$ induce un retículo $(\Gamma_{rsc}, \sqsubseteq, \gamma_{\mu_\bot}, \gamma_{\mu_\top}, \sqcup_{rsc}, \sqcap_{rsc})$ que está incluido en $(\Gamma, \sqsubseteq, \gamma_\bot, \gamma_\top, \sqcup, \sqcap)$, pero *no* es un subretículo suyo. Esto se debe a que la operación supremo difiere de uno a otro retículo. Siendo $\gamma_\mu \sqcup_{rsc} \gamma_{\mu'} = \gamma_{\mu \sqcup \mu'}$ y $\gamma_\mu \sqcap_{rsc} \gamma_{\mu'} = \gamma_{\mu \sqcap \mu'}$, tenemos que $\gamma_\mu \sqcap_{rsc} \gamma_{\mu'} = \gamma_\mu \sqcap \gamma_{\mu'}$ (es decir, los ínfimos coinciden).

Proposición 15.3.7 *Si $\gamma_\mu, \gamma_{\mu'} \in \Gamma_{rsc}$, entonces $\gamma_\mu \sqcap_{rsc} \gamma_{\mu'} = \gamma_\mu \sqcap \gamma_{\mu'}$.*

DEMOSTRACIÓN. Demostramos, por inducción estructural, que, para todo término t, $O^{\gamma_\mu \sqcap_{rsc} \gamma_{\mu'}}(t) = O^{\gamma_\mu \sqcap \gamma_{\mu'}}(t)$. Si t es una variable o una constante, entonces $O^{\gamma_\mu \sqcap_{rsc} \gamma_{\mu'}}(t) = \{\epsilon\} = O^{\gamma_\mu}(t) \cap O^{\gamma_{\mu'}}(t) = O^{\gamma_\mu \sqcap \gamma_{\mu'}}(t)$. Si $t = f(\tilde{t})$, entonces $u = i.u' \in O^{\gamma_\mu \sqcap_{rsc} \gamma_{\mu'}}(t) \Leftrightarrow i.u' \in O^{\gamma_{\mu \sqcap \mu'}}(t)$, por definición de $\sqcap_{rsc}$. Tenemos que $i.u' \in O^{\gamma_{\mu \sqcap \mu'}}(t) \Leftrightarrow i \in \mu(f) \cap \mu'(f) \wedge u' \in O^{\gamma_{\mu \sqcap \mu'}}(t_i)$. Si $u' \in O^{\gamma_{\mu \sqcap \mu'}}(t_i)$, entonces, por la hipótesis de inducción, $u' \in O^{\gamma_\mu \sqcap \gamma_{\mu'}}(t_i)$. Por tanto, por definición de $\sqcap$, $i \in \mu(f) \wedge i \in \mu'(f) \wedge u' \in O^{\gamma_\mu \sqcap \gamma_{\mu'}}(t_i) \Leftrightarrow i \in \mu(f) \wedge i \in \mu'(f) \wedge u' \in O^{\gamma_\mu}(t_i) \wedge u' \in O^{\gamma_{\mu'}}(t_i) \Leftrightarrow i \in \mu(f) \wedge u' \in O^{\gamma_\mu}(t_i) \wedge i \in \mu'(f) \wedge u' \in O^{\gamma_{\mu'}}(t_i) \Leftrightarrow i.u' \in O^{\gamma_\mu}(t) \wedge i.u' \in O^{\gamma_{\mu'}}(t) \Leftrightarrow u \in O^{\gamma_\mu}(t) \cap O^{\gamma_{\mu'}}(t) \Leftrightarrow u \in O^{\gamma_\mu \sqcap \gamma_{\mu'}}(t)$. □

Sin embargo, la operación supremo no coincide en ambos retículos y, por tanto, aun siendo $\gamma_\mu, \gamma_{\mu'} \in \Gamma_{rsc}$, podría ser que $\gamma_\mu \sqcup \gamma_{\mu'} \notin \Gamma_{rsc}$.

Ejemplo 15.3.8 *Consideremos la signatura $\Sigma = \{\mathtt{f}, \mathtt{g}\}$, donde $ar(\mathtt{f}) = ar(\mathtt{g}) = 2$. Supongamos que $\mu(\mathtt{f}) = \mu(\mathtt{g}) = \{1\}$ y $\mu'(\mathtt{f}) = \mu'(\mathtt{g}) = \{2\}$. Sea el término $t = \mathtt{f}(\mathtt{g}(\mathtt{x}, \mathtt{y}), \mathtt{g}(\mathtt{x}, \mathtt{y}))$. Tenemos que*

$$O^{\gamma_\mu}(t) = \{\epsilon, 1, 1.1\} \text{ y } O^{\gamma_{\mu'}}(t) = \{\epsilon, 2, 2.2\}$$

Luego,

$$O^{\gamma_\mu \sqcup \gamma_{\mu'}}(t) = \{\epsilon, 1, 2, 1.1, 2.2\},$$

pero

$$O^{\gamma_\mu \sqcup_{rsc} \gamma_{\mu'}}(t) = O^{\gamma_{\mu \sqcup \mu'}}(t) = \{\epsilon, 1, 2, 1.1, 1.2, 2.1, 2.2\}.$$

En general, se satisface la siguiente propiedad.

Proposición 15.3.9 *Si $\gamma, \gamma' \in \Gamma_{rsc}$, entonces $\gamma_\mu \sqcup \gamma_{\mu'} \sqsubseteq \gamma_\mu \sqcup_{rsc} \gamma_{\mu'}$.*

DEMOSTRACIÓN. Dado que $\gamma_\mu, \gamma_{\mu'} \sqsubseteq \gamma_{\mu \sqcup \mu'} = \gamma_\mu \sqcup_{rsc} \gamma_{\mu'}$, por la Proposición 3.1.12 tenemos $O^{\gamma_\mu}(t) \subseteq O^{\gamma_\mu \sqcup_{csr} \gamma_{\mu'}}(t)$ y $O^{\gamma_{\mu'}}(t) \subseteq O^{\gamma_\mu \sqcup_{csr} \gamma_{\mu'}}(t)$, es decir, $O^{\gamma_\mu \sqcup \gamma_{\mu'}}(t) = O^{\gamma_\mu}(t) \cup O^{\gamma_{\mu'}}(t) \subseteq O^{\gamma_\mu \sqcup_{rsc} \gamma_{\mu'}}(t)$. Por tanto, $\gamma_\mu \sqcup \gamma_{\mu'} \sqsubseteq \gamma_\mu \sqcup_{rsc} \gamma_{\mu'}$. □

También es importante advertir que $\gamma_\top = \gamma_{\mu_\top}$ pero $\gamma_\bot \sqsubset \gamma_{\mu_\bot}$ (debido a que $\epsilon \in O^{\gamma_{\mu_\bot}}(t)$ para todo término t).

15.3.3 Aproximación de restricciones de reemplazamiento mediante restricciones sensibles al contexto

Podemos definir funciones de aproximación (parciales):

$$\alpha^{\sqsubseteq}, \alpha^{\sqsupseteq} : \Gamma \to \Gamma_{rsc}$$

para establecer aproximaciones sensibles al contexto (o simplemente aproximaciones *sc*) superiores e inferiores de una restricción de reemplazamiento γ, es decir:

$$\alpha^{\sqsubseteq}(\gamma) \sqsubseteq \gamma \sqsubseteq \alpha^{\sqsupseteq}(\gamma)$$

Como veremos en la Sección 15.4, estas aproximaciones serán útiles para establecer las propiedades computacionales de la reescritura restringida por comparación con las propiedades de la *rsc*.

Observación 15.3.10 *Puede suceder que $\alpha^{\sqsubseteq}(\gamma)$ no exista. Esto se debe a que, para toda μ, tenemos que $\epsilon \in O^{\gamma_\mu}(t)$. Por tanto, $\alpha^{\sqsubseteq}(\gamma)$ existe si y sólo si $\gamma \in \Gamma^{\mathsf{ROOT}}$. Por el contrario, $\alpha^{\sqsupseteq}$ siempre puede definirse como una función total puesto que $\gamma_\top = \gamma_{\mu_\top}$.*

A continuación definimos funciones de aproximación que, a cada restricción de reemplazamiento γ, le asocian restricciones sensibles al contexto $\gamma_{\mu^\uparrow}$ y $\gamma_{\mu^\downarrow}$ superiores e inferiores, respectivamente. Demostramos que éstas son las mejores aproximaciones *sc* que se pueden obtener. También analizamos la utilización de las operaciones del retículo para obtener una aproximación superior $\gamma^\uparrow$ más precisa que, aunque no es una restricción sensible al contexto, puede expresarse como combinación de restricciones sensibles al contexto.

Definición 15.3.11 *Sea $\gamma \in \Gamma(\Sigma)$. Definimos la restricción de reemplazamiento sensible al contexto $\gamma_{\mu^\downarrow}$ asociada a γ mediante la función de reemplazamiento $\mu^\downarrow$ como sigue: para todo $f \in \Sigma$*

$$\mu^\downarrow(f) = \{i \in \mathbb{N}^+_{ar(f)} \mid \forall f(\tilde{t}) \in \mathcal{T}(\Sigma, V), i \in O^\gamma(f(\tilde{t}))\}$$

Esta función de reemplazamiento sensible al contexto sólo acota inferiormente a γ (tratándose en ese caso de una aproximación inferior) si γ posee las propiedades ROOT y COMP.

Proposición 15.3.12 *Si $\gamma \in \Gamma^{\mathsf{ROOT+COMP}}$, entonces $\gamma_{\mu^\downarrow} \sqsubseteq \gamma$.*

Demostración. Dado un término t, demostramos, por inducción estructural, que $O^{\gamma_{\mu^\downarrow}}(t) \subseteq O^\gamma(t)$. Si t es una constante o una variable, por ROOT tenemos que $O^{\gamma_{\mu^\downarrow}}(t) = \{\epsilon\} = O^\gamma(t)$. Sea $t = f(\tilde{t})$ y $u = i.u' \in O^{\gamma_{\mu^\downarrow}}(t)$, para $i \in \mathbb{N}^+_{ar(f)}$. Dado que $i \in \mu^\downarrow(f)$, por definición de $\mu^\downarrow$ debe tenerse que $i \in O^\gamma(t)$. Por la hipótesis de inducción, $u' \in O^\gamma(t_i)$. Empleando COMP llegamos a la conclusión deseada. □

De acuerdo con la caracterización dada para γ_τ al final de la Sección 15.2.1, podemos acotar γ_τ mediante la función $\gamma_{\mu^\downarrow}$ correspondiente.

Ahora demostramos que, en general (sin necesidad de exigir que $\gamma \in \Gamma^{\mathsf{ROOT+COMP}}$), $\gamma_{\mu^\downarrow}$ es la mejor aproximación inferior *sc* que puede definirse.

Proposición 15.3.13 *Sea $\gamma \in \Gamma(\Sigma)$ y $\mu \in M_\Sigma$. Si $\mu \not\sqsubseteq \mu^\downarrow$, entonces $\gamma_\mu \not\sqsubseteq \gamma$.*

DEMOSTRACIÓN. Por el Lema 3.1.6, existe un símbolo $f \in \Sigma$ y un índice $i \in \mu(f)$ tales que $i \notin \mu^\downarrow(f)$. Si $i \notin \mu^\downarrow(f)$, entonces existe un término $t = f(\tilde{t})$ tal que $i \notin O^\gamma(t)$. Puesto que $i \in O^{\gamma_\mu}(t)$, tenemos $O^{\gamma_\mu}(t) \not\subseteq O^\gamma(t)$ y, por tanto, $\gamma_\mu \not\sqsubseteq \gamma$. □

Corolario 15.3.14 *Sea $\gamma \in \Gamma(\Sigma)$ y $\mu \in M_\Sigma$. Si $\gamma_\mu \sqsubseteq \gamma$, entonces $\gamma_\mu \sqsubseteq \gamma_{\mu^\downarrow}$.*

A continuación definimos una aproximación superior sensible al contexto y demostramos también que es la mejor que se puede definir.

Definición 15.3.15 *Sea $\gamma \in \Gamma(\Sigma)$. Definimos la restricción de reemplazamiento sensible al contexto $\gamma_{\mu^\uparrow}$ asociada a γ mediante la función de reemplazamiento $\mu^\uparrow$ como sigue: para todo $f \in \Sigma$*

$$\mu^\uparrow(f) = \{i \in \mathbb{N}^+_{ar(f)} \mid \exists C[f(\tilde{t})]_u \in \mathcal{T}(\Sigma, V), u.i \in O^\gamma(C[f(\tilde{t})]_u)\}$$

Para restricciones de reemplazamiento pertenecientes a Γ^{DOWN}, tenemos lo siguiente.

Proposición 15.3.16 *Si $\gamma \in \Gamma^{\mathsf{DOWN}}$, entonces $\gamma \sqsubseteq \gamma_{\mu^\uparrow}$.*

DEMOSTRACIÓN. Dado un término t, demostramos, por inducción sobre la longitud de $u \in O^\gamma(t)$, que $O^\gamma(t) \subseteq O^{\mu^\uparrow}(t)$. Si $|u| = 0$, entonces $u = \epsilon \in O^{\mu^\uparrow}(t)$. Si $|u| > 0$ consideremos $t = C[f(\tilde{t})]_{u'}$, con $u = u'.i$, $1 \leq i \leq ar(f)$. Por DOWN, $u' \in O^\gamma(C[f(\tilde{t})]_{u'})$. Por la hipótesis de inducción, $u' \in O^{\mu^\uparrow}(C[f(\tilde{t})]_{u'})$ y, por definición de $\mu^\uparrow$, $i \in \mu^\uparrow(f)$. Por tanto, $u \in O^{\mu^\uparrow}(C[f(\tilde{t})]_{u'}) = O^{\mu^\uparrow}(t)$. Esto concluye la demostración. □

De esta forma, tanto γ_Ω como γ_τ serían acotados por las $\gamma_{\mu^\uparrow}$ correspondientes. Ahora demostramos que esta es la mejor aproximación superior sensible al contexto que podemos obtener para una determinada restricción de reemplazamiento.

Proposición 15.3.17 *Sea $\gamma \in \Gamma(\Sigma)$ y $\mu \in M_\Sigma$. Si $\mu^\uparrow \not\sqsubseteq \mu$, entonces $\gamma \not\sqsubseteq \gamma_\mu$.*

DEMOSTRACIÓN. Por el Lema 3.1.6, existen $f \in \Sigma$ e $i \in \mu^\uparrow(f)$ tales que $i \notin \mu(f)$. Supongamos que $\gamma \sqsubseteq \gamma_\mu$. Debe tenerse $O^\gamma(t) \subseteq O^{\gamma_\mu}(t)$ para todo término t. Puesto que $i \in \mu^\uparrow(f)$, existe un término $t = C[f(\tilde{t})]_u$ tal que $u.i \in O^\gamma(t)$. Sin embargo, puesto que $i \notin \mu(f)$, $u.i \notin O^{\gamma_\mu}(t)$ lo cual nos lleva a una contradicción. Por tanto, $\gamma \not\sqsubseteq \gamma_\mu$. □

Corolario 15.3.18 *Sea $\gamma \in \Gamma(\Sigma)$ y $\mu \in M_\Sigma$. Si $\gamma \sqsubseteq \gamma_\mu$, entonces $\gamma_{\mu^\uparrow} \sqsubseteq \gamma_\mu$.*

La siguiente proposición establece la relación esperada entre las funciones de reemplazamiento $\mu^\uparrow$ y $\mu^\downarrow$ correspondientes a una restricción de reemplazamiento γ dada.

Proposición 15.3.19 *Para todo $\gamma \in \Gamma$, $\mu^\downarrow \sqsubseteq \mu^\uparrow$.*

DEMOSTRACIÓN. Inmediata, por definición de $\mu^\downarrow$ y $\mu^\uparrow$. □

Corolario 15.3.20 *Si $\gamma \in \Gamma^{\mathsf{ROOT+COMP+DOWN}}$, entonces $\gamma_{\mu^\downarrow} \sqsubseteq \gamma \sqsubseteq \gamma_{\mu^\uparrow}$.*

DEMOSTRACIÓN. Inmediata, empleando las Proposiciones 15.3.16 y 15.3.12. □

Con el siguiente ejemplo justificamos la necesidad de las condiciones impuestas sobre las restricciones consideradas para conseguir aproximarlas tal como se ha hecho.

Ejemplo 15.3.21 *Sea γ como sigue: si $t = \mathtt{h(g(f(a)))}$, $O^\gamma(t) = \{1.1, 1.1.1\}$ y $O^\gamma(s) = \{\epsilon\}$ para $s \neq t$. Tenemos:*

$$\mu^\downarrow(\mathtt{f}) = \mu^\downarrow(\mathtt{g}) = \mu^\downarrow(\mathtt{h}) = \emptyset,$$

$$\mu^\uparrow(\mathtt{f}) = \mu^\uparrow(\mathtt{g}) = \{1\} \text{ y } \mu^\uparrow(\mathtt{h}) = \emptyset.$$

Obsérvese que $\gamma \notin \Gamma^{\mathsf{DOWN}}$ y $\gamma \notin \Gamma^{\mathsf{COMP}}$. Es fácil ver que $\gamma_{\mu^\downarrow}$ no es una cota inferior de γ (pues $\epsilon \notin O^\gamma(t)$ pero $\epsilon \in O^{\gamma_{\mu^\downarrow}}(t)$) y también que $\gamma_{\mu^\uparrow}$ no es una cota superior (pues $1.1, 1.1.1 \notin O^{\gamma_{\mu^\uparrow}}(t)$).

En algunos casos, las funciones de reemplazamiento $\mu^\uparrow$ y $\mu^\downarrow$ asociadas a una restricción de reemplazamiento coinciden.

Proposición 15.3.22 *Si $\gamma \in \Gamma^{\mathsf{SUBTERM+REPL+CONTEXT}}$, entonces $\mu^\downarrow = \mu^\uparrow$.*

DEMOSTRACIÓN. Que $\mu^\downarrow \sqsubseteq \mu^\uparrow$ se sigue de la Proposición 15.3.19. Para demostrar que $\mu^\uparrow \sqsubseteq \mu^\downarrow$, supongamos que $i \in \mu^\uparrow(f)$. Entonces existe un término $t = C[f(\tilde{t})]_u$ tal que $u.i \in O^\gamma(t)$. Por SUBTERM, $i \in O^\gamma(f(\tilde{t}))$. Dados los términos s_j, $1 \leq j \leq ar(f)$, por CONTEXT, $i \in O^\gamma(f(\tilde{s}[t_i]_i))$ y, por REPL, $i \in O^\gamma(f(\tilde{s}[s_i]_i))$. Por tanto $i \in \mu^\downarrow(f)$. Esto concluye la demostración. □

15.3.4 Caracterización axiomática de las restricciones sensibles al contexto

En esta sección obtenemos una caracterización precisa del conjunto de restricciones sensibles al contexto Γ_{rsc} en términos de las propiedades básicas de la Sección 15.2.

Proposición 15.3.23 $\Gamma_{rsc} \subseteq \Gamma^{\mathsf{ROOT+COMP+DECOMP}}$.

DEMOSTRACIÓN. Dada una función de reemplazamiento $\mu \in M_\Sigma$, ROOT se satisface de forma inmediata por definición de γ_μ. Las propiedades COMP y DECOMP vienen expresadas por la Proposición 3.2.1. □

Proposición 15.3.24 *Si* $\gamma \in \Gamma^{\mathsf{ROOT+COMP+DECOMP}}$, *entonces* $\gamma_{\mu\downarrow} = \gamma_{\mu\uparrow} = \gamma \in \Gamma_{rsc}$.

DEMOSTRACIÓN. Por DECOMP, tenemos DOWN y, por el Corolario 15.3.20, $\gamma_{\mu\downarrow} \sqsubseteq \gamma \sqsubseteq \gamma_{\mu\uparrow}$. Por DECOMP, también tenemos SUBTERM y, por la Proposición 15.2.5 y la Proposición 15.3.22, concluimos el resultado deseado. □

El siguiente teorema proporciona una caracterización completa de las restricciones sensibles al contexto en términos de las propiedades de la Sección 15.2.

Teorema 15.3.25 $\Gamma_{rsc} = \Gamma^{\mathsf{ROOT+COMP+DECOMP}}$.

DEMOSTRACIÓN. Inmediata, empleando las Proposiciones 15.3.23 y 15.3.24. □

Observación 15.3.26 *Es interesante comprobar cómo la Proposición 3.2.1 y la propiedad* $\epsilon \in O^\mu(t)$ *para todo* t *caracterizan por completo las restricciones de reemplazamiento sensibles al contexto.*

15.3.5 Descomposición de restricciones de reemplazamiento

Dada una restricción de reemplazamiento γ, la aproximación superior $\gamma_{\mu\uparrow}$ presenta un invonveniente: crecerá rápidamente hasta alcanzar el valor $\gamma_\top$ en tanto que existan ocurrencias γ-reemplazantes de argumentos de un símbolo de función f situado sobre ocurrencias más y más profundas.

Por este motivo, es interesante investigar la posibilidad de obtención de aproximaciones más refinadas que caractericen γ con más precisión.

Debido a que las operaciones *supremo* de los retículos Γ y Γ_{rsc} no coinciden, y dado que la segunda acota superiormente a la primera (Proposición 15.3.9), el empleo del supremo $\sqcup$ del retículo $(\Gamma, \sqsubseteq, \gamma_\bot, \gamma_\top, \sqcup, \sqcap)$ sugiere una forma diferente de utilizar las restricciones sensibles al contexto para aproximar una restricción de reemplazamiento γ. Si podemos descomponer γ como $\gamma = \gamma_1 \sqcup \cdots \sqcup \gamma_n$, podemos aproximar cada componente γ_i, para cada $1 \leq i \leq n$ mediante su correspondiente aproximación $\gamma_{\mu_i\uparrow}$. Entonces, la restricción de reemplazamiento $\gamma^\uparrow = \gamma_{\mu_1\uparrow} \sqcup \cdots \sqcup \gamma_{\mu_n\uparrow}$ es una aproximación más precisa que $\gamma_{\mu\uparrow}$. En primer lugar, demostramos que $\gamma^\uparrow$ es una verdadera aproximación superior de γ.

Proposición 15.3.27 *Sea* $\gamma = \gamma_1 \sqcup \cdots \sqcup \gamma_n$ *tal que, para todo* j, $1 \leq j \leq n$, $\gamma_j \in \Gamma^{\mathsf{DOWN}}$. *Si* $\gamma^\uparrow = \gamma_{\mu_1\uparrow} \sqcup \cdots \sqcup \gamma_{\mu_n\uparrow}$, *entonces* $\gamma \sqsubseteq \gamma^\uparrow$.

DEMOSTRACIÓN. Por la Proposición 15.3.16, tenemos que $\gamma_j \sqsubseteq \gamma_{\mu_j^\uparrow}$ para todo j, $1 \leq j \leq n$. Por monotonía de $\sqcup$, se sigue la conclusión. □

Por desgracia, no podemos prescindir del requisito de que los componentes γ_j de la descomposición $\gamma = \gamma_1 \sqcup \cdots \sqcup \gamma_n$ satisfagan DOWN.

Ejemplo 15.3.28 *Consideremos la restricción de reemplazamiento γ definida como sigue: $O^\gamma(t) = \{\epsilon, 1, 2, 1.1, 2.2\}$ si $t =$* `f(g(x,y),g(x,y))` *y $O^\gamma(s) = \{\epsilon\}$ si $s \neq t$. Si definimos γ_1 y γ_2 como sigue:*

$$O^{\gamma_1}(t) = \{\epsilon, 2, 1.1\} \text{ y } O^{\gamma_2}(t) = \{\epsilon, 1, 2.2\},$$

y $O^{\gamma_1}(s) = O^{\gamma_2}(s) = \{\epsilon\}$ si $s \neq t$, entonces $\gamma = \gamma_1 \sqcup \gamma_2$. Las aproximaciones sc *superiores de γ_1 y γ_2 vienen dadas por $\mu_1^\uparrow$ y $\mu_2^\uparrow$ como sigue:*

$$\mu_1^\uparrow(\mathtt{f}) = \{2\}, \mu_1^\uparrow(\mathtt{g}) = \{1\} \quad y \quad \mu_2^\uparrow(\mathtt{f}) = \{1\}, \mu_2^\uparrow(\mathtt{g}) = \{2\}$$

Entonces, $O^{\gamma_{\mu_1^\uparrow}}(t) = \{\epsilon, 2, 2.1\}$ y $O^{\gamma_{\mu_2^\uparrow}}(t) = \{\epsilon, 1, 1.2\}$; por tanto,

$$O^{\gamma^\uparrow}(t) = \{\epsilon, 1, 2, 1.2, 2.1\}.$$

Esto significa que $\gamma \not\sqsubseteq \gamma^\uparrow$.

De acuerdo con el siguiente resultado, la propia restricción γ que se desea aproximar debe satisfacer DOWN.

Proposición 15.3.29 *Si $\gamma \in \Gamma$ es tal que $\gamma = \gamma_1 \sqcup \cdots \sqcup \gamma_n$ y $\gamma_i \in \Gamma^{\mathsf{DOWN}}$ para $1 \leq i \leq n$, entonces $\gamma \in \Gamma^{\mathsf{DOWN}}$.*

DEMOSTRACIÓN. Si $u.v \in O^\gamma(t)$, entonces $u.v \in O^{\gamma_i}(t)$ para algún $i \in \mathbb{N}_n^+$. Por DOWN, $u \in O^{\gamma_i}(t)$; es decir, $u \in O^\gamma(t)$. □

La restricción $\gamma^\uparrow$ *no* es, en general, una restricción sensible al contexto (el propio Ejemplo 15.3.28 demuestra esto), pero constituye una aproximación más precisa que $\gamma_{\mu^\uparrow}$. La siguiente proposición formaliza esta afirmación.

Proposición 15.3.30 *Sean $\gamma, \gamma_1, \ldots, \gamma_n \in \Gamma$ tales que $\gamma = \gamma_1 \sqcup \cdots \sqcup \gamma_n$ y sean $\gamma_{\mu_j^\uparrow}$, $1 \leq j \leq n$ las aproximaciones superiores* sc *para cada componente γ_j. Entonces, $\gamma_{\mu^\uparrow} = \gamma_{\mu_1^\uparrow} \sqcup_{rsc} \cdots \sqcup_{rsc} \gamma_{\mu_n^\uparrow}$.*

DEMOSTRACIÓN. Denotemos como $\tilde{\mu}^\uparrow$ la función de reemplazamiento $\mu_1^\uparrow \sqcup \cdots \sqcup \mu_n^\uparrow$. Entonces, por definición de $\sqcup_{rsc}$, $\gamma_{\tilde{\mu}^\uparrow} = \gamma_{\mu_1^\uparrow} \sqcup_{rsc} \cdots \sqcup_{rsc} \gamma_{\mu_n^\uparrow}$. Ahora demostramos que $\gamma_{\mu^\uparrow} \sqsubseteq \gamma_{\tilde{\mu}^\uparrow}$ y $\gamma_{\tilde{\mu}^\uparrow} \sqsubseteq \gamma_{\mu^\uparrow}$.

($\gamma_{\mu^\uparrow} \sqsubseteq \gamma_{\tilde{\mu}^\uparrow}$) Por inducción en la longitud de $u \in O^{\gamma_{\mu^\uparrow}}(t)$. El caso base, cuando $u = \epsilon$, es evidente. Si $u.i \in O^{\gamma_{\mu^\uparrow}}(C[f(\tilde{t})]_u)$, puesto que $i \in \mu^\uparrow(f)$, existe un término

$s = C'[f(\tilde{s})]_v$ tal que $v.i \in O^{\gamma}(s)$. Dado que $\gamma = \gamma_1 \sqcup \cdots \sqcup \gamma_n$, tenemos que $v.i \in O^{\gamma_j}(s)$ para algún j, $1 \leq j \leq n$ y, por tanto, $i \in \mu_j^{\uparrow}(f)$. Puesto que $u \in O^{\gamma_{\mu^\uparrow}}(C[f(\tilde{t})]_u)$, por la hipótesis de inducción, $u \in O^{\gamma_{\tilde{\mu}^\uparrow}}(C[f(\tilde{t})]_u)$. Puesto que $\mu_j^{\uparrow} \sqsubseteq \tilde{\mu}^{\uparrow}$, tenemos que $i \in \tilde{\mu}^{\uparrow}(f)$ y, así, $u.i \in O^{\gamma_{\tilde{\mu}^\uparrow}}(C[f(\tilde{t})]_u)$, lo cual completa la demostración.

($\gamma_{\tilde{\mu}^\uparrow} \sqsubseteq \gamma_{\mu^\uparrow}$) Similar. □

Observemos que la Proposición 15.3.30 indica que la aproximación *sc* superior a una restricción de reemplazamiento γ, cuando se calcula tal como se describe en la Definición 15.3.15, puede obtenerse a partir de *cualquier* descomposición de γ.

Observación 15.3.31 *Las Proposiciones 15.3.30 y 15.3.9 demuestran que $\gamma^{\uparrow}$ es mejor aproximación superior que $\gamma_{\mu^\uparrow}$, es decir $\gamma^{\uparrow} \sqsubseteq \gamma_{\mu^\uparrow}$. Sin embargo, a pesar de que puede obtenerse como una combinación de restricciones sensibles al contexto, $\gamma^{\uparrow}$ no lo es.*

Consideremos la aproximación *sc* superior $\gamma_{\mu^\uparrow}$ tal como se define al final del Ejemplo 15.3.8. Observemos que se trata de $\gamma_{\mu_\top} = \gamma_\top$. Sin embargo, si descomponemos γ como se hizo en el Ejemplo 15.3.8, obtenemos $\gamma^{\uparrow}$ que es una aproximación estrictamente mejor puesto que $\gamma^{\uparrow} \sqsubset \gamma_\top$.

De forma análoga, siempre que podamos descomponer γ como $\gamma = \gamma_1 \sqcap \cdots \sqcap \gamma_n$, podemos definir $\gamma^{\downarrow}$ como $\gamma^{\downarrow} = \gamma_{\mu_1^\downarrow} \sqcap \cdots \sqcap \gamma_{\mu_n^\downarrow}$. Se verifica la propiedad correspondiente para $\gamma_{\mu^\downarrow}$.

Proposición 15.3.32 *Sean $\gamma, \gamma_1, \ldots, \gamma_n \in \Gamma$ tales que $\gamma = \gamma_1 \sqcap \cdots \sqcap \gamma_n$ y sean $\gamma_{\mu_j^\downarrow}$, $1 \leq j \leq n$ las aproximaciones* sc *inferiores para cada componente γ_j. Entonces, $\gamma_{\mu^\downarrow} = \gamma_{\mu_1^\downarrow} \sqcap_{rsc} \cdots \sqcap_{rsc} \gamma_{\mu_n^\downarrow}$.*

Sin embargo, debido a que la Proposición 15.3.7 establece que $\sqcap$ y $\sqcap_{rsc}$ coinciden en Γ_{rsc}, no podemos obtener ningún refinamiento de la precisión de la aproximación si empleamos $\gamma^{\downarrow}$ en vez de $\gamma_{\mu^\downarrow}$. De hecho, $\gamma^{\downarrow} = \gamma_{\mu^\downarrow}$.

Los resultados anteriores justifican el interés de $\gamma^{\uparrow}$ como una buena aproximación superior de γ. Sin embargo, dado que la definición de $\gamma^{\uparrow}$ depende de una descomposición $\gamma = \gamma_1 \sqcup \cdots \sqcup \gamma_n$ de γ, y puesto que existen múltiples formas de realizar dicha descomposición, nos podemos preguntar si $\gamma^{\uparrow}$ está bien definida. Es decir, ¿depende la definición de $\gamma^{\uparrow}$ de la descomposición particular de γ que hayamos escogido? Por desgracia, la respuesta es afirmativa y los Ejemplos 15.3.8 y 15.3.28 son una muestra de ello. Por tanto, en el caso general, empezando con una factorización arbitraria de γ, no hay forma razonable de definir $\gamma^{\uparrow}$. Incluso con descomposiciones cuyos factores satisfagan DOWN.

Ejemplo 15.3.33 *Consideremos la restricción de reemplazamiento γ del Ejemplo 15.3.28. Podemos descomponerla como sigue:*

$$\gamma = \gamma_1 \sqcup \gamma_2,$$

donde

$$O^{\gamma_1}(t) = \{\epsilon, 1, 1.1\} \text{ y } O^{\gamma_2}(t) = \{\epsilon, 2, 2.2\}$$

(siendo t como en el Ejemplo 15.3.28) y $O^{\gamma_1}(s) = O^{\gamma_2}(s) = \{\epsilon\}$ para $s \neq t$. Esta descomposición satisface que $\gamma_1, \gamma_2 \in \Gamma^{\mathsf{DOWN}}$. En este caso,

$$\mu_1^{\uparrow}(\mathtt{f}) = \mu_1^{\uparrow}(\mathtt{g}) = \{1\} \quad y \quad \mu_2^{\uparrow}(\mathtt{f}) = \mu_2^{\uparrow}(\mathtt{g}) = \{2\},$$

luego $\gamma_1^{\uparrow}(t) = \{\epsilon, 1, 2, 1.1, 2.2\}$.

Por la Proposición 15.3.29, $\gamma \in \Gamma^{\mathsf{DOWN}}$ y, trivialmente, $\gamma_{\bot} \in \Gamma^{\mathsf{DOWN}}$. Luego $\gamma = \gamma \sqcup \gamma_{\bot}$ es otra "descomposición DOWN *" de γ. En este caso,*

$$\mu_1^{\uparrow}(\mathtt{f}) = \mu_1^{\uparrow}(\mathtt{g}) = \{1, 2\} \quad y \quad \mu_2^{\uparrow}(\mathtt{f}) = \mu_2^{\uparrow}(\mathtt{g}) = \emptyset$$

con lo cual $\gamma_2^{\uparrow}(t) = \{\epsilon, 1, 2, 1.1, 1.2, 2.1, 2.2\}$.

Claramente, $\gamma_1^{\uparrow} \sqsubset \gamma_2^{\uparrow}$.

Sin embargo, la situación es distinta para $\gamma_{\mu^{\uparrow}}$ cuya definición no depende de la factorización particular de γ, tal como demuestra la Proposición 15.3.30. También insistimos en que la Proposición 15.3.27 permanece válida en cuanto que establece las propiedades de aproximación de $\gamma^{\uparrow}$ (dada una factorización particular de γ). Afortunadamente, como veremos más adelante, estas aproximaciones superiores $\gamma_1^{\uparrow}, \ldots, \gamma_n^{\uparrow}$ de una restricción de reemplazamiento γ son cerradas bajo la operación $\sqcap$; es decir, $\gamma^{\uparrow} = \gamma_1^{\uparrow} \sqcap \cdots \sqcap \gamma_n^{\uparrow}$ es una (mejor) cota superior de γ que cada una de ellas por separado.

15.3.6 Un dominio de aproximación más preciso

Tras la discusión anterior, parece obvio que, mediante combinaciones de restricciones sensibles al contexto, podemos aproximar las restricciones de reemplazamiento de forma más adecuada. Por tanto, es interesante estudiar en qué medida podemos combinar las restricciones pertenecientes a Γ_{rsc} mediante las operaciones del retículo, con el fin de obtener nuevas restricciones que no pertenecen a Γ_{rsc}. De esta forma, podemos definir un nuevo subretículo $\Gamma_{rsc}^{\sqcup\sqcap}$ de Γ que contiene Γ_{rsc} (pero que es todavía finito). Sea $\Gamma_{rsc}^{\sqcup\sqcap}$ el conjunto

$$\Gamma_{rsc}^{\sqcup\sqcap} = \{\gamma \in \Gamma \mid \exists M_1, \ldots, M_m \subseteq M_{\Sigma}.\ \gamma = \sqcup_{1 \leq j \leq m} \sqcap_{\mu \in M_j} \gamma_{\mu}\}.$$

Siendo el retículo Γ distributivo y completo, es sencillo demostrar que $\Gamma_{rsc}^{\sqcup\sqcap}$ expresa realmente todas las posibles restricciones de reemplazamiento que pueden obtenerse mediante combinación de restricciones sensibles al contexto.

Sin embargo, podemos dar una formulación equivalente, pero más simple, de $\Gamma_{rsc}^{\sqcup\sqcap}$.

Definición 15.3.34 *Sea Σ una signatura. Sea $\Gamma_{rsc}^{\sqcup}$ el conjunto*

$$\Gamma_{rsc}^{\sqcup} = \{\gamma \in \Gamma \mid \exists \mu_1, \ldots, \mu_n \in M_{\Sigma}.\ \gamma = \sqcup_{1 \leq j \leq n} \gamma_{\mu_j}\}$$

Empleando la Proposición 15.3.7, no es difícil demostrar lo siguiente:

Proposición 15.3.35 $\Gamma_{rsc}^{\sqcup\sqcap} = \Gamma_{rsc}^{\sqcup}$

DEMOSTRACIÓN.

$$\begin{aligned}
\Gamma_{rsc}^{\sqcup\sqcap} &= \{\gamma \in \Gamma \mid \exists M_1, \ldots, M_m \subseteq M_\Sigma .\ \gamma = \sqcup_{1\leq j\leq m} \sqcap_{\mu\in M_j} \gamma_\mu\} \\
&= \{\gamma \in \Gamma \mid \exists M_1, \ldots, M_m \subseteq M_\Sigma .\ \gamma = \sqcup_{1\leq j\leq m} \gamma_{\sqcap M_j}\} \\
&= \{\gamma \in \Gamma \mid \exists M_1, \ldots, M_m \subseteq M_\Sigma . \\
&\qquad\qquad \forall j\ 1 \leq j \leq m . \mu_j = \sqcap M_j \wedge \gamma = \sqcup_{1\leq j\leq m} \gamma_{\mu_j}\} \\
&= \Gamma_{rsc}^{\sqcup}
\end{aligned}$$

□

Por tanto, si deseamos salirnos del retículo Γ_{rsc} empleando combinaciones de las operaciones del retículo Γ, es suficiente con utilizar el supremo del retículo; al utilizar el ínfimo no obtenemos nada más.

Proposición 15.3.36 $(\Gamma_{rsc}^{\sqcup}, \sqsubseteq, \gamma_{\mu_\bot}, \gamma_{\mu_\top}, \sqcup, \sqcap)$ *es un subretículo de* $(\Gamma, \sqsubseteq, \gamma_\bot, \gamma_\top, \sqcup, \sqcap)$. *Si* Γ_{rsc} *es finito, entonces* $\Gamma_{rsc}^{\sqcup}$ *también lo es.*

DEMOSTRACIÓN. Por definición, $\Gamma_{rsc}^{\sqcup}$ $(= \Gamma_{rsc}^{\sqcup\sqcap})$ es cerrado bajo $\sqcup$ y $\sqcap$, es decir, es un subretículo de Γ. Además, Γ_{rsc} está en correspondencia biunívoca con M_Σ. Si Γ_{rsc} es finito, también M_Σ lo es, luego el número de combinaciones posibles de funciones de reemplazamiento en M_Σ es finito, de donde se sigue que $\Gamma_{rsc}^{\sqcup}$ también es finito. □

Obviamente, $\Gamma_{rsc} \subseteq \Gamma_{rsc}^{\sqcup}$. Sin embargo los ejemplos anteriores muestran que $\Gamma_{rsc}^{\sqcup}$ es un superconjunto *propio* de Γ_{rsc}.

¿Qué podemos afirmar acerca de las propiedades de $\Gamma_{rsc}^{\sqcup}$? Tratándose de un subretículo, $\Gamma_{rsc}^{\sqcup}$ es cerrado bajo $\sqcup$ y $\sqcap$. También podemos concluir lo siguiente.

Proposición 15.3.37 $\Gamma_{rsc}^{\sqcup} \subseteq \Gamma^{\mathsf{ROOT+DECOMP+CONTEXT+REPL}}$.

DEMOSTRACIÓN. Por el Teorema 15.3.25 y la Proposición 15.2.9, tenemos que $\Gamma_{rsc} \subseteq \Gamma^{\mathsf{ROOT+DECOMP+CONTEXT+REPL}}$. De acuerdo con la Definición 15.3.34 y las Proposiciones 15.3.3 y 15.3.6, concluimos el resultado. □

Sin embargo, la propiedad COMP no resulta preservada por la operación $\sqcup$ y ello significa que no podemos asegurar que $\Gamma_{rsc}^{\sqcup} \subseteq \Gamma^{\mathsf{COMP}}$, como demuestra el siguiente ejemplo.

Ejemplo 15.3.38 *Consideremos el término* $t =$ `f(g(x,y),g(x,y))` *y definamos* $\mu(\mathtt{f}) = \mu(\mathtt{g}) = \{1\}$ *y* $\mu'(\mathtt{f}) = \{2\}$, $\mu'(\mathtt{g}) = \emptyset$. *Sea* $\gamma = \gamma_\mu \sqcup \gamma_{\mu'} \in \Gamma_{rsc}^{\sqcup}$. *Observemos que* $O^\gamma(t) = \{\epsilon, 1, 2, 1.1\}$. *Sea* $t' = t[t]_2$. *Entonces, a pesar de que* $2 \in O^\gamma(t)$ *y* $1.1 \in O^\gamma(t)$, *tenemos que* $2.1.1 \notin O^\gamma(t') = \{\epsilon, 1, 2, 1.1, 2.2\}$. *Por tanto* $\gamma \notin \Gamma^{\mathsf{COMP}}$.

Observemos que la Proposición 15.3.5 implica una forma más débil de la propiedad COMP al considerar $\Gamma^{\sqcup}_{rsc}$.

Proposición 15.3.39 *Sea* $\gamma = \gamma_{\mu_1} \sqcup \cdots \sqcup \gamma_{\mu_n} \in \Gamma^{\sqcup}_{rsc}$ *y* $\mu = \mu_1 \sqcap \cdots \sqcap \mu_n$. *Sea* $C \in \mathcal{T}(\Sigma, V)$, $t \in \mathcal{T}(\Sigma, V)$. *Si* $u \in O^{\gamma_\mu}(C) \wedge v \in O^{\gamma}(t)$, *entonces* $u.v \in O^{\gamma}(C[t]_u)$.

DEMOSTRACIÓN. Inmediata, empleando las Proposiciones 15.3.5 y 15.3.7. □

El siguiente resultado es una consecuencia inmediata del hecho de que $\Gamma^{\sqcup}_{rsc}$ es un subretículo de Γ.

Proposición 15.3.40 *Sean* $\gamma \in \Gamma$ *y* $\gamma_\uparrow, \gamma'_\uparrow \in \Gamma^{\sqcup}_{rsc}$. *Si* $\gamma \sqsubseteq \gamma_\uparrow$ *y* $\gamma \sqsubseteq \gamma'_\uparrow$, *entonces* $\gamma \sqsubseteq \gamma_\uparrow \sqcap \gamma'_\uparrow \in \Gamma^{\sqcup}_{rsc}$.

DEMOSTRACIÓN. Siendo γ una cota inferior tanto de $\gamma_\uparrow$ como de $\gamma'_\uparrow$, por definición de $\sqcap$, $\gamma_\uparrow \sqcap \gamma'_\uparrow$ es la mayor cota inferior de $\{\gamma_\uparrow, \gamma'_\uparrow\}$, luego $\gamma \sqsubseteq \gamma_\uparrow \sqcap \gamma'_\uparrow$. Claramente, $\gamma_\uparrow \sqcap \gamma'_\uparrow \in \Gamma^{\sqcup\sqcap}_{rsc}$ y, por la Proposición 15.3.35, $\gamma_\uparrow \sqcap \gamma'_\uparrow \in \Gamma^{\sqcup}_{rsc}$. □

Observación 15.3.41 *La Proposición 15.3.40 sugiere un método para obtener la mejor aproximación superior* $\gamma^\uparrow \in \Gamma^{\sqcup}_{rsc}$ *para una determinada restricción de reemplazamiento* γ*: realizamos tantas descomposiciones de* γ *como sea posible de forma que los factores verifiquen* DOWN. *Obtendremos un conjunto* $\Gamma^{\uparrow}_{\gamma}$ *de aproximaciones superiores de* γ. *El ínfimo de* $\Gamma^{\uparrow}_{\gamma}$ *es* $\gamma^\uparrow$, *la mejor de todas.*

15.3.7 Más operaciones sobre restricciones de reemplazamiento

Si empleamos $\sqcup$ en lugar de $\sqcup_{rsc}$ sobre restricciones sensibles al contexto, perdemos la propiedad COMP. Esto tiene consecuencias bastante negativas que podemos evitar añadiendo una nueva operación a las operaciones del retículo.

Definición 15.3.42 *Definimos la operación binaria* $\sqcup_c$ *sobre restricciones de reemplazamiento como sigue: Sean* $\gamma, \gamma' \in \Gamma$.

$$O^{\gamma \sqcup_c \gamma'}(t) = \{u.v \in O(t) \mid u \in O^{\gamma}(t) \wedge v \in O^{\gamma'}(t|_u)\}$$

Propiedades algebraicas elementales del operador $\sqcup_c$

Claramente, $\sqcup_c$ es un operador monótono.

Proposición 15.3.43 (Monotonía de $\sqcup_c$) *Sean* $\gamma_1, \gamma'_1, \gamma_2, \gamma'_2 \in \Gamma$. *Si* $\gamma_1 \sqsubseteq \gamma'_1$ *y* $\gamma_2 \sqsubseteq \gamma'_2$, *entonces* $\gamma_1 \sqcup_c \gamma_2 \sqsubseteq \gamma'_1 \sqcup_c \gamma'_2$.

La operación $\sqcup_c$ es asociativa:

Proposición 15.3.44 (Asociatividad de $\sqcup_c$) *Si $\gamma, \gamma', \gamma'' \in \Gamma$, entonces*

$$\gamma \sqcup_c (\gamma' \sqcup_c \gamma'') = (\gamma \sqcup_c \gamma') \sqcup_c \gamma''.$$

DEMOSTRACIÓN.

$$\begin{aligned} O^{\gamma \sqcup_c (\gamma' \sqcup_c \gamma'')}(t) &= \{u.v \in O(t) \mid u \in O^{\gamma}(t) \wedge v \in O^{\gamma' \sqcup_c \gamma''}(t|_u)\} \\ &= \{u.v'.w \in O(t) \mid u \in O^{\gamma}(t) \wedge v' \in O^{\gamma'}(t|_u) \wedge w \in O^{\gamma''}(t|_{u.v'})\} \\ &= \{u.v'.w \in O(t) \mid u.v' \in O^{\gamma \sqcup_c \gamma'}(t) \wedge w \in O^{\gamma''}(t|_{u.v'})\} \\ &= O^{(\gamma \sqcup_c \gamma') \sqcup_c \gamma''}(t) \end{aligned}$$

Esto completa la demostración. □

También se verifican las siguientes propiedades.

Proposición 15.3.45 *Sean $\gamma, \gamma' \in \Gamma$. Si $\gamma \in \Gamma^{\mathsf{ROOT}}$, entonces $\gamma' \sqsubseteq \gamma \sqcup_c \gamma'$ y $\gamma' \sqsubseteq \gamma' \sqcup_c \gamma$.*

DEMOSTRACIÓN. Dado que $\gamma \in \Gamma^{\mathsf{ROOT}}$, $\epsilon \in O^{\gamma}(t)$ para todo término t. Luego para todo $u \in O^{\gamma'}(t)$, tenemos que $O^{\gamma'}(t) \subseteq O^{\gamma \sqcup_c \gamma'}(t)$; es decir, $\gamma' \sqsubseteq \gamma \sqcup_c \gamma'$. De forma similar concluimos la otra implicación. □

El operador $\sqcup_c$, cuando actúa sobre restricciones de reemplazamiento pertenecientes a Γ^{ROOT}, siempre produce una restricción que acota superiormente a la obtenida mediante $\sqcup$.

Proposición 15.3.46 *Si $\gamma, \gamma' \in \Gamma^{\mathsf{ROOT}}$, entonces $\gamma \sqcup \gamma' \sqsubseteq \gamma \sqcup_c \gamma'$.*

DEMOSTRACIÓN. Por la Proposición 15.3.45, $\gamma \sqsubseteq \gamma \sqcup_c \gamma'$ y $\gamma' \sqsubseteq \gamma \sqcup_c \gamma'$, es decir, $\gamma \sqcup_c \gamma'$ es una cota superior tanto de γ como de γ'. Puesto que $\gamma \sqcup \gamma'$ es la menor de entre las cotas superiores de γ y γ', se sigue la conclusión. □

En general, $\sqcup_c$ *no* es conmutativa ni idempotente. Sin embargo, tenemos la siguiente propiedad.

Proposición 15.3.47 *Si $\gamma \in \Gamma^{\mathsf{ROOT+COMP}}$, entonces $\gamma \sqcup_c \gamma = \gamma$.*

DEMOSTRACIÓN. Por la Proposición 15.3.45, $\gamma \sqsubseteq \gamma \sqcup_c \gamma$. Sea $u.v \in O^{\gamma \sqcup_c \gamma}(t)$ tal que $u \in O^{\gamma}(t)$ y $v \in O^{\gamma}(t|_u)$. Por COMP, $u.v \in O^{\gamma}(t)$. Por tanto, $O^{\gamma \sqcup_c \gamma}(t) \subseteq O^{\gamma}(t)$, o sea, $\gamma \sqcup_c \gamma \sqsubseteq \gamma$. Esto concluye la demostración. □

Preservación de las propiedades básicas

La operación $\sqcup_c$ preserva ROOT y REPL.

Proposición 15.3.48 *Si* $\gamma, \gamma' \in \Gamma^{\mathsf{ROOT}}$ *(*$\in \Gamma^{\mathsf{REPL}}$*), entonces* $\gamma \sqcup_c \gamma' \in \Gamma^{\mathsf{ROOT}}$ *(*$\in \Gamma^{\mathsf{REPL}}$*).*

DEMOSTRACIÓN. Si $\gamma, \gamma' \in \Gamma^{\mathsf{ROOT}}$, por la Proposición 15.3.3, $\gamma \sqcup \gamma' \in \Gamma^{\mathsf{ROOT}}$. Puesto que, por la Proposición 15.3.46, $\gamma \sqcup \gamma' \sqsubseteq \gamma \sqcup_c \gamma'$, obtenemos que $\gamma \sqcup_c \gamma' \in \Gamma^{\mathsf{ROOT}}$.

Si $\gamma, \gamma' \in \Gamma^{\mathsf{REPL}}$, consideremos $u.v \in O^{\gamma \sqcup_c \gamma'}(C[t]_{u.v})$ tal que $u \in O^{\gamma}(C[t]_{u.v})$ y $v \in O^{\gamma'}(C[t]_{u.v}|_u)$. Por REPL, $u \in O^{\gamma}(C[s]_{u.v})$ y $v \in O^{\gamma'}(C[s]_{u.v}|_u)$, lo que significa que $u.v \in O^{\gamma \sqcup_c \gamma'}(C[s]_{u.v})$. □

Sin embargo, DECOMP y CONTEXT no son preservadas por la operación $\sqcup_c$. No obstante, se preservan (de forma combinada) algunas propiedades interesantes.

Proposición 15.3.49 *Si* $\gamma \in \Gamma^{\mathsf{DOWN}}$ *y* $\gamma' \in \Gamma^{\mathsf{ROOT+DOWN}}$*, entonces* $\gamma \sqcup_c \gamma' \in \Gamma^{\mathsf{DOWN}}$*.*

DEMOSTRACIÓN. Sea $u.v \in O^{\gamma \sqcup_c \gamma'}(C[t]_u)$ tal que $u \in O^{\gamma}(C[t]_u)$ y $v \in O^{\gamma'}(t)$. Si tomamos $u.v'$ con $v' < v$, entonces, por DOWN, tenemos que $v' \in O^{\gamma'}(t)$. Por tanto, $u.v' \in O^{\gamma \sqcup_c \gamma'}(C[t]_u)$. Se consideramos $u' < u$, entonces, empleando ROOT, tenemos que $\epsilon \in O^{\gamma'}(C[t]_u|_{u'})$. Por tanto $u' \in O^{\gamma \sqcup_c \gamma'}(C[t]_u)$. □

Proposición 15.3.50 *Si* $\gamma, \gamma' \in \Gamma^{\mathsf{ROOT+SUBTERM}}$*, entonces* $\gamma \sqcup_c \gamma' \in \Gamma^{\mathsf{SUBTERM}}$*.*

DEMOSTRACIÓN. Sea $u.v \in O^{\gamma \sqcup_c \gamma'}(C[t]_u)$ tal que $u \in O^{\gamma}(C[t]_u)$ y $v \in O^{\gamma'}(t)$. Si $v = v'.w$, por SUBTERM tenemos $w \in O^{\gamma'}(t|_{v'})$. Puesto que $\gamma \in \Gamma^{\mathsf{ROOT}}$, $w \in O^{\gamma \sqcup_c \gamma'}(t|_{v'})$. Supongamos que $u = u'.w$. Puesto que $v \in O^{\gamma'}(t)$ y, por SUBTERM, se tiene $w \in O^{\gamma}(C[t]_u|_{u'})$ y, dado que $\gamma' \in \Gamma^{\mathsf{ROOT}}$, tenemos $w.v \in O^{\gamma \sqcup_c \gamma'}(C[t]_u|_{u'})$. □

Por tanto, $\sqcup_c$ preserva ROOT+DECOMP.

Teorema 15.3.51 *Si* $\gamma, \gamma' \in \Gamma^{\mathsf{ROOT+DECOMP}}$*, entonces* $\gamma \sqcup_c \gamma' \in \Gamma^{\mathsf{ROOT+DECOMP}}$*.*

También se verifica una propiedad similar para CONTEXT y REPL.

Proposición 15.3.52 *Si* $\gamma \in \Gamma^{\mathsf{CONTEXT+REPL}}$ *y* $\gamma' \in \Gamma^{\mathsf{CONTEXT}}$*, entonces* $\gamma \sqcup_c \gamma' \in \Gamma^{\mathsf{CONTEXT}}$*.*

DEMOSTRACIÓN. Si $u.v \in O^{\gamma \sqcup_c \gamma'}(C)$, podemos asumir que $u \in O^{\gamma}(C) \wedge v \in O^{\gamma'}(C|_u)$. Sea $v' \in O(C)$ y $u.v \parallel v'$. Debemos considerar dos casos: $u \parallel v'$ y $u < v'$.

Si $u \parallel v'$, entonces, puesto que $u \in O^{\gamma}(C)$, por CONTEXT, $u \in O^{\gamma}(C[t]_{v'})$ y $u.v \in O^{\gamma \sqcup_c \gamma'}(C[t]_{v'})$.

Si $u < v'$ por REPL, $u \in O^{\gamma}(C[t]_{v'})$. Por tanto $v' = u.w$ y $w \parallel v$. Entonces, por CONTEXT, $v \in O^{\gamma'}(C[t]_{v'}|_u)$. Por tanto, $u.v \in O^{\gamma \sqcup_c \gamma'}(C[t]_{v'})$. □

De esta forma, $\sqcup_c$ preserva CONTEXT+REPL.

Teorema 15.3.53 *Si* $\gamma, \gamma' \in \Gamma^{\mathsf{CONTEXT+REPL}}$, *entonces* $\gamma \sqcup_c \gamma' \in \Gamma^{\mathsf{CONTEXT+REPL}}$

La operación $\sqcup_c$ y las restricciones de reemplazamiento sensibles al contexto

Siempre que consideremos combinaciones de restricciones de reemplazamiento en Γ_{rsc}, se verifica lo siguiente.

Proposición 15.3.54 *Si* $\gamma, \gamma' \in \Gamma_{rsc}$, *entonces* $\gamma \sqcup_c \gamma' \sqsubseteq \gamma \sqcup_{rsc} \gamma'$.

DEMOSTRACIÓN. Sean μ y μ' las funciones de reemplazamiento correspondientes a γ y γ' respectivamente; es decir, $\gamma = \gamma_\mu$ y $\gamma' = \gamma_{\mu'}$. Sea $u.v \in O^{\gamma_\mu \sqcup_c \gamma_{\mu'}}$ tal que $u \in O^{\gamma_\mu}(t)$ y $v \in O^{\gamma_{\mu'}}(t|_u)$. Luego $u \in O^{\gamma_{\mu \sqcup \mu'}}(t)$ y $v \in O^{\gamma_{\mu \sqcup \mu'}}(t|_u)$. Por la Proposición 3.2.1, $u.v \in O^{\gamma_{\mu \sqcup \mu'}}(t) = O^{\gamma_\mu \sqcup_{rsc} \gamma_{\mu'}}(t)$. □

Mediante un ejemplo mostramos que, en general, las operaciones $\sqcup_c$ y $\sqcup_{rsc}$ difieren cuando se aplican sobre restricciones sensibles al contexto.

Ejemplo 15.3.55 *Sean* μ *y* μ' *dos funciones de reemplazamiento tales que* $\mu(\mathtt{f}) = \{1\}$, $\mu(\mathtt{g}) = \emptyset$ *y* $\mu'(\mathtt{f}) = \emptyset$, $\mu'(\mathtt{g}) = \{1\}$. *Consideremos el término* $t = \mathtt{f(g(f(x)))}$. *Se tiene que* $1.1.1 \in O^{\gamma_\mu \sqcup_{rsc} \gamma_{\mu'}}(t)$ *pero*

- $\epsilon \in O^{\gamma_\mu}(\mathtt{f(g(f(x)))})$ *mientras que* $1.1.1 \notin O^{\gamma_{\mu'}}(\mathtt{f(g(f(x)))})$.
- $1 \in O^{\gamma_\mu}(\mathtt{f(g(f(x)))})$ *mientras que* $1.1 \notin O^{\gamma_{\mu'}}(\mathtt{g(f(x))})$.
- $1.1, 1.1.1 \notin O^{\gamma_\mu}(t)$.

Luego $1.1.1 \notin O^{\gamma \sqcup_c \gamma'}(t)$.

El ejemplo anterior demuestra que la operación $\sqcup_{rsc}$ es mucho más potente (en el sentido de 'crear' nuevas posibilidades de reemplazamiento) que $\sqcup_c$.

Por tanto, podemos definir un conjunto $\Gamma_{rsc}^{\sqcup_c}$ en el que combinaremos restricciones sensibles al contexto mediante la operación $\sqcup_c$.

Definición 15.3.56 *Sea* Σ *una signatura. El conjunto* $\Gamma_{rsc}^{\sqcup_c}$ *se define como sigue:*

$$\Gamma_{rsc}^{\sqcup_c} = \{\gamma \in \Gamma \mid \exists \mu_1, \ldots, \mu_n \in M_\Sigma.\ \gamma = \gamma_{\mu_1} \sqcup_c \cdots \sqcup_c \gamma_{\mu_n}\}$$

Claramente, $\Gamma_{rsc} \subseteq \Gamma_{rsc}^{\sqcup_c}$. Los resultados anteriores nos permiten caracterizar este conjunto como sigue.

Teorema 15.3.57 $\Gamma^{\sqcup_c}_{rsc} \subseteq \Gamma^{\mathsf{ROOT+DECOMP+CONTEXT+REPL}}$.

DEMOSTRACIÓN. Por el Teorema 15.3.25 y la Proposición 15.2.9, tenemos que $\Gamma_{rsc} \subseteq \Gamma^{\mathsf{ROOT+DECOMP+CONTEXT+REPL}}$. Por definición de $\Gamma^{\sqcup_c}_{rsc}$ y las propiedades de preservación de ROOT+DECOMP y CONTEXT+REPL para $\sqcup_c$ (Teoremas 15.3.51 y 15.3.53), concluimos el resultado deseado. □

También podemos considerar el conjunto

$$\Gamma^{\sqcup\sqcup_c}_{rsc} = \{\gamma \in \Gamma \mid \exists \gamma_1, \ldots, \gamma_n \in \Gamma^{\sqcup}_{rsc}.\ \gamma = \gamma_1 \sqcup_c \cdots \sqcup_c \gamma_n\}$$

para el cual se verifica que $\Gamma^{\sqcup\sqcup_c}_{rsc} \subseteq \Gamma^{\mathsf{ROOT+DECOMP+CONTEXT+REPL}}$.

Recuperación de la propiedad COMP mediante $\sqcup_c$

La operación $\sqcup_c$ puede utilizarse para 'introducir' la propiedad COMP en una restricción de reemplazamiento γ.

Definición 15.3.58 *Sea $\gamma \in \Gamma$. Definimos la operación $^+ : \Gamma \to \Gamma$ como sigue:*

$$\gamma^+ = \sqcup_{n>0} \gamma^n$$

donde

$$\gamma^1 = \gamma \text{ y, para todo } n > 1, \gamma^n = \gamma \sqcup_c \gamma^{n-1},$$

Observemos que la descripción de la operación $^+$ dada en la Definición 15.3.58 es consistente puesto que, siendo Γ un retículo completo, el conjunto $\{\gamma^1, \ldots, \gamma^n, \ldots\}$ tiene un supremo que es γ^+.

Proposición 15.3.59 *Para todo $p, q \in \mathbb{N}^+$, $\gamma^p \sqcup_c \gamma^q = \gamma^{p+q}$.*

DEMOSTRACIÓN. Por inducción sobre p. Si $p = 1$, entonces $\gamma^p \sqcup_c \gamma^q = \gamma \sqcup_c \gamma^q = \gamma^{1+q}$. Si $p > 1$, por definición, $\gamma^p = \gamma \sqcup_c \gamma^{p-1}$. Luego $\gamma^p \sqcup_c \gamma^q = (\gamma \sqcup_c \gamma^{p-1}) \sqcup_c \gamma^q$. Por la Proposición 15.3.44, $(\gamma \sqcup_c \gamma^{p-1}) \sqcup_c \gamma^q = \gamma \sqcup_c (\gamma^{p-1} \sqcup_c \gamma^q)$. Por la hipótesis de inducción, $\gamma \sqcup_c (\gamma^{p-1} \sqcup_c \gamma^q) = \gamma \sqcup_c \gamma^{p-1+q} = \gamma^{p+q}$. □

A partir de las definiciones anteriores, obtenemos el siguiente resultado, que formaliza la idea de que $^+$ es la operación de cierre asociada a la propiedad COMP.

Proposición 15.3.60 *Si $\gamma \in \Gamma$, entonces $\gamma^+ \in \Gamma^{\mathsf{COMP}}$.*

DEMOSTRACIÓN. Sea $u \in O^{\gamma^+}(C[t]_u)$ y $v \in O^{\gamma^+}(t)$. Por definición de $^+$, existen números naturales p, q tales que $u \in O^{\gamma^p}(C[t]_u)$ y $v \in O^{\gamma^q}(t)$, luego, por la Proposición 15.3.59, $u.v \in O^{\gamma^p \sqcup_c \gamma^q}(C[t]_u) = O^{\gamma^{p+q}}(C[t]_u) \subseteq O^{\gamma^+}(C[t]_u)$. □

La operación $\sqcup_c$ deja inalteradas las restricciones de reemplazamiento que ya satisfacen la propiedad COMP.

Proposición 15.3.61 *Si* $\gamma \in \Gamma^{\mathsf{ROOT+COMP}}$, *entonces* $\gamma^+ = \gamma$.

DEMOSTRACIÓN. Por inducción, y aplicando la Proposición 15.3.47, se demuestra que, para todo n, $\gamma^n = \gamma$. Por idempotencia de $\sqcup$, se obtiene $\gamma^+ = \sqcup_{n>0}\gamma^n = \sqcup_{n>0}\gamma = \gamma$. □

Puesto que $\Gamma^{\mathsf{ROOT+DECOMP}}$ es cerrado bajo $\sqcup_c$ y $\sqcup$ (Teorema 15.3.51 y Proposición 15.3.3), la Proposición 15.3.60 junto con el Teorema 15.3.25 establece una conexión interesante entre $\Gamma^{\mathsf{ROOT+DECOMP}}$ y Γ_{rsc}.

Teorema 15.3.62 *Si* $\gamma \in \Gamma^{\mathsf{ROOT+DECOMP}}$, *entonces* $\gamma^+ \in \Gamma_{rsc}$.

DEMOSTRACIÓN. Inmediata, por la Proposición 15.3.60 y el Teorema 15.3.25. □

Por tanto, es posible *abstraer* todo el conjunto $\Gamma^{\mathsf{ROOT+DECOMP}}$ empleando Γ_{rsc} como dominio abstracto y $^+$ como función de abstracción. En particular, al considerar el subconjunto $\Gamma^{\sqcup}_{rsc}$, obtenemos lo siguiente:

Proposición 15.3.63 *Si* $\gamma \in \Gamma^{\sqcup}_{rsc}$, *entonces* $\gamma^+ = \gamma_{\mu^\uparrow}$.

DEMOSTRACIÓN. Puesto que $\gamma \in \Gamma^{\sqcup}_{rsc}$, existen $\mu_1, \ldots, \mu_n$ tales que $\gamma = \gamma_{\mu_1} \sqcup \cdots \sqcup \gamma_{\mu_n}$. En primer lugar, demostramos que $\gamma^+ \sqsubseteq \gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n}$. Por la Proposición 15.3.9, $\gamma = \gamma_{\mu_1} \sqcup \cdots \sqcup \gamma_{\mu_n} \sqsubseteq \gamma_{\mu_1} \sqcup_{rsc} \cdots \sqcup_{rsc} \gamma_{\mu_n} = \gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n}$. Por monotonía de $\sqcup_c$ (Proposición 15.3.43), para todo $m > 0$ tenemos que $\gamma^m \sqsubseteq \gamma^m_{\mu_1 \sqcup \cdots \sqcup \mu_n}$. Por la Proposición 15.3.47, $\gamma^m_{\mu_1 \sqcup \cdots \sqcup \mu_n} = \gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n}$. Por tanto, $\gamma^m \sqsubseteq \gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n}$ y, debido a la idempotencia de $\sqcup$, $\gamma^+ = \sqcup_{m>0}\gamma^m \sqsubseteq \sqcup_{m>0}\gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n} = \gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n}$.

Ahora demostramos que $\gamma^+ \not\sqsubset \gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n}$ lo cual nos llevará a la conclusión deseada. Supongamos que $\gamma^+ \sqsubset \gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n}$. Puesto que, por el Teorema 15.3.62, $\gamma^+ \in \Gamma_{rsc}$, sea μ' tal que $\gamma^+ = \gamma_{\mu'}$. Sea $\mu = \mu_1 \sqcup \cdots \sqcup \mu_n$. Entonces, existe un $f \in \Sigma$ tal que $i \in \mu(f)$ e $i \notin \mu'(f)$. Por definición de μ, existe un j, $1 \leq j \leq n$, tal que $i \in \mu_j(f)$. Por tanto, si $t = f(\bar{t})$ tenemos que $i \in O^{\gamma_{\mu_j}}(t)$, es decir, $i \in O^{\gamma}(t)$ y, por la Proposición 15.3.45 y por definición de γ^+, tenemos $i \in O^{\gamma^+}(t) = O^{\gamma_{\mu'}}(t)$. Sin embargo, si $i \notin \mu'(f)$, entonces $i \notin O^{\gamma_{\mu'}}(t)$, lo cual lleva a una contradicción.

Dado que $\gamma_{\mu_1 \sqcup \cdots \sqcup \mu_n} = \gamma_{\mu_1} \sqcup_{rsc} \cdots \sqcup_{rsc} \gamma_{\mu_n}$, por la Proposición 15.3.30 obtenemos la conclusión. □

A partir de la caracterización de γ_Ω dada al final de la Sección 15.2.1 obtenemos el siguiente resultado, que es una consecuencia inmediata del Teorema 15.3.62.

Teorema 15.3.64 $\gamma^+_\Omega \in \Gamma_{rsc}$.

15.4 Propiedades computacionales de la reescritura restringida

En esta sección analizamos algunas propiedades computacionales de la relación de reducción $\hookrightarrow_{\mathcal{R}(\gamma)}$ inducida por un SRTE $\mathcal{R}$ bajo una restricción de reemplazamiento γ. Por analogía con las propiedades correspondientes de la relación de reescritura $\rightarrow_{\mathcal{R}}$, las denominaremos, como ya hicimos en el caso de la *rsc*, precediendo el nombre que identifica la propiedad con la restricción de reemplazamiento γ involucrada. Por tanto, hablaremos de la γ-confluencia, γ-terminación, etc., de un SRTE $\mathcal{R}$ refiriéndonos a la confluencia, terminación, etc., de la relación $\hookrightarrow_{\mathcal{R}(\gamma)}$. A continuación analizamos el papel desempeñado por las propiedades básicas de la Sección 15.2 en la caracterización de las propiedades computacionales de la γ-reescritura.

Proposición 15.4.1 (γ-terminación) *Sea $\mathcal{R}$ un SRTE y $\gamma, \gamma' \in \Gamma$ tales que $\gamma \sqsubseteq \gamma'$. Si $\mathcal{R}$ es γ'-terminante, entonces $\mathcal{R}$ es γ-terminante.*

DEMOSTRACIÓN. Hacemos $S = \hookrightarrow_{\mathcal{R}(\gamma)}$, $T = \hookrightarrow_{\mathcal{R}(\gamma')}$ y $h = id_{\mathcal{T}(\Sigma,V)}$ en el Teorema 5.1.2. La Proposición 15.3.2 asegura que $id_{\mathcal{T}(\Sigma,V)}$ es un homomorfismo entre $\hookrightarrow_{\mathcal{R}(\gamma)}$ y $\hookrightarrow_{\mathcal{R}(\gamma')}$ y, por tanto, entre $\hookrightarrow_{\mathcal{R}(\gamma)}$ y $\hookrightarrow^{+}_{\mathcal{R}(\gamma')}$. □

Trivialmente, si un SRT es terminante, también es γ-terminante para toda restricción $\gamma \in \Gamma$. Sin embargo, si un SRT es terminante, emplear una restricción de reemplazamiento para mejorar la terminación no tiene ningún sentido. No obstante, dado que existen criterios efectivos para demostrar la terminación de la *rsc* (Capítulo 5), podemos utilizar la aproximación *sc* superior $\alpha^{\sqsupseteq}(\gamma) = \gamma_{\mu\uparrow}$ para demostrar la terminación puesto que, siendo $\gamma \sqsubseteq \alpha^{\sqsubseteq}(\gamma)$, $\mathcal{R}$ es γ-terminante si es $\alpha^{\sqsupseteq}(\gamma)$-terminante.

La propiedad COMP está relacionada con la preservación de la reescritura restringida al tener ésta lugar en contextos reemplazantes.

Proposición 15.4.2 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\gamma \in \Gamma^{\mathsf{COMP}}$ y $C, t \in \mathcal{T}(\Sigma, V)$. Si $u \in O^{\gamma}(C[t]_u)$ y $t \overset{v}{\hookrightarrow} t'$, entonces $C[t]_u \overset{u.v}{\hookrightarrow} C[t']_u$. Si, además, $\gamma \in \Gamma^{\mathsf{REPL}}$, entonces $t \hookrightarrow^{*} s$ implica $C[t]_u \hookrightarrow^{*} C[s]_u$.*

DEMOSTRACIÓN. Si $t \overset{v}{\hookrightarrow} s$, entonces $t|_v$ es un redex y $v \in O^{\gamma}(t)$. Por COMP, $u.v \in O^{\gamma}(C[t]_u)$. Por tanto, puesto que $(C[t]_u)|_{u.v}$ es un redex, tenemos que $C[t]_u \overset{u.v}{\hookrightarrow} C[t']_u$. Si $\gamma \in \Gamma^{\mathsf{REPL}}$, entonces $u \in O^{\gamma}(C[t']_u)$. Por tanto, si $t \hookrightarrow t' \hookrightarrow^{*} s$, aplicando ahora inducción sobre la longitud de la derivación, también tenemos que $C[t]_u \hookrightarrow^{*} C[s]_u$. □

La siguiente proposición establece una propiedad interesante de $\sqcup_c$ que emplearemos más adelante.

Proposición 15.4.3 *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\gamma \in \Gamma^{\mathsf{REPL}}$, $\gamma' \in \Gamma$ y $C, t \in \mathcal{T}(\Sigma, V)$. Si $u \in O^{\gamma}(C[t]_u)$ y $t \hookrightarrow^{*}_{\mathcal{R}(\gamma')} s$, entonces $C[t]_u \hookrightarrow^{*}_{\mathcal{R}(\gamma \sqcup_c \gamma')} C[s]_u$.*

DEMOSTRACIÓN. Por inducción en la longitud n de la γ'-derivación. Si $n = 0$ es inmediato. Si $n > 0$, escribimos la derivación $t \hookrightarrow^*_{\mathcal{R}(\gamma')} s$ como sigue: $t \overset{v}{\hookrightarrow}_{\mathcal{R}(\gamma')} t' \hookrightarrow^*_{\mathcal{R}(\gamma')} s$. Puesto que $v \in O^{\gamma'}(t)$, tenemos que $u.v \in O^{\gamma \sqcup_c \gamma'}(C[t]_u)$. Por tanto, $C[t]_u \hookrightarrow_{\mathcal{R}(\gamma \sqcup_c \gamma')} C[t']_u$. Dado que $\gamma \in \Gamma^{\mathsf{REPL}}$, tenemos que $u \in O^{\gamma}(C[t']_u)$. Entonces, por la hipótesis de inducción, obtenemos la conclusión deseada. □

SUBST implica la estabilidad de la reescritura restringida.

Proposición 15.4.4 (Estabilidad de $\hookrightarrow$) *Sea $\mathcal{R} = (\Sigma, R)$ un SRTE, $\gamma \in \Gamma^{\mathsf{SUBST}}$, $\sigma \in Subst(\Sigma, V)$ y $t \in \mathcal{T}(\Sigma, V)$. Si $t \overset{u}{\hookrightarrow} s$, entonces $\sigma(t) \overset{u}{\hookrightarrow} \sigma(s)$.*

DEMOSTRACIÓN. Similar a la anterior. □

15.4.1 Completitud de la reescritura restringida

Para expresar el hecho de que a un término se le puede asociar un conjunto de términos (posiblemente vacío) pertenecientes a una familia canónica, emplearemos una función

$$S : \mathcal{T}(\Sigma, V) \to \mathcal{P}(\mathcal{T}^{\infty}(\Sigma, V))$$

(que llamaremos una *semántica de reescritura*) que a cada término t le asigna un conjunto de términos $S(t)$ pertenecientes a una determinada familia canónica de términos (posiblemente infinitos) que pueden obtenerse mediante reescritura (no restringida) a partir de t. Por ejemplo, la semántica est asocia a cada término t el conjunto $\mathsf{est}(t)$ de sus formas estables. Análogamente, fn y val asocia a cada término t el conjunto de sus formas normales $\mathsf{fn}(t)$ y valores $\mathsf{val}(t)$ respectivamente. Escribimos $\mathcal{B}$-$\mathsf{val}(t)$ si nos limitamos a algún subconjunto $\mathcal{B} \subseteq \mathcal{C}$ de símbolos constructores para definir los valores de interés. Ahora podemos definir una noción de completitud de una restricción de reemplazamiento relativa a una familia de formas canónicas.

Definición 15.4.5 (Completitud de una restricción de reemplazamiento) *Dada una semántica $S : \mathcal{T}(\Sigma, V) \to \mathcal{P}(\mathcal{T}^{\infty}(\Sigma, V))$ y una restricción de reemplazamiento $\gamma \in \Gamma$, decimos que γ es S-completa si para todo t, existe un $T \subseteq S(t)$ con $T = \emptyset \Leftrightarrow S(t) = \emptyset$ tal que, para todo $s \in T$, $t \hookrightarrow^* s$.*

Es decir, una restricción de reemplazamiento γ es S-completa si, para todo término t, somos capaces de obtener, mediante γ-reducción, algún término canónico $s \in S(t)$ especificado por la semántica de t. La siguiente proposición resulta evidente también.

Proposición 15.4.6 (Monotonía de la γ-completitud) *Sean $\gamma, \gamma' \in \Gamma$ tales que $\gamma \sqsubseteq \gamma'$. Si γ es S-completa, entonces γ' es S-completa.*

Al disponer del operador $\sqcup_c$ (más restrictivo que $\sqcup_{rsc}$, de acuerdo con la Proposición 15.3.54) surge otra vía de solución al problema de la pérdida del comportamiento terminante al pasar de la función de reemplazamiento $\mu^{com}_{\mathcal{R}}$ a la función $\mu^{\mathcal{B}}_{\mathcal{R}}$ para obtener la evaluación completa de una expresión inicial que ya fue discutido en el Capítulo 11. Ahora podemos emplear $\gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma_{\mu^{com}_{\mathcal{R}}}$ en vez de $\gamma_{\mu^{\mathcal{B}}_{\mathcal{R}}} = \gamma_{\mu_{\mathcal{B}}} \sqcup_{rsc} \gamma_{\mu^{com}_{\mathcal{R}}}$ (donde $\mu_{\mathcal{B}}$ es como en la Definición 11.1.3).

Teorema 15.4.7 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE confluente, $\mathcal{B} \subseteq \mathcal{C}$ y $\delta \in \mathcal{T}(\mathcal{B}, V)$. Sea $\gamma \in \Gamma$* est*-completa. Si $\gamma' = \gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma$, entonces γ' es $\mathcal{B}$-*val*-completa.*

DEMOSTRACIÓN. Sea $t \rightarrow^* \delta$ para algún $\{\delta\} = \mathcal{B}\text{-val}(t)$ (por confluencia, debe tratarse de un conjunto unitario). Realizamos la demostración por inducción sobre la estructura de δ.

Si $\delta \in \Sigma \cup V$, entonces δ es la única forma estable de t. Dado que γ es est-completa, tenemos que $t \hookrightarrow^*_{\mathcal{R}(\gamma)} \delta$. Puesto que $\gamma_{\mu_{\mathcal{B}}} \in \Gamma_{rsc} \subseteq \Gamma^{\mathsf{ROOT}}$, por la Proposición 15.3.45, $\gamma \sqsubseteq \gamma'$. Por tanto, por la Proposición 15.4.6, tenemos que $t \hookrightarrow^*_{\mathcal{R}(\gamma')} \delta$.

Si $\delta = c(\tilde{\delta})$, puesto que γ es est-completa, podemos descomponer la derivación $t \rightarrow^* \delta$ como $t \hookrightarrow^*_{\mathcal{R}(\gamma)} s \rightarrow^* \delta$, donde s es una forma estable constructora cuyo símbolo constructor en cabeza $c = root(s)$ coincide con $root(\delta)$ (por confluencia). Sea $t \hookrightarrow^* c(s_1, \ldots, s_k) \rightarrow^* \delta$. Por la hipótesis de inducción, dado que, por confluencia, $\mathcal{B}\text{-val}(s_i) = \{\delta_i\}$ y, por definición de $\mathcal{B}$-val-completitud, tenemos que $s_i \hookrightarrow^*_{\mathcal{R}(\gamma')} \delta_i$ para todo $1 \leq i \leq k$. Dado que los argumentos de $c \in \mathcal{B}$ son reemplazantes (pues $\mu_{\mathcal{B}}(c) = \mathbb{N}^+_{ar(c)}$ para todo $c \in \mathcal{B}$), por la Proposición 15.4.3, $c(s_1, \ldots, s_k) \hookrightarrow^*_{\mathcal{R}(\gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma')} \delta$. Por asociatividad de $\sqcup_c$ (Proposición 15.3.44) y por la Proposición 15.3.47, $\gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma' = \gamma_{\mu_{\mathcal{B}}} \sqcup_c (\gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma) = (\gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma_{\mu_{\mathcal{B}}}) \sqcup_c \gamma = \gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma = \gamma'$. Esto completa la demostración. □

Este resultado proporciona soporte formal para la utilización práctica de la operación $\sqcup_c$. La siguiente proposición asegura que ningún redex no deseado va a ser considerado al utilizar $\gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma$ en lugar de γ.

Proposición 15.4.8 *Sea $\mathcal{R} = (\Sigma, R) = (\mathcal{C} \uplus \mathcal{F}, R)$ un SRTE, $\mathcal{B} \subseteq \mathcal{C}$ y $\gamma \in \Gamma$. Sea $\gamma' = \gamma_{\mu_{\mathcal{B}}} \sqcup_c \gamma$ y $t = C[t_1, \ldots, t_n]$, donde $C[\,] = CCM(t) \subseteq \mathcal{T}(\mathcal{B} \uplus \{\Box\}, V)$. Sean $u_1, \ldots, u_n \in O(t)$ las ocurrencias de $t_1, \ldots, t_n$ en t. Entonces, $O^{\gamma'}_{\mathcal{R}}(t) = \cup_{1 \leq i \leq n} u_i . O^{\gamma}_{\mathcal{R}}(t_i)$.*

DEMOSTRACIÓN. Por inducción sobre la estructura del contexto constructor común $C[\,]$ de t. Si $C[\,] = x \in V$, o $C[\,] = c \in \mathcal{B}$ entonces $t = x$ o $t = c$ y el resultado es inmediato. Si $C[\,] = \Box$, entonces $root(t) \in \mathcal{F}$, puesto que $\mu_{\mathcal{B}}(f) = \emptyset$ para todo $f \in \mathcal{F}$, tenemos que $O^{\gamma_{\mu_{\mathcal{B}}}}(t) = \{\epsilon\}$. Por definición de γ', debe ser $O^{\gamma}(t) = O^{\gamma'}(t)$. Entonces $O^{\gamma}_{\mathcal{R}}(t) = O^{\gamma'}_{\mathcal{R}}(t)$, lo que permite concluir el resultado deseado. Si $C[\,] = c(C_1[\,], \ldots, C_k[\,])$, se tiene que $u_1, \ldots, u_n \in O^{\gamma_{\mu_{\mathcal{B}}}}(t)$. Dado un u_i, para $1 \leq i \leq n$, sabemos que no hay ninguna ocurrencia v_i de un redex tal que $v_i < u_i$, puesto que

$root(t|_{v_i}) \in \mathcal{B}$. Por la hipótesis de inducción y por definición de γ' y de $\sqcup_c$, obtenemos la conclusión deseada. □

Por tanto, empleando $\sqcup_c$ obtenemos un método alternativo (a la transformación V) para evitar las contracciones de redexes inútiles o que puedan iniciar secuencias de reducción infinita en los pasos marcados con $(*)$ o $(**)$ en los Ejemplos 11.2.1 o 11.2.2; el Teorema 15.4.7 asegura la $\mathcal{B}$-val-completitud de $\gamma_{\mu_\mathcal{B}} \sqcup_c \gamma$ en el cómputo de valores.

15.4.2 Confluencia de la reescritura restringida

En esta sección analizamos el efecto de incluir restricciones de reemplazamiento en la confluencia de la relación de reescritura. De forma similar a lo realizado para la *rsc*, obtenemos un resultado sobre γ-confluencia mediante la generalización del *parallel moves lemma* (Lema 2.3.9).

De forma similar a como se estudió la confluencia fuerte para la *rsc* en la Sección 6.3, analizamos los requisitos que una restricción de reemplazamiento γ debe reunir para poder aplicar el *parallel moves lemma* a la γ-reescritura (paralela). En primer lugar, necesitamos mantener en las derivaciones γ-restringidas la propiedad siguiente:

$$v \in O^\gamma_\mathcal{R}(t) \Rightarrow v\backslash A^\gamma \subseteq O^\gamma_\mathcal{R}(s)$$

donde la notación empleada se refiere a la siguiente definición.

Definición 15.4.9 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\gamma \in \Gamma$. Dada una γ-derivación elemental $A^\gamma : t \overset{[u,\alpha]}{\hookrightarrow}_{\mathcal{R}(\gamma)} s$, donde $\alpha : l \to r \in R$ y $v \in O^\gamma_\mathcal{R}(t)$, el conjunto $v\backslash A^\gamma$ de* residuos *de v por A^γ es $v\backslash A^\gamma = v\backslash A$.*

De forma similar a la *rsc*, un residuo de un redex γ-reemplazante no tiene porqué ser γ-reemplazante (ver el Ejemplo 6.3.2). Por el momento, exigiremos también que los SRTs considerados pertenezcan a la clase determinada por la siguiente definición.

Definición 15.4.10 *Sea $\mathcal{R} = (\Sigma, R)$ un SRT y $\gamma \in \Gamma$. Decimos que $\alpha : l \to r \in R$ tiene* variables γ-reemplazantes homogéneas por la izquierda *si, para toda $x \in Var^\gamma(l)$, $O_x(l) = O^\gamma_x(l)$ y $O_x(r) = O^\gamma_x(r)$. El SRT $\mathcal{R}$ tiene variables γ-reemplazantes homogéneas por la izquierda si todas sus reglas $l \to r \in R$ tienen variables γ-reemplazantes homogéneas por la izquierda*

De los SRTs con variables γ-reemplazantes homogéneas por la izquierda se espera que verifiquen la siguiente propiedad: las instancias de variables γ-reemplazantes en las *lhs*'s son γ-reemplazantes en las *rhs*'s. Veremos que para que se verifique esta propiedad podemos imponer (como condición suficiente) que $\gamma \in \Gamma^{\mathsf{SUBST}}$. Además, COMP nos permite disponer de esta propiedad en el interior de cualquier contexto.

Al considerar SRTs con variables γ-reemplazantes homogéneas por la izquierda (con $\gamma \in \Gamma^{\mathsf{COMP+SUBST}}$), los residuos de redexes reemplazantes son reemplazantes. Entonces, las extensiones usuales de la noción de residuo son correctas. En particular, el concepto de derivación residual $B\backslash A$. Ciertamente, si no podemos asegurar que $v\backslash A^\gamma \subseteq O^\gamma_{\mathcal{R}}(s)$, no podremos asegurar que $U\backslash A^\gamma \subseteq O^\gamma_{\mathcal{R}}(s)$. Puesto que la derivación residual $B^\gamma\backslash A^\gamma$ contrae el conjunto $U\backslash A^\gamma$, dicha derivación no estaría bien definida si este conjunto contuviera redexes no reemplazantes.

La contracción de ocurrencias de redexes relativos a SRTs ortogonales *no* elimina otros redexes ya presentes. Esto es así porque las *lhs*'s de las reglas no definen pares críticos que podrían destruirlos. Con relación a los γ-redexes, también podemos asegurar esta propiedad si prohibimos (mediante γ) el reemplazamiento de redexes que podrían definir pares críticos. Las nociones de par γ-crítico y SRT γ-ortogonal son completamente análogas a las definiciones dadas para la *rsc* (Definiciones 6.1.5 y 6.3.3). De nuevo, para hacer operativas estas definiciones de modo que se consigan las propiedades esperadas, podemos imponer como condiciones suficientes que las restricciones de reemplazamiento involucradas satisfagan $\gamma \in \Gamma^{\mathsf{SUBTERM+}\overline{\mathsf{SUBST}}}$. Esto asegura que las ocurrencias no γ-reemplazantes de las *lhs*'s no se convierten en reemplazantes en las instancias de la *lhs*. Por otro lado, $\gamma \in \Gamma^{\mathsf{SUBTERM}}$ asegura que esta propiedad se mantiene cuando las ocurrencias de redexes están situadas en un contexto $C[\,]$ cualquiera. Si existiera un término $t = C[\sigma(l)]_u$ donde se encontrara un γ-redex situado sobre una ocurrencia $u.v \in O^\gamma(t)$, con $v \in \overline{O^\gamma}(l)$ tal que se definiera un par crítico, entonces, por la propiedad SUBTERM, $v \in O^\gamma(\sigma(l))$. Pero $\overline{\mathsf{SUBST}}$ asegura que $v \notin O^\gamma(\sigma(l))$ y llegamos a una contradicción.

A continuación establecemos que el conjunto de condiciones impuestas para los programas y las restricciones de reemplazamiento implican el resultado deseado.

Proposición 15.4.11 *Sea $\gamma \in \Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}}$ tal que $\mathcal{R} = (\Sigma, R)$ es γ-ortogonal y tiene variables γ-reemplazantes homogéneas por la izquierda. Sea $A^\gamma : t \overset{[u,\alpha]}{\hookrightarrow}_{\mathcal{R}(\gamma)} s$ una γ-derivación elemental, donde $\alpha : l \to r \in R$ y $v \in O^\gamma_{\mathcal{R}}(t)$. Entonces $v\backslash A^\gamma \subseteq O^\gamma_{\mathcal{R}}(s)$.*

Demostración. Seguimos los apartados etiquetados con (i), (ii) e (iii) de la definición de residuo (Definición 2.3.8) para *deducir* qué propiedades de las restricciones de reemplazamiento son suficientes para preservar la propiedad de que los residuos de redexes γ-reemplazantes también lo sean.

1. A partir de (i), exigimos CONTEXT puesto que, tras el reemplazamiento realizado por la derivación A^γ a la ocurrencia $u \in O^\gamma_{\mathcal{R}}(t)$ tal que $u \parallel v$, debe tenerse $v\backslash A^\gamma = \{v\} \subseteq O^\gamma_{\mathcal{R}}(s)$. Si $\gamma \in \Gamma^{\mathsf{CONTEXT}}$, esta condición se verifica.

2. De acuerdo con (ii), exigimos REPL. De esta manera, el reemplazamiento realizado por la γ-derivación elemental A que, puesto que $v < u$, tiene lugar *debajo*

de v, no hace que $s|_v$ deje de ser un redex (debido a la ortogonalidad) e incluso un γ-redex (por REPL).

3. De acuerdo con (iii), imponemos que $\mathcal{R}$ tenga variables γ-reemplazantes homogéneas por la izquierda y también, de acuerdo con la discusión previa, $\gamma \in \Gamma^{\mathsf{DOWN+SUBTERM+COMP+REPL+\overline{SUBST}}}$. De esta forma, SUBTERM asegura que v_1 es una ocurrencia reemplazante de $t|_{u.w}$. Por otro lado, DOWN asegura que $u.w \in O^\gamma(t)$. Por SUBTERM, $w \in O^\gamma(t|_u) = O^\gamma(\sigma(l))$. Dado que $w \in O(l)$, por $\overline{\mathsf{SUBST}}$, $w \in O^\gamma(l)$. Por tanto, $l|_w = x \in Var^\gamma(l)$. Puesto que $\mathcal{R}$ tiene variables γ-reemplazantes homogéneas por la izquierda, se verifica que $w_1 \in O^\gamma(s|_u)$. Mediante REPL, aseguramos que u permanece siendo reeplazante. Por fin, COMP asegura la propiedad deseada.

4. También necesitamos imponer la ortogonalidad con el fin de justificar (iii). De no ser así, no podríamos expresar con propiedad v como $v = u.w.v_1$ donde $l|_w$ es una variable. Sin embargo, podemos relajar este requisito exigiendo simplemente la γ-ortogonalidad del SRT.

Finalmente, puesto que requerimos CONTEXT y REPL, por la Proposición 15.2.6, no es necesario hacer explícita la necesidad de SUBST y $\overline{\mathsf{SUBST}}$. De forma análoga, podemos emplear DECOMP en vez de DOWN+SUBTERM. □

Ahora estamos en condiciones de generalizar el *parallel moves lemma* a la reescritura restringida.

Lema 15.4.12 *Sea $\mathcal{R}$ un SRT y $\gamma \in \Gamma^{\mathsf{COMP+DECOMP+CONTEXT+REPL}}$, tal que $\mathcal{R}$ es γ-ortogonal y tiene variables γ-reemplazantes homogéneas por la izquierda. Sean A^γ y B^γ dos γ-multiderivaciones elementales que arrancan de un mismo término t. Entonces, $B^\gamma(A^\gamma \backslash B^\gamma)$ y $A^\gamma(B^\gamma \backslash A^\gamma)$ son γ-derivaciones que parten de t y llevan al mismo término s y, para toda ocurrencia redex $u \in O^\gamma_{\mathcal{R}}(t)$, tenemos que $u \backslash B^\gamma(A^\gamma \backslash B^\gamma) = u \backslash A^\gamma(B^\gamma \backslash A^\gamma)$.*

El Lema 15.4.12 establece la confluencia fuerte de $\hookrightarrow_{\mathcal{R}(\gamma)}$. Para concluir la confluencia de la relación, podemos emplear el Lema 2.1.1 y, para ello, primero debemos demostrar que $\hookrightarrow^+ = \Vdash^+$. Esto resulta sencillo si imponemos que $\gamma \in \Gamma^{\mathsf{CONTEXT+REPL}}$. Entonces obtenemos el siguiente resultado.

Teorema 15.4.13 *Sea $\mathcal{R}$ un SRT y $\gamma \in \Gamma^{\mathsf{DECOMP+COMP+CONTEXT+REPL}}$, tal que $\mathcal{R}$ es γ-ortogonal y tiene variables γ-reemplazantes homogéneas por la izquierda. Entonces $\mathcal{R}$ es γ-confluente.*

No obstante, este resultado no añade mucho más a los ya obtenidos para la *rsc*. La siguiente proposición justifica esta afirmación.

Proposición 15.4.14 $\Gamma^{\mathsf{DECOMP+COMP+CONTEXT+REPL}} = \Gamma_{rsc} \cup \{\gamma_\perp\}$

DEMOSTRACIÓN. Inmediata, por el Teorema 15.3.24 y la Proposición 15.2.9 □

Esto parece indicar que el método de demostración empleado para establecer la confluencia resulta difícilmente adaptable a restricciones de la reescritura distintas de la *rsc*.

Conclusiones y trabajo futuro

El material presentado en esta tesis demuestra que el empleo de relaciones de reducción definidas a partir de un SRT mediante restricciones de reemplazamiento sintácticas y, en particular, la reescritura sensible al contexto, constituyen una alternativa viable (y más eficaz) a la reescritura sin (o con otro tipo de) restricciones. Se ha demostrado la posibilidad de determinar formalmente sus propiedades computacionales haciendo uso de métodos estándar o mediante una sencilla generalización de éstos.

En algunos ambientes computacionales (en particular en programación funcional, donde el cómputo de valores constituye el objetivo de la ejecución de un programa, y en sistemas perezosos, donde el cómputo de formas estables resulta de particular interés) la *rsc* se muestra semánticamente equivalente a la reescritura estándar. En otros casos constituye una base adecuada para definir procedimientos o estrategias que permitan obtener los resultados deseados con un ajuste de control apropiado. Además, gracias a su simplicidad, las restricciones sensibles al contexto pueden implementarse fácilmente. El formalismo empleado para su definición y estudio es simple y permite razonar con sencillez acerca de sus propiedades.

Trabajo futuro

Hemos analizado con cierta profundidad las propiedades de la *rsc* y, dado que muchas de las demostraciones de los resultados estándar para la reescritura pueden reformularse con relativa sencillez para acomodar las peculiaridades introducidas por la *rsc*, podemos extraer la conclusión de que su definición a dos niveles, simbólico y estructural, encaja de forma natural en la teoría de reescritura. Por esa misma razón, parece lógico pensar que prácticamente cualquier resultado en reescritura es susceptible de revisión, analizando las repercusiones resultantes de la imposición de restricciones de reemplazamiento sensibles al contexto.

En particular, un análisis más profundo de las cuestiones y métodos relacionados con la demostración de la terminación de la *rsc* parece deseable, puesto que la mejora del comportamiento respecto a la terminación constituye una de las potencialidades más evidentes de una restricción de la reescritura y la posibilidad de establecer

formalmente ésta resulta de indudable interés.

Mencionemos algunas de las grandes áreas de trabajo en reescritura que no han sido tratadas aquí. En particular, no se ofrece ningún resultado acerca de la *modularidad* de las propiedades computacionales estudiadas (terminación, confluencia y completitud). La modularidad se refiere, normalmente, a combinaciones de SRTs distintos para constituir un único SRT. En el caso de la *rsc* también resulta de interés considerar la modularidad de las propiedades citadas respecto a combinaciones de funciones de reemplazamiento asociadas al mismo (o distintos) SRTs.

En otros temas de investigación como el estrechamiento y los sistemas de reescritura condicionales podemos esperar mayores beneficios si realizamos un análisis más profundo acerca de la interacción de éstos con las restricciones de reemplazamiento sensibles al contexto. En el caso del estrechamiento, una extensión evidente (e inexplorada) de la idea de restricción de reemplazamiento es la de *restricción de la instanciación* que podría estar ligada a las (ocurrencias de) variables de los objetivos a evaluar. La forma más simple de hacer esto sería forzar a que las variables situadas sobre ocurrencias no reemplazantes del objetivo no pudieran instanciarse. Otra posibilidad sería asociar restricciones *de instanciación* a los símbolos de variable, pero esta idea parece requerir una definición más compleja del propio concepto de estrechamiento controlado por restricciones sintácticas (de reemplazamiento/instanciación). También, dadas las buenas propiedades de la *rsc* relativas a la mejora de la terminación, parece prometedora la combinación de la *rsc* y el *esc* en un marco análogo al del estrechamiento con normalización o simplificación previa.

En el caso de los SRTs condicionales, cabe explorar la posibilidad de mejorar las cuestiones de decidibilidad involucradas en los cómputos con dichos sistemas al incluir las restricciones de reemplazamiento (de forma similar a como la terminación resulta mejorada al imponer éstas). Esto parece particularmente interesante en el caso de SRTCs con variables extra donde la condición suficiente usual de *decreasingness* [DO90] no resulta fácilmente aplicable.

También resultaría de interés estudiar las propiedades del estrechamiento sensible al contexto para SRTCs.

En el ámbito de la evaluación parcial de programas lógico-funcionales [AFV98, AFV98b, AFJV97, AHLV98b], la noción de argumento redundante de un símbolo de función respecto a una semántica de reescritura o estrechamiento [AHLV98] puede considerarse, desde el punto de vista computacional, como un caso extremo de argumento no reemplazante. Aunque ambos conceptos difieren en la fundamentación semántica subyacente (de carácter más operacional en el caso de las restricciones de reemplazamiento estudiadas en esta tesis), el tratamiento formal, en ambos casos, presenta similitudes evidentes que sugieren la existencia de conexiones más profundas dignas de un análisis detallado.

Finalmente, pensamos que merece la pena profundizar en el desarrollo de la noción de restricción de reemplazamiento en general (Capítulo 15) y en el desarrollo ulterior del marco lógico-algebraico presentado. Por ejemplo, la aplicación de dicho marco a estrategias de reescritura tradicionales permitiría disponer de un álgebra de estrategias cuyas propiedades podrían determinarse a partir de los constituyentes de las mismas y de las operaciones involucradas en su descripción.

Bibliografía

[AEH94] S. Antoy R. Echahed, and M. Hanus. A needed narrowing strategy. In *Proc. of the 21st Annual ACM Symposium on Principles of Programming Languages, POPL'94*, pages 268-279, ACM Press, 1994.

[AFJV97] M. Alpuente, M. Falaschi, P. Julián, and G. Vidal. Specialization of Lazy Functional Logic Programs. *Proc. of the ACM SIGPLAN Conference on Partial Evaluation and Semantics-Based Program Manipulation, PEPM'97*, Sigplan Notices, 32(12):151-162, ACM Press, 1997.

[AFV98] M. Alpuente, M. Falaschi, and G. Vidal. Partial Evaluation of Functional Logic Programs. *ACM Transactions on Programming Languages and Systems*, 1998, to appear.

[AFV98b] M. Alpuente, M. Falaschi, and G. Vidal. A Unifying View of Functional and Logic Program Specialization. *ACM Computing Surveys*, 1998, to appear.

[AG97] T. Arts and J. Giesl. Automatically Proving Termination Where Simplification Orderings Fail. In M. Bidoit and M. Dauchet, editors, *Proc. of 7th International Conference on the Theory and Practice of Software Development, TAPSOFT'97*, LNCS 1214, Springer-Verlag, Berlin, 1997.

[AHLV98] M. Alpuente, M. Hanus, S. Lucas, and G. Vidal. Redundancy Analyses in Narrowing and Rewriting. In preparation, 1998.

[AHLV98b] M. Alpuente, M. Hanus, S. Lucas, and G. Vidal. Specialization of Functional Logic Programs Based on Needed Narrowing. In preparation, 1998.

[ALN87] H. Aït-Kaci, P. Lincoln, and R. Nasr. Le Fun: Logic, equations and Functions. In *Proc. of 4th International Symposium on Logic Programming*, pages 17-23, IEEE Press, 1987.

[AM96] S. Antoy and A. Middeldorp. A Sequential Reduction Strategy. *Theoretical Computer Science* 165:75-95, 1996.

[AN80] A. Arnold and M. Nivat. The metric space of infinite trees. Algebraic and topological properties. *Fundamenta Informaticae* 4:445-476, 1980.

[Ant92] S. Antoy. Definitional Trees. In H. Kirchner and G. Levi, editors, *Proc. of 3rd International Conference on Algebraic and Logic Programming, ALP'92*, LNCS 632:143-157, Springer-Verlag, Berlin, 1992.

[Art96] T. Arts. Termination by absence of infinite chains of dependency pairs. In H. Kirchner, editor, *Proc. of 21st International Colloquium on Trees in Algebra and Programming, CAAP'96*, LNCS 1059:196-210, Springer-Verlag, Berlin, 1996.

[BCG75] A. Bigard , M. Crestey et J. Grappy. Problèmes d'algèbre générale. Dunod, Paris. Versión española: Problemas de álgebra general. Reverté, 1975.

[BD94] L. Bachmair and N. Dershowitz. Equational Inference, canonical proofs and proof orderings. *Journal of the ACM* 41(2):236-276, 1994.

[BEGKPS87] H.P. Barendregt, M.C.J.D. van Eekelen, J.R.W. Glauert, J.R. Kennaway, M.J. Plasmeijer, and M.R. Sleep. Term Graph Rewriting. In J.W. de Bakker, A.J. Nijman, and P.C. Treleaven, editors, *Proc. of 2nd Conference on Parallel Architectures and Languages Europe, PARLE'87* LNCS 259:141-158, Springer-Verlag, Berlin, 1987.

[Ben92] P.N. Benton. Strictness Analysis of Lazy Functional Programs. PhD Thesis, University of Cambridge Computer Laboratory, 1992.

[BEØ93] D. Bert, R. Echahed, and B.M. Østvold. Abstract Rewriting. In P. Cousot, M. Falaschi, G. File, and A. Rauzy, editors, *Proc. of 3rd International Workshop on Static Analysis, WSA'93* LNCS 724:178-192, Springer-Verlag, Berlin, 1993.

[BK86] J.A. Begstra and J.W. Klop. Conditional Rewrite Rules: Confluence and Termination. *Journal of Computer and System Sciences* 32:323-362, 1986.

[BKLOS93] M. Bidoit, H.-J. Kreowski, P. Lescanne, F. Orejas, and D. Sannella, editors. Algebraic System Specificacion and Development. A Survey and Annotated Bibliography. LNCS 501, Springer-Verlag, Berlin, 1993.

[BL87] F. Bellegarde and P. Lescanne. Transformation ordering. In H. Ehrig, R. Kowalski, G. Levi, and U. Montanari, editors, *Proc. of 12th International Colloquium on Trees in Algebra and Programming, CAAP'87*, LNCS 249:69-80, Springer-Verlag, Berlin, 1987.

[Blo76] S.L. Bloom. Varieties of Ordered Algebras. *Journal of Computer and System Sciences*, 13:200-212, 1976.

[BM77] G. Birkhoff and S. MacLane. A Survey of Modern Algebra (*4th edition*). Macmillan Publishing Co., Inc., New York, 1977.

[Bou85] G. Boudol. Computational semantics of term rewriting systems. In M. Nivat and J. Reynolds, editors, *Algebraic Methods in Semantics*, pages 169-236, Cambridge University Press, Cambridge, 1985.

[Bur87] G.L. Burn. Evaluation Transformers - A Model for the Parallel Evaluation of Functional Languages (Extended Abstract). In G. Kahn, editor, *Proc. of the Functional Programming Languages and Computer Architecture Conference*, LNCS 274:446-470, Springer-Verlag, Berlin, 1987.

[Bur91] G.L. Burn. Lazy Functional Languages: Abstract Interpretation and Compilation. Pitman, London and The MIT Press, Cambridge, MA, 1991.

[Com95] H. Comon. Sequentiality, Second Order Monadic Logic and Tree Automata. In *10th Annual IEEE Symposium on Logic in Computer Science, LICS'95*, pages 508-517, IEEE Computer Society Press, 1995.

[Cou80] B. Courcelle. Completions of ordered magmas. *Fundamenta Informaticae* 3(1):105-116, 1980.

[Cou83] B. Courcelle. Fundamental Properties of Infinite Trees. *Theoretical Computer Science* 25:95-169, 1983.

[Cou90] B. Courcelle. Recursive Applicative Program Schemes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, pages 459-492, Elsevier, 1990.

[CP85] C. Clack and S.L. Peyton-Jones. Strictness analysis – a practical approach. In J.-P. Jouannaud, editor, *Functional Programming and Computer Architecture*, LNCS 201:35-39, Springer-Verlag, Berlin, 1985.

[Dav92] A.J.T. Davie. An Introduction to Functional Programming Systems Using Haskell. Cambridge University Press, Cambridge, 1992.

[DD76] P. Dubreil et M.L. Dubreil-Jacotin. Leçons d'algèbre moderne. Dunod, Paris, 1961. Versión española: Lecciones de álgebra moderna. Reverté, Barcelona, 1976.

[Der82] N. Dershowitz. Orderings for term rewriting systems. *Theoretical Computer Science* 17:279-301, 1982.

[Der87] N. Dershowitz. Termination of rewriting. *Journal of Symbolic Computation*, 3:69-115, 1987.

[Der93] N. Dershowitz. A Taste of Rewrite Systems. In P.E. Lauer, editor, *Proc. of Functional Programming, Concurrency, Simulation and Automated Reasoning*, LNCS 693:199-228, Springer-Verlag, Berlin, 1993.

[DH95] N. Dershowitz and C. Hoot. Natural Termination. *Theoretical Computer Science* 142(2):179-207, 1995.

[DJ90] N. Dershowitz and J.-P. Jouannaud. Rewrite Systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B: Formal Models and Semantics, pages 243-320. Elsevier, Amsterdam and The MIT Press, Cambridge, MA, 1990.

[DKP91] N. Dershowitz, S. Kaplan, and D. Plaisted. Rewrite, rewrite, rewrite, rewrite, rewrite. *Theoretical Computer Science* 83:71-96, 1991.

[DM79] N. Dershowitz and Z. Manna. Proving Termination with Multisets Orderings. *Communications of the ACM* 22(8):465-476, 1979.

[DM97] I. Durand and A. Middeldorp. Decidable Call by Need Computations in Term Rewriting (Extended Abstract). In W. McCune, editor, *Proc. of 14th International Conference on Automated Deduction, CADE'97*, LNAI 1249:4-18, Springer-Verlag, Berlin, 1997.

[DO90] N. Dershowitz and M. Okada. A rationale for conditional equational programming. *Theoretical Computer Science* 75:111-138, 1990.

[DOS88] N. Dershowitz, M. Okada, and G. Sivakumar. Canonical conditional rewrite systems. In E. Lusk and R. Overbeek, editors *Proc. of 9th International Conference on Automated Deduction, CADE'88*, LNCS 310:538-549, Springer-Verlag, Berlin, 1988.

[DOS88b] N. Dershowitz, M. Okada, and G. Sivakumar. Confluence of conditional rewrite systems. In S. Kaplan and J.-P. Jouannaud, editors, *Proc. of 1st International Workshop on Conditional Term Rewriting Systems, CTRS'87*, LNCS 308:31-44, Springer-Verlag, Berlin, 1988.

[DP88] N. Dershowitz and D.A. Plaisted. Equational Programming. In J.E. Hayes, D. Michie, and J. Richards, editors, *Machine Intelligence*, 11:21-56. Clarendon Press, Oxford, 1988.

[DS87] N. Dershowitz and G. Sivakumar. Solving Goals in Equational Languages. In S. Kaplan and J.-P. Jouannaud, editors *Proc. of 1st International Workshop on Conditional Term Rewriting Systems, CTRS'87*, LNCS 308:45-55, Springer-Verlag, Berlin, 1988.

[DS93] I. Durand and B. Salinier. Constructor equivalent term rewriting systems. *Information Processing Letters*, 47:131-137, 1993.

[DS96] F.R. Drake and D. Singh. Intermediate set theory. John Wiley & sons, New York, 1996.

[DTHL87] M. Dauchet, S. Tison, T. Heuillard, and P. Lescanne. Decidability of Confluence for Ground Term Rewriting Systems. In *Proc. of 2nd Annual IEEE Symposium on Logic in Computer Science, LICS'87*, pages 353-359, IEEE Computer Society Press, 1987.

[Dur94] I. Durand. Bounded, Strongly Sequential and Forward-Branching Term Rewriting Systems. *Journal of Symbolic Computation*, 18:319-352, 1994.

[Dur95] I. Durand. Deciding strong sequentiality for orthogonal term rewriting systems is co-np. *Technical report*, 1090-95, LaBri, 1995.

[EM85] H. Ehrig and B. Mahr. Fundamentals of Algebraic Specification. volume 6 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, Berlin, 1985.

[ENPS92] M.C.J.D. van Eekelen, E.G.J.M.H. Nöcker, M.J. Plasmeijer, and J.E.W. Smetsers. Concurrent Clean. Language Manual - Version 0.8. Technical Report 92-18, University of Nijmegen, 1991.

[EP87] M.C.J.D. van Eekelen and M.J. Plasmeijer. Specification of Reduction Strategies in Term Rewriting Systems. In J.H. Fasel and R.M. Keller, editors, *Graph Reduction. Proceedings of a Workshop*, LNCS 279:215-239. Springer-Verlag, Berlin, 1987.

[Fay79] M.J. Fay. First-order Unification in an Equational Theory. In *Proc. of 4th Workshop on Automated Deduction*, pages 161-167, Academic Press, 1979.

[Fer95] M.C. Fernández Ferreira. Termination of Term Rewriting. Well-foundedness, Totality and Transformations. PhD thesis, Faculteit Wiskunde en Informatica, Universiteit Utrecht, November 1995.

[FGJM85] K. Futatsugi, J. Goguen, J.-P. Jouannaud, and J. Meseguer. Principles of OBJ2. In *Conference Record of the 12th Annual ACM Symposium on Principles of Programming Languages, POPL'85*, pages 52-66, ACM Press, 1985.

[FH88] A.J. Field and P.G. Harrison. Functional Programming. Addison-Wesley Publishing Company, Reading, MA, 1988.

[FN97] K. Futatsugi and A. Nakagawa. An Overview of CAFE Specification Environment – An algebraic approach for creating, verifying, and maintaining formal specification over networks –. In *Proc. of 1st International Conference on Formal Engineering Methods*, 1997.

[FR98] M.C.F. Ferreira and A. Ribeiro. Context-Sensitive AC-rewriting. Draft, 1998.

[Fri85] L. Fribourg. SLOG: a logic programming language interpreter based on clausal superposition and rewriting. In *Proc. of 2nd. IEEE International Symposium on Logic Programming,*, pages 172-185, IEEE Press, 1985.

[FW76] D.P. Friedman and D.S. Wise. CONS should not evaluate its arguments. In S. Michaelson and R. Milner, editors, *Automata, Languages and Programming*, pages 257-284, Edinburgh University Press, 1976.

[Gar98] J.J. García Guaita. Utilización de la reescritura sensible al contexto en programación. Universidad Politécnica de Valencia, Facultad de Informática, Proyecto Fin de Carrera, 1998.

[GHKLMS80] G. Gierz, K.H. Hofmann, K. Keimel, J.D. Lawson, M. Mislove, and D.S. Scott. A Compendium of Continous Lattices. Springer-Verlag, Berlin, 1980.

[GKM83] J.V. Guttag, D. Kapur, and D.R. Musser. On proving uniform termination and restricted termination of rewriting systems. *SIAM Journal of Computing*, 12(1):189-214, 1983.

[GKM87] J. Goguen, C. Kirchner, and J. Meseguer. Concurrent Term Rewriting as a Model of Computation. In J.H. Fasel and R.M. Keller, editors, *Graph Reduction. Proceedings of a Workshop*, LNCS 279:53-93, Springer-Verlag, Berlin, 1987.

[GLMP91] E. Giovanneti, G. Levi, C. Moiso, and C. Palamidessi. Kernel LEAF: A Logic plus Functional Language. *Journal of Computer and System Sciences* 42(2):139-185, 1991.

[Gra96] B. Gramlich. Termination and Confluence Properties of Structured Rewrite Systems. PhD thesis, Fachbereich Informatik, Universität Kaiserlautern, January 1996.

[GTW78] J.A. Goguen, J.W. Thatcher, and E.G. Wagner. An Initial Algebra Approach to the Specification, Correctness, and Implementation of Abstract Data Types. In R.T. Yeh, editor, *Current Trends in Programming Methodology, vol. IV: data structuring*, pages 80-149, Prentice-Hall, Englewood Cliffs, NJ, 1978.

[Gue81] I. Guessarian. Algebraic Semantics. LNCS 99, Springer-Verlag, Berlin, 1981.

[GW92] H. Ganzinger and U. Waldmann. Termination Proofs of Well-Moded Logic Programs via Conditional Rewrite Systems. In M. Rusinowitch and J.L. Rémy, editors, *Proc. of 3rd International Workshop on Conditional Term Rewriting Systems, CTRS'92*, LNCS 656:430-437, Springer-Verlag, Berlin, 1993.

[GWMFJ93] J.A. Goguen, T. Winkler, J. Meseguer, K. Futatsugi, and J.-P. Jouannaud. Introducig OBJ*. In D. Coleman, R. Gallimore, and J. Goguen, editors, *Applications of Algebraic Specifications Using OBJ*, 1993.

[Han94] M. Hanus. The integration of functions into logic programming: From theory to practice. *Journal of Logic Programming* 19&20:583-628, 1994.

[Han95] M. Hanus. Analysis of residuating programs. *Journal of Logic Programming*, 24(3):219-245, 1995.

[Han95b] M. Hanus. On Extra Variables in (Equational) Logic Programming. In *Proc. of the 12th International Conference on Logic Programming, ICLP'95*, pages 665-679, The MIT Press, Cambridge, MA, 1995.

[Han95c] M. Hanus. Functional Logic Languages: Combine Search and Efficient Evaluation. In J.W. Lloyd, editor, *Proc. of the 1995 International Symposium on Logic Programming, ILPS'95*, pages 625-626, The MIT Press, Cambridge, MA, 1995.

[Han97] M. Hanus. Lazy Narrowing with Simplification. *Journal of Computer Languages*, 23(2-4):61-85, 1997.

[HL78] G. Huet and D.S. Lankford. On the Uniform Halting Problem for Term Rewriting Systems. Technical Report 283, IRIA Laboria, LeChesnay, France, 1978.

[HL79] G. Huet and J.J. Lévy. Call by need computations in nonambiguous linear term rewriting systems. Technical Report 359, IRIA Laboria, LeChesnay, France, 1979.

[HL91] G. Huet and J.J. Lévy. Computations in orthogonal term rewriting systems I, II. In J.L. Lassez and G. Plotkin, editors, *Computational logic: essays in honour of J. Alan Robinson*, pages 395-414 and 415-443. The MIT Press, Cambridge, MA, 1991.

[HLM98] M. Hanus, S. Lucas, and A. Middeldorp. Strongly sequential and inductively sequential term rewriting systems. *Information Processing Letters*, 67(1):1-8, 1998.

[HM76] P. Henderson and J. Morris, Jr. A Lazy Evaluator. In *Conference record of the 3rd. Annual ACM Symposium on Principles of Programming Languages, POPL'76*, pages 95-103, ACM Press, 1976.

[HO79] C.M. Hoffmann and M.J. O'Donnell. An interpreter generator using tree pattern matching. In *Conference Record of the 6th Annual ACM Symposium on Principles of Programming Languages, POPL'79*, pages 169-179, ACM Press, 1979.

[HO82] C.M. Hoffmann and M.J. O'Donnell. Pattern matching in trees. *Journal of the ACM* 29(1):68-95, 1982.

[HO82b] C.M. Hoffmann and M.J. O'Donnell. Programming with equations. *ACM Transactions on Programming Languages and Systems* 4(1):83-112, 1982.

[Höl89] S. Hölldobler. Foundations of Equational Logic Programming. LNAI 353, Springer-Verlag, Berlin, 1989.

[HU79] J.E. Hopcroft and J.D. Ullman. Introduction to automata theory, languages, and computation. Addison-Wesley Publishing Company, 1979.

[Hue80] G. Huet. Confluent reductions: abstract properties and applications to term rewriting systems. *Journal of the ACM* 27:797-821, 1980.

[Hul80] J.M. Hullot. Canonical Forms and Unification. In *Proc. of 5th Conference on Automated Deduction, CADE'80*, LNCS 87:318-334, Springer-Verlag, Berlin, 1980.

[HW87] C.V. Hall and D.S. Wise. Compiling Strictness into Streams. In *Conference Record of the 14th Annual ACM Symposium on Principles of Programming Languages, POPL'87*, pages 132-143, ACM Press, 1987.

[Jac96] F. Jacquemard. Decidable approximations to Term Rewriting Systems. In H. Ganzinger, editor, *Proc. of 7th International Conference on Rewriting Techniques and Applications, RTA'96*, LNCS 1103:362-376, Springer-Verlag, Berlin, 1996.

[Jon92] M.P. Jones. Introduction to Gofer 2.20. Oxford Programming Research Group, 1992.

[Jou94] J.-P. Jouannaud. Rewrite Proofs and Computations. In H. Schwichtenberg, editor, *Proof and Computation*, NATO Series F: Computer and Sciences, 139:173-218, Springer-Verlag, Berlin, 1994.

[JS94] J.-P. Jouannaud and W. Sadfi. Strong Sequentiality of Left-Linear Overlapping Rewrite Systems. In N. Dershowitz and N. Lindenstrauss, editors, *Proc. of 4th Workshop on Conditional Term Rewriting Systems, CTRS'94*, LNCS 968:235-246, Springer-Verlag, Berlin, 1995.

[KP78] G. Kahn et G. Plotkin. Domains concrets. Technical Report 336, IRIA Laboria, LeChesnay, France, 1978.

[KP93] G. Kahn and G. Plotkin. Concrete Domains. *Theoretical Computer Science*, 121:187-277, 1993.

[KW95] J.F.Th. Kamperman and H.R. Walters. Lazy Rewriting and Eager Machinery. In J. Hsiang, editor, *Proc. of 6th International Conference on Rewriting Techniques and Applications, RTA'95*, LNCS 914:147-162, Springer-Verlag, Berlin, 1995.

[Kap84] S. Kaplan. Conditional Rewrite Rules. *Theoretical Computer Science* 33:175-193, 1984.

[Kel75] J. Kelley. General Topology. van Nostrand, New York, 1955. Versión española: Topología General. EUDEBA, Buenos Aires, 1975.

[Ken89] R. Kennaway. Sequential Evaluation Strategies for Parallel-Or and Related Reduction Systems. *Journal of Pure and Applied Algebra*, 43:31-56, 1989.

[Ken94] R. Kennaway. A Conflict Between Call-By-Need Computation and Parallelism. In N. Dershowitz and N. Lindenstrauss, editors, *Proc.*

of 4th Workshop on Conditional Term Rewriting Systems, CTRS'94, LNCS 968:247-261, Springer-Verlag, Berlin, 1994.

[KKS91] M.R.K. Krishna Rao, D. Kapur, and R.K. Shyamasundar. A Transformational Methodology for Proving Termination of Logic Programs. In E. Börger, G. Jäger, H. Kleine Büning, and M.M. Richter, editors, *Proc. of 5th Workshop on Computer Science Logic, CSL'91*, LNCS 626:213-226, Springer-Verlag, Berlin, 1992.

[KKSV95] R. Kennaway, J.W. Klop, M.R. Sleep, and F.-J. de Vries. Transfinite reductions in Orthogonal Term Rewriting Systems. *Information and Computation* 119(1):18-38, 1995.

[Klo92] J.W. Klop. Term Rewriting Systems. In S. Abramsky, D.M. Gabbay, and T.S.E. Maibaum. *Handbook of Logic in Computer Science*, volume 2, pages 1-116. Oxford University Press, 1992.

[KM91] J.W. Klop and A. Middeldorp. Sequentiality in Orthogonal Term Rewriting Systems. *Journal of Symbolic Computation* 12:161-195, 1991.

[KOV96] R. Kennaway, V. van Oostrom, and F.J. de Vries. Meaningless Terms in Rewriting. In M. Hanus and M. Rodríguez-Artalejo, editors, *Proc. of 5th International Conference on Algebraic and Logic Programming, ALP'96*, LNCS 1139:254-268, Springer-Verlag, Berlin, 1996.

[KPRRS92] O. Kaser, S. Pawagi, C.R. Ramakrishnan, I.V. Ramakrishnan, and R.C. Sekar. Fast Parallel Implementation of Lazy Languages - The EQUALS Experience. In *Proc. of ACM Conference on Lisp and Functional Programming, LFP'92*, ACM Press, 1992.

[Lal93] R. Lalement. Computation as Logic. Masson, Paris-Prentice Hall International, London, 1993.

[Lin88] G. Lindstrom. Static Analysis of Functional Programs with Logical Variables. In P. Deransart, B. Lorho, and J. Maluszynski, editors, *Proc. of International Workshop on Programming Languages Implementation and Logic Programming, PLILP'88*, LNCS 348:1-19, Springer-Verlag, Berlin, 1989.

[Llo98] J.W. Lloyd. Programming in an Integrated Functional and Logic Language. *Journal of Functional and Logic Programming*, to appear.

[LLR93] R. Loogen, F. López-Fraguas, and M. Rodríguez Artalejo. A Demand Driven Computation Strategy for Lazy Narrowing. In M. Bruynooghe and J. Penjam, editors, *Proc. of 5th International Symposium*

on Programming Language Implementation and Logic Programming, PLILP'93, LNCS 714:184-200, Springer-Verlag, Berlin, 1993.

[LO96] S. Lucas and J. Oliver. A new proposal of Concurrent Process Calculus. In K.G. Jeffery, J. Král, and M. Bartŏsek, editors, *Proc. of XXIII Seminar on Current Trends in Theory and Practice of Informatics, SOFSEM'96*, LNCS 1175:385-392, Springer-Verlag, Berlin, 1996.

[Luc95] S. Lucas. Fundamentals of context-sensitive rewriting. In M. Bartŏsek, J. Staudek, and J. Wiedermann, editors, *Proc. of XXII Seminar on Current Trends in Theory and Practice of Informatics, SOFSEM'95*, LNCS 1012:405-412, Springer-Verlag, Berlin, 1995.

[Luc96] S. Lucas. Context-Sensitive Computations in Confluent Programs. In H. Kuchen and S.D. Swierstra, editors, *Proc. of 8th International Symposium on Programming Languages, Implementations, Logics, and Programs, PLILP'96*, LNCS 1140:408-422, Springer-Verlag, Berlin, 1996.

[Luc96b] S. Lucas. Termination of Context-Sensitive Rewriting by Rewriting. In F. Meyer auf der Heide and B. Monien, editors, *Proc. of 23rd. International Colloquium on Automata, Languages and Programming, ICALP'96*, LNCS 1099:122-133, Springer-Verlag, Berlin, 1996.

[Luc96c] S. Lucas. Using replacement restrictions in functional-logic languages. Fifth Compulog-Network Area Meeting on Language Design and Semantic Analysis Methods. Università degli Studi di Pisa, 1996.

[Luc97] S. Lucas. Computational properties of term rewriting with replacement restrictions. In M. Falaschi, M. Navarro, and A. Policriti, editors, *Proc. of 1997 Joint Conference on Declarative Programming, APPIA-GULP-PRODE'97*, pages 393-404, Università di Udine, 1997.

[Luc97b] S. Lucas. Efficient Strong Sequentiality Using Replacement Restrictions. In F. Plášil and K. Jeffery, editors, *Proc. of XXIV Seminar on Current Trends in Theory and Practice of Informatics, SOFSEM'97*, LNCS 1338:489-496, Springer-Verlag, Berlin, 1997.

[Luc97c] S. Lucas. Needed Reductions with Context-Sensitive Rewriting. In M. Hanus, J. Heering, and K. Meinke, editors, *Proc. of 6th International Conference on Algebraic and Logic Programming, ALP'97*, LNCS 1298:129-143, Springer-Verlag, Berlin, 1997.

[Luc97d] S. Lucas. Transformations for Efficient Evaluations in Functional Programming. In H. Glaser and P. Hartel, editors, *Proc of 9th International Symposium on Programming Languages, Implementations, Logics and Programs, PLILP'97*, LNCS 1292:127-141, Springer-Verlag, Berlin, 1997.

[Luc98] S. Lucas. Context-sensitive computations in functional and functional logic programs. *Journal of Functional and Logic Programming*, 1998(1), January 1998.

[Luc98b] S. Lucas. Root-neededness and approximations of neededness. *Information Processing Letters*, 67(5):245-254, 1998.

[Man74] Z. Manna. Mathematical Theory of Computation. McGraw-Hill, New York, 1974.

[Man74b] M.J. Mansfield. Introduction to Topology. Litton Educational Publishing, Inc, 1963. Versión española: Introducción a la Topología. Alhambra, Madrid, 1974.

[Mar90] L. Maranget. Optimal Derivations in Weak Lambda-calculi and in Orthogonal Term Rewriting Systems. In *Conference Record of the 18th Annual ACM Symposium on Principles of Programming Languages, POPL'90*, pages 255-269, ACM Press, 1990.

[Mar94] M. Marchiori. Logic Programs and Term Rewriting Systems. In G. Levi and M. Rodríguez-Artalejo, editors, *Proc. of 4th International Conference on Algebraic and Logic Programming, ALP'94*, LNCS 850:223-241, Springer-Verlag, Berlin, 1994.

[Mar96] M. Marchiori. Unravelings and Ultra-properties. In M. Hanus and M. Rodríguez-Artalejo, editors, *Proc. of 5th International Conference on Algebraic and Logic Programming, ALP'96*, LNCS 1039:107-121, Springer-Verlag, Berlin, 1996.

[McC60] J. McCarthy. Recursive Functions of Symbolic Expressions and their Computations by Machine, Part I. *Communications of the ACM*, 3(4), 1960.

[McC78] J. McCarthy. History of Lisp. In R.L. Wexelblat, editor, *History of Programming Languages*, Academic Press, New York, 1978.

[MH94] A. Middeldorp and E. Hamoen. Completeness Results for Basic Narrowing. *Applicable Algebra in Engineering, Communication and Computing*, 5:213-253, 1994.

[Mid97] A. Middeldorp. Call by Need Computations to Root-Stable Form. In *Conference Record of the 24th Annual ACM Symposium on Principles of Programming Languages, POPL'97*, pages 94-105, ACM Press, 1997.

[Mil91] R. Milner. The polyadic π-calculus: A tutorial. In F.L. Brauer, W Bauer and H Schwintenberg, editors, *Logic and Algebra of Specifications*, Springer-Verlag, Berlin, 1993.

[MKMWH93] J.J. Moreno-Navarro, H. Kuchen, J. Mariño, S. Winkler and W. Hans. Efficient Lazy Narrowing using Demandedness Analysis. In M. Bruynooghe and J. Penjam, editors, *Proc. of 5th International Symposium on Programming Language Implementation and Logic Programming, PLILP'93*, LNCS 714:167-183, Springer-Verlag, Berlin, 1993.

[MN92] A. Mycroft and A.C. Norman. Optimising Compilation. Part II: lazy functional languages. In *Proc. of XIX Seminar on Current Trends in Theory and Practice of Informatics, SOFSEM'92*, 1992.

[Möl85] B. Möller. On the Algebraic Specification of Infinite Objects - Ordered and Continuous Models of Algebraic Types. *Acta Informatica*, 22:537-578, 1985.

[MR92] J.J. Moreno-Navarro and M. Rodríguez-Artalejo. Logic programming with functions and predicates: the language BABEL. *Journal of Logic Programming*, 12:191-223, 1992.

[MS97] A. Miniussi and R. Strandh. An Efficient Algorithm for Recognizing the Forward-Branching Class of Term-Rewriting Systems. In M. Falaschi, M. Navarro, and A. Policriti, editors, *Proc. of 1997 Joint Conference on Declarative Programming, APPIA-GULP-PRODE'97*, pages 445-456, Università di Udine, 1997.

[Myc80] A. Mycroft. The theory and practice of transforming call-by-need into call-by-value. In B. Robinet, editor, *Proc. of the 4th International Symposium on Programming*, LNCS 83:269-281, Springer-Verlag, Berlin, 1980.

[MZ97] A. Middeldorp and H. Zantema. Simple Termination of Rewrite Systems. *Theoretical Computer Science*, 175:127-158, 1997.

[New42] M.H.A. Newman. On theories with a combinatorial definition of 'equivalence'. *Ann. Math.*, 43:223-243, 1942.

[Niv75] M. Nivat. On the Interpretation of Recursive Program Schemes. *Symposia Mathematica*, vol. 15, pages 255-281, Academic Press, 1975.

[NSEP92] E.G.J.M.H. Nöcker, J.E.W. Smetsers, M.C.J.D. van Eekelen, and M.J. Plasmeijer. Concurrent Clean. In E.H.L. Aarts, J. Leeuwen, and M. Rem, editors, *Proc. of Parallel Architectures and Languages Europe, PARLE'91*, LNCS 506:202-219, Springer-Verlag, Berlin, 1991.

[NWE97] N. Nedjah, C.D. Walter, and S.E. Elridge. Optimal Left-to-Right Pattern-Matching Automata. In M. Hanus, J. Heering, and K. Meinke, editors, *Proc. of 6th International Conference on Algebraic and Logic Programming, ALP'97*, LNCS 1298:273-286, Springer-Verlag, Berlin, 1997.

[O'D77] M.J. O'Donnell. Computing in Systems Described by Equations. LNCS 58, Springer-Verlag, Berlin, 1977.

[O'D85] M.J. O'Donnell. Equational Logic as a Programming Language. The MIT Press, Cambridge, MA, 1985.

[O'D87] M.J. O'Donnell. Term Rewriting Implementation of Equational Logic Programming. In P. Lescanne, editor, *Proc. of 2nd International Conference on Rewriting Techniques and Applications, RTA'87*, LNCS 256:1-12, Springer-Verlag, Berlin, 1987.

[O'D95] M.J. O'Donnell. Equational Logic Programming. In C.J. Hogger, D.M. Gabbay, and J.A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 4, chapter 3. Oxford University Press, 1995.

[OF97] K. Ogata and K. Futatsugi. Implementation of Term Rewritings with the Evaluation Strategy. In H. Glaser and P. Hartel, editors, *Proc of 9th International Symposium on Programming Languages, Implementations, Logics and Programs, PLILP'97*, LNCS 1292:225-239, Springer-Verlag, Berlin, 1997.

[Oli96] J. Oliver. Extensión del λ-cálculo para la Modelización de Procesos Concurrentes. PhD thesis, Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Septiembre 1996.

[Oya87] M. Oyamaguchi. The Church-Rosser property for ground term rewriting systems is decidable. *Theoretical Computer Science*, 49(1):43-79, 1987.

[Oya93] M. Oyamaguchi. NV-Sequentiality: a decidable condition for call-by-need computations in term-rewriting systems. *SIAM Journal of Computing*, 22(1):114-135, 1993.

[PE93] R. Plasmeijer and M. van Eekelen. Functional Programming and Parallel Graph Rewriting. Addison-Wesley, Reading, MA, 1993.

[Pey87] S.L. Peyton-Jones. The Implementation of Functional Programming Languages. Prentice-Hall International, London, 1987.

[PH97] J. Peterson and K. Hammond, editors. Report on the Programming Language Haskell. A Non-strict, Purely Functional Language. Version 1.4. *Yale University, Department of Computer Science* Tech Report, April 1997.

[Pla94] D. Plaisted. Equational Reasoning and Term Rewriting Systems. In D. Gabbay and J. Siekmann, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, Oxford University Press, volume 1, 1994.

[Rea93] C. Reade. Elements of Functional Programming. Addison-Wesley, Reading, MA, 1993.

[Red85] U.S. Reddy. Narrowing as the Operational Semantics of Functional Languages. In *Proc. of IEEE International Symposium on Logic Programming*, pages 138-151, IEEE Computer Society Press, 1985.

[Red95] U.S. Reddy. Bridging the gap between logic and functional programming. In J.W. Lloyd, editor, *Proc. of the 1995 International Symposium on Logic Programming, ILPS'95*, pages 627-628, The MIT Press, Cambridge, MA, 1995.

[Rod95] M. Rodríguez-Artalejo. The Challenge of Declarative Programming. In J.W. Lloyd, editor, *Proc. of the 1995 International Symposium on Logic Programming, ILPS'95*, pages 629-630, The MIT Press, Cambridge, MA, 1995.

[Ros73] B.K. Rosen. Tree manipulating systems and Church-Rosser Theorems. *Journal of the ACM*, 20(1):160-187, January 1973.

[Roy94] P. van Roy. 1983-1993: The wonder years of sequential Prolog implementation. *Journal of Logic Programming*, 19& 20:385-441, 1994.

[RS84] F. Reinhardt und H. Soeder. DTV - Atlas zur Mathematik - Band 1 - Grundlagen, Algebra und Geometrie. Deutscher Taschenbuch Verlag GmbH & Co. KG. München, 1974. Versión española: Atlas de Matemáticas, 1. Fundamentos, álgebra y geometría. Alianza Editorial, Madrid, 1984.

[Rus86] M. Rusinowitch. Demonstration automatique par des techniques de réécriture. PhD thesis, Nancy, October, 1987.

[Sal95] B. Salinier. Simulation de systèmes de réécriture de termes par des systèmes constructeurs. PhD thesis, Bourdeaux, December 1995.

[Sco70] D. Scott. Outline of a Mathematical Theory of Computation. Technical Monograph PRG-2, November 1970, Oxford University Computing Laboratory, 1970.

[Sco82] D. Scott. Domains for Denotational Semantics. In M. Nielsen and E.M. Schmidt, editors, *Proc. of International Colloquium on Automata, Languages and Programming, ICALP'82*, LNCS 140:577-613, Springer-Verlag, Berlin, 1982.

[She94] D.J. Sherman. Run-time and compile-time improvements to equational programs. PhD thesis, Department of Computer Science, The University of Chicago, June 1994.

[Sla74] J.R. Slagle. Automated theorem-proving for theories with simplifiers, commutativity, and associativity. *Journal of the ACM*, 21(4):622-642, 1974.

[SLG94] V. Stoltenberg-Hansen, I. Lindström, and E.R. Griffor. Mathematical Theory of Domains. Cambridge University Press, Cambridge, 1994.

[SMI95] T. Suzuki, A. Middeldorp, and T. Ida. Level-Confluence of Conditional Rewrite Systems with Extra Variables in Right-Hand Sides. In J. Hsiang, editor, *Proc. of 6th International Conference on Rewriting Techniques and Applications, RTA'95*, LNCS 914:179-193, Springer-Verlag, Berlin, 1995.

[Smu71] R. Smullyan. Theory of Formal Systems. Princeton University Press, N.J., 1971. Revised Edition.

[SPR89] R.C. Sekar, S. Pawagi and I.V. Ramakrishnan. Transforming Strongly Sequential Rewrite Systems with Constructors for Efficient Parallel Execution. In N. Dershowitz, editor, *Proc. of 3rd. International Conference on Rewriting Techniques and Applications, RTA'89*, LNCS 355:404-418, Springer-Verlag, Berlin, 1989.

[SPR90] R.C. Sekar, S. Pawagi and I.V. Ramakrishnan. Small Domains Spell Fast Strictness Analysis. In *Conference Record of the 16th Annual ACM Symposium on Principles of Programming Languages, POPL'89*, pages 169-183, ACM Press, 1990.

[SR93] R.C. Sekar and I.V. Ramakrishnan. Programming in Equational Logic: Beyond Strong Sequentiality. *Information and Computation* 104:78-109, 1993.

[SRR95] R. Sekar, R.C. Ramesh, and I.V. Ramakrishnan. Adaptative Pattern-Matching. *SIAM Journal on Computing*, 24(5):1207-1234, 1995.

[Str88] R. Strandh. Compiling Equational Programs into Efficient Machine Code. PhD thesis, Johns Hopkins University, Baltimore, Maryland, 1988.

[Str89] R. Strandh. Classes of Equational Programs that Compile into Efficient Machine Code. In N. Dershowitz, editor, *Proc. of 3rd International Conference on Rewriting Techniques and Applications, RTA'89*, LNCS 355:449-461, Springer-Verlag, Berlin, 1989.

[Tha85] S. Thatte. On The Correspondence Between Two Classes of Reduction Systems. *Information Processing Letters*, 20:83-85, 1985.

[Tha87] S. Thatte. A Refinement of Strong Sequentiality for Term Rewriting with Constructors. *Information and Computation*, 72:46-65, 1987.

[Toy92] Y. Toyama. Strong Sequentiality of Left-Linear Overlapping Rewrite Systems. In *Proc. of 7th Annual IEEE Symposium on Logic in Computer Science, LICS'92*, pages 274-284, IEEE Computer Society Press, 1992.

[TSEP93] Y. Toyama, S. Smetsers, M.C.J.D. van Eekelen and R. Plasmeijer. The Functional Strategy and Transitive Term Rewriting Systems. In M.R. Sleep, M.J. Plasmeijer and M.C.J.D. van Eekelen, editors, *Term Graph Rewriting - Theory and Practice*, pages 117-129, John Wiley & sons, 1993.

[Vui74] J. Vuillemin. Correct and optimal implementation of recursion in a simple programming language. *Journal of Computer and System Sciences* 9(3):332-354, 1974.

[Wad87] P. Wadler. Strictness Analysis on Non-Flat Domains (by Abstract Interpretation over Finite Domains). In S. Abramsky and C. Hankin, editors, *Abstract Interpretation of Declarative Languages*, pages 266-275, Ellis Horwood Ltd., John Wiley and sons, 1987.

[WH89] P.H. Winston and B.K.P. Horn. Lisp. Third Edition. Addison-Wesley, Reading, MA, 1989.

[Wir95] C.P. Wirth. Syntactic Confluence Criteria for Positive/Negative - Conditional Term Rewriting Systems. SEKI-Report SR-95-09, Universität Kaiserlautern, 1995.

[WWT78] J.B. Wright, E.G. Wagner, and J.W. Thatcher. A uniform approach to inductive posets and inductive closure. *Theoretical Computer Science*, 7:57-77, 1978.

[Zan94] H. Zantema. Termination of term rewriting: interpretation and type elimination. *Journal of Symbolic Computation* 17:33-50, 1994.

[Zan95] H. Zantema. Termination of Term Rewriting by Semantic Labelling. *Fundamenta Informaticae* 20:43-60, 1995.

[Zan97] H. Zantema. Termination of Context-Sensitive Rewriting. In H. Comon, editor, *Proc. of 8th International Conference on Rewriting Techniques and Applications, RTA'97*, LNCS 1232:172-186, Springer-Verlag, Berlin, 1997.

Índice de Materias

análisis de *strictness*, 5
anotaciones
    de *strictness*, 5, 61, 191
    en reescritura, 6, 61
    globales, 5, 61
    locales, 61
    sintácticas, 5
árbol definicional, 165
argumento redundante, 284
autómata finito, 166

cadena, 19
call-by-name, 59
call-by-need, 32
computación segura, 58
confluencia, 105
conjunto
    acotado inferiormente, 19
    acotado superiormente, 19
    estrictamente ordenado, 19
    ordenado, 19
        maximal de un, 19
        minimal de un, 19
conjunto cociente, 18
conjunto necesario de redexes
    para la estabilización, 37
    para la normalización, 36
conjuntos, 15
    diferencia de, 15
    intersección de, 15
    unión de, 15
cono, 19
    inverso, 19
contexto, 24
    rígido, 39
contexto constructor máximo, 194
contexto reemplazante máximo, 51, 222
cota
    inferior, 19
    superior, 19
cpo, *ver* orden parcial completo
cubrimiento, 26

derivación de reescritura, 25
derivación residual, 29
disciplina de constructores, 27
distancia, 18
    entre términos, 29

ecuación, 240
elementos compatibles, 19
emparejamiento, 22
esc, *ver* estrechamiento sensible al contexto
espacio métrico, 18
    completo, 18
estabilización, 27, 225
estrategia
    E, 63, 97, 141, 182, 229
    E regular, 63
    local, 141
estrategia de reescritura, 2, 31
    de normalización infinita, 233
    descomposición de una, 33

determinista, 31
estabilizante, 32
implementación de una, 163
independiente del contexto, 34
infinitamente normalizante, 34
infinitamente normalizante y equitativa, 35
leftmost-innermost, 144
leftmost-outermost, 31, 144, 173
monopaso, 31
multipaso, 31
normalizante, 32, 127, 234
paralela, 32
parallel-innermost, 144
parallel-outermost, 31, 144, 233
secuencial, 32, 122, 127
sin anticipación, 33
estrategia de reescritura genérica, 143, 144
de reducción de índices, 150, 173
estabilizante, 144
estructural, 146, 147
normalizante, 144
restricción *sc* de una, 149
estrategia genérica, *ver* estrategia de reescritura genérica
estrategia sensible al contexto, 121, 122
de reducción de índices, 137
descomposición de una, 122
genérica, 143, 149
sc-normalizante, 149
de reducción de índices, 160
estructural, 149
inducida, 158
implementación de una, 163
inducida, 159
implementación de una, 167
μ-normalizante, 234
estrategias
locales, 6, 62
perezosas, 3
estrechamiento, 1, 7, 237, 240
con normalización, 284
con simplificación, 284
restricciones de instanciación en el, 284
restricciones de reemplazamiento en el, 238, 284
estrechamiento sensible al contexto, 238
corrección y completitud del, 240
evaluación, 2, 187, 189, 194
estricta, 3
impaciente, 3
no estricta, 3
perezosa, 3
por reescritura sensible al contexto, 189
voraz, 3
evaluación parcial, 284
expresión canónica, 1
extensión sensible al contexto, 155
extensión *sc*, *ver* extensión sensible al contexto

forma
E-normal, 63
estable, 2, 27, 219, 220, 222
constructora, 27, 189, 194
fuertemente estable, 2, 38, 222
μ-normal, 54, 77, 222, 223
normal, 2, 28, 219, 220, 222, 223
en cabeza, 27
normal infinita, 2, 231
R-normal, 17
forma canónica, 27, 276
función, 16
biyectiva (biyección), 16
continua, 20
estricta, 20

idempotente, 16
identidad, 16
inversa, 16
inyectiva, 16
monótona, 20
parcial, 16
potencia de una, 16
punto fijo de una, 16
restricción de una, 16
sobreyectiva, 16
función de descomposición, 246
función de fallo, 166, 167
función de reemplazamiento, 45
canónica, 70
con anotaciones de *strictness*, 72, 191
de evaluación, 189
normal, 247
función de selección, 146
ejemplo de, 147
función de transferencia, 166

γ-confluencia, 275, 278
γ-reescritura, 252
γ-terminación, 275

hueco, 24

igualdad estricta, 241
index point, 166
index tree, 165–167, 180
forward branching, 165, 180
índice
fuertemente secuencial, 40, 128
secuencial, 40, 128
transitivo, 254
ínfimo, 19

λ-cálculo, 59
lenguajes de programación
algebraicos, 1, 6, 62
ecuacionales, 1
funcionales, 1
perezosos, 5, 61
imperativos, 145
lógico-funcionales, 1, 7, 237
lógicos, 1, 145, 237
límite, 18

métrica, 18
compleción, 18
matching dag, 164, 165, 167
maximal, 19
máximo, 19
mecanismo de reescritura, *ver* reescritura, mecanismo de
minimal, 19
mínimo, 19
modularidad, 284
μ-confluencia, 106, 115
μ-contracción
de sistemas de reescritura, 89
función de, 87
μ-derivación, 53
μ-reescritura, 52
μ-terminación, 84, 232
demostración de la, 94
multiderivación, 28
elemental, 28

narrowing, *ver* estrechamiento
normalización, 225
infinita, 232

ocurrencia, 21
reemplazante, 48
Ω-reducción, 37
Ω-término, 21
operación, 27
operador de cierre, 20
orden, 19
bien fundado, 19

parcial, 19
parcial completo, 19
total, 19
orden sobre funciones de reemplazamiento, 46
orden sobre restricciones de reemplazamiento, 258

par crítico, 26
convergente, 26
trivial, 26
uniformemente convergente a la raíz, 42
par μ-crítico, 107
μ-convergente, 108
par válido, 149
parallel moves lemma, 29
y confluencia fuerte, 108, 111
π-cálculo, 59
precedencia, 30
preredex, 25
estricto, 25
procedimiento $norm_\mu$, 225
completitud del, 227
corrección del, 226
proceso de reescritura, *ver* reescritura, mecanismo de
profundidad
de un subtérmino, 22
de un paso de reducción condicional, 245
programación funcional, 1, 187, 192, 195, 216
programación lógica, 1
punto de recuperación, 167
inmediato, 167
punto fijo, 16
técnicas de, 198
teorema del, 20

rcsc, *ver* reescritura condicional sensible al contexto
recursive path ordering, 30
redex, 25
esquema de, 25
fuertemente necesario, 40
fuertemente secuencial, 40
necesario
para la estabilización, 36
para la normalización, 35, 122
reemplazante, 54
strongly root-needed, 182
reducción condicional, 60, 116, 229
reducción necesaria, 32
para la estabilización, 121
para la normalización, 121–123
reducto, 25
reescritura, 1, 25
derivación de, 25
infinita, 29, 231
mecanismo de, 66, 163
paralela, 28
perezosa, 61, 117, 217, 229
regla de, 24
relación de, 25
restringida, 252
secuencia de, 25
semántica de, 276
reescritura condicional sensible al contexto, 245
reescritura sensible al contexto, 3, 4, 52
implementación de la, 171
y anotaciones sintácticas, 6
y estrechamiento, 7, 238
y reescritura, 70
y sistemas de reescritura condicionales, 7, 245
y teorías AC, 7, 249

relación, 17
  antisimétrica, 17
  cierre de una, 17
  confluente, 17
  de equivalencia, 18
  de orden, 19
  fuertemente confluente, 17
  irreflexiva, 17
  localmente confluente, 17
  reflexiva, 17
  simétrica, 17
  terminante, 17
  transitiva, 17
relaciones
  composición de, 17
  homomorfismo entre, 17
renombramiento, 22
residuo, 28
restricción de reemplazamiento
  aproximación de una, 259
  aproximación mediante restricciones sensibles al contexto, 261, 267, 272
  completitud de una, 276
  descomposición de una, 264
  propiedades básicas, 253, 256
    COMP, 254
    CONTEXT, 255
    DECOMP, 254
    DOWN, 254
    REPL, 255
    ROOT, 253
    SUBST, 256
    $\overline{\text{SUBST}}$, 256
    SUBTERM, 254
  sensible al contexto, 251, 264
  sintáctica, 2, 3, 5, 251, 252
restricción sensible al contexto, *ver* restricción de reemplazamiento sensible al contexto
restricción *sc* de una estrategia genérica, 149
retículo completo, 19
  de funciones de reemplazamiento, 46
  de restricciones de reemplazamiento, 258
  distributivo, 20
rpo, *ver* recursive path ordering
rsc, *ver* reescritura sensible al contexto

satisfacción condicional, 243, 244
secuencia
  convergente, 18
  de Cauchy, 18
  infinita, 18
  límite de una, 18
secuencia de reducción
  finita, 17
  infinita, 18
  necesaria, 35
secuencia de reescritura, 25
  equitativa, 34
  estabilizante, 28
  fuertemente convergente, 30
  infinita, 25, 231
  infinitamente normalizante, 30
  normalizante, 28
  perpetua, 34
secuencialidad, 32
secuencialidad fuerte, 41
  decidibilidad de la, 41, 128, 153
  y contracción de sistemas de reescritura, 151, 154
  y estabilización, 128
  y normalización, 42
semántica de reescritura, 276
signatura, 20
  μ-contracta, 87

símbolo
constructor, 27
definido, 27
definido externo, 195, 197, 207
sistema de inferencia, 59
sistema de reescritura, 24
μ-conservativo, 90
básico (GR), 26
basado en constructores, 27
bounded (B), 165
canónico, 30, 188
con variables extra, 24
con variables reemplazantes homogéneas por la izquierda, 106, 191
confluente, 30
cuasi ortogonal, 26, 128
débilmente ortogonal, 26
forward branching (FB), 165, 180
fuertemente secuencial (SS), 41, 128
infinitamente normalizante, 30
lineal por la izquierda (LL), 26
μ-canónico, 188
μ-normal por la izquierda, 174
no μ-confuso, 157
no confuso, 157
normal por la izquierda (LN), 26, 173
normalizante, 28, 224
ortogonal (ORT), 26, 122
sc-básico, 177
secuencial, 127
subyacente, 243
terminante, 30
transitivo (TR), 165
uniformemente convergente a la raíz, 42
sistema de reescritura condicional, 242
estándar, 244
normal, 244, 246
semi ecuacional, 244
SRT, *ver* sistema de reescritura
SRT BC, *ver* sistema de reescritura basado en constructores
SRTC, *ver* sistema de reescritura condicional
SRTE, *ver* sistema de reescritura con variables extra
strictness, 58, 72, 191
supremo, 19
sustitución, 22
básica, 22
de emparejamiento, 22
factible, 243
identidad, 22
vacía, 22

término, 21
μ-compatible, 70
básico, 21
constructor, 27
de primer orden, 21
estabilizable, 28
estable, 27
frágil, 39
fuertemente estable, 38
infinito, 29, 57
lineal, 23
μ-compatible, 67
normalizable, 28
profundidad de un subtérmino en un, 22
rígido, 39
reducible, 28
teoría AC, 249
teoría ecuacional condicional, 242
terminación, 30, 83
tipos algebraicos, 190, 195
transformación V, 205, 207
corrección y completitud de la, 215

desplegada, 217
tuplas, 16

ultrapropiedad, 246
umg, *ver* unificador más general
unificación, 22
unificador, 22
más general, 22, 238

valor, 27
en programación funcional, 27, 187, 216
infinito, 231