Universidad Nacional de Rosario

Tesina de Grado
para la obtención del grado de
Licenciado en Ciencias de la Computación

Confluencia en Sistemas de Reescritura Probabilista

Autor:
Guido Martínez

Director:
Dr. Alejandro Díaz-Caro

Departamento de Ciencias de la Computación
Facultad de Ciencias Exactas, Ingeniería y Agrimensura
Av. Pellegrini 250, Rosario, Santa Fe, Argentina

Entrega: 28 de febrero de 2017
Defensa: 27 de marzo de 2017
Última edición: 28 de abril de 2017
Versión: `v1-34-gb4e247f`

Resumen

Los sistemas de reescritura abstracta son la manera estándar de dar una semántica operacional (small-step) a un lenguaje de programación. Estudiarlos a nivel abstracto, olvidando los detalles de un cálculo en particular, da buenos frutos ya que muchas propiedades y teoremas son reusables entre lenguajes distintos.

Algunos lenguajes tienen una ejecución *probabilista*. Para modelar esto, existen nociones de sistemas de reescritura probabilista. Sin embargo, la definición más usual de los mismos no permite no determinismo en su reducción y, tal vez como consecuencia, propiedades como la confluencia en este tipo de sistemas están poco estudiadas.

Proponemos una noción de sistemas de reescritura *multiprobabilista* que permite no determinismo en la reducción, mostrando cómo generalizan a otras nociones pero permitiendo a la vez nuevos comportamientos. En particular, permiten modelar un lenguaje probabilista sin fijar una estrategia de reducción. Damos dos ejemplos de lenguajes concretos interesantes que pueden modelarse con estos sistemas.

Basándonos en trabajo previo, se define una noción de *confluencia de distribuciones*. Por el lado teórico, estudiamos esta noción abstractamente junto a sus consecuencias, simplificando los diagramas a demostrar y traspasando teoremas clásicos de confluencia a la nueva noción, obteniendo criterios simplificados para decidir si la propiedad se cumple. Además, damos evidencia de que nuestra representación es general, y debería ser ampliamente aplicable.

Por el lado concreto, brindamos dos resultados interesantes: (1) Se simplifica una demostración de una confluencia similar existente en la literatura (2) Se destilan las propiedades necesarias para tener confluencia, dando un cálculo simple a modo de ejemplo.

Índice general

Preliminares

Introducimos algunos conceptos previos, ajenos a la reescritura, para establecer la notación que usaremos.

Se asumen conocimientos matemáticos previos sobre teoría de conjuntos básica, relaciones, probabilidad, números complejos y definiciones de lenguajes por gramáticas. También, se asume familiaridad con definiciones inductivas y cómo razonar sobre ellas.

Relativo a los lenguajes de programación, asumimos familiaridad con el λ-cálculo, su semántica operacional y la operación de sustitución que evita capturas.

Listas

Notamos con $\mathscr{L}(A)$ al conjunto de *listas* sobre un conjunto base A. Las listas sobre A tienen la siguiente definición inductiva

$$\frac{}{[] \in \mathscr{L}(A)} \qquad \frac{x \in A \quad xs \in \mathscr{L}(A)}{x : xs \in \mathscr{L}(A)}$$

Como convención notacional, el operador de *construcción* ':' asociará a derecha. A veces usaremos la notación $[x_1, x_2, \cdots, x_n]$ para representar a la lista $x_1 : x_2 : \cdots : x_n : []$. Notar que cada lista es de longitud *finita*. A una lista de longitud 1 la llamamos *singleton*.

Definimos la *concatenación* de dos listas, por recursión en el primer argumento, con las siguientes ecuaciones

$$\begin{array}{rcl} [] +\!\!+ \ ys & = & ys \\ (x : xs) +\!\!+ \ ys & = & x : (xs +\!\!+ \ ys) \end{array}$$

Cuando queramos expresar una lista por comprensión usaremos la notación $[x_i]_i$ para notar a la lista

$$[x_1, x_2, \cdots, x_n]$$

donde la cantidad n no es relevante o puede ser inferida del contexto.

Distribuciones

Nos interesará en particular representar distribuciones de probabilidad sobre un conjunto A. Matemáticamente, son funciones $A \to [0,1]$ *normalizadas*: la suma de las probabilidades para todos los elementos posibles es exactamente 1. Las representaremos como listas. Esto fuerza a trabajar sólo con distribuciones con soporte finito, pero permite un análisis riguroso.

Definimos el conjunto de *distribuciones sobre* A, $\mathscr{D}(A)$, como $\mathscr{L}(\mathbb{R}^+ \times A)$. Interpretamos cada (p, a) en la lista como asignar la probabilidad p al elemento a. Sin embargo, no hay ninguna condición de que los elementos no se repitan o de que las probabilidades totales sumen 1. Esto es intencional.

Definimos el *peso* de una distribución como la suma de todas sus probabilidades, dado por las ecuaciones:

$$\begin{array}{lcl} w([]) & = & 0 \\ w((p,a) : ds) & = & p + w(ds) \end{array}$$

Cuando nos interese que el peso de una distribución sea 1, usaremos el tipo

$$\mathscr{D}_1(A) = \{d \in \mathscr{D}(A) \mid w(d) = 1\}$$

que representa a las distribuciones de peso 1 (o distribuciones normalizadas). A veces necesitaremos distribuciones normalizadas o nulas, para lo cual definimos el tipo

$$\mathscr{D}_{0,1}(A) = \{d \in \mathscr{D}(A) \mid d = [] \vee w(d) = 1\}$$

Representamos con αD a la distribución D escalada por un factor de $\alpha \in \mathbb{R}^+$, definida por las ecuaciones:

$$\begin{array}{lcl} \alpha[] & = & [] \\ \alpha((p,a) : ds) & = & (\alpha p, a) : \alpha ds \end{array}$$

Podemos extender una función $f : A \to B$ a distribuciones ($\hat{f} : \mathscr{D}(A) \to \mathscr{D}(B)$) de la siguiente manera:

$$\begin{array}{lcl} \hat{f}([]) & = & [] \\ \hat{f}((p,a) : ds) & = & (p, f(a)) : \hat{f}(ds) \end{array}$$

Capítulo 1

Introducción a la reescritura

"Good writing is mostly rewriting."

—Anónimo

En este capítulo se describe la noción tradicional de sistemas de reescritura, sus aplicaciones, y propiedades y teoremas relevantes. Estudiamos en particular la confluencia, mostrando algunas consecuencias de la misma y criterios para demostrarla.

Mostramos cómo los ARS son usados para modelar la semántica de lenguajes de programación tomando al λ-cálculo como ejemplo y dando una prueba de su confluencia.

Luego, introducimos la propiedad de *conmutación secuencial* que no es usual en el estudio de sistemas de reescritura pero será usada en los siguientes capítulos.

1.1. Sistemas de reescritura — ARS

El estudio de sistemas de reescritura abstracta (ARS) se concentra principalmente en las nociones abstractas de *reducción* o *transformación*. Ejemplos comunes son la ejecución paso a paso de una computación, la simplificación de una fórmula matemática o el recorrido paso a paso en un grafo dirigido.

Formalmente, un ARS tiene una definición muy simple:

> **Definición 1.1.** Un ARS (*abstract rewriting system*) $\mathcal{A}$ es un par $(A, \to)$ donde A es un conjunto llamado *portador*, y $\to$ es una relación binaria sobre A, llamada *relación de reducción* (o *de reescritura*) (es decir, $\to \in \mathcal{P}(A \times A)$.)

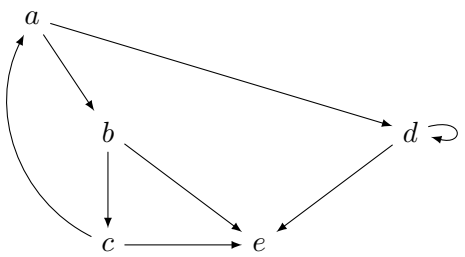
En general, usaremos la notación infija $a \to b$ (a leerse como "a reescribe (o reduce) a b en un paso") en vez de $(a, b) \in \to$. Si la relación es representada por una letra como R, usaremos $\to_R$ cuando se quiera usar de manera infija. Usaremos, en general, las letras R, S, T, α, β para denotar relaciones. Podemos definir un ARS listando todos los pares de elementos relacionados o de manera

gráfica. Al dar una definición extensional o gráfica, se entiende que las únicas reducciones posibles son las listadas, y que el portador es el mínimo conjunto que contiene a todos los elementos mencionados.

Ejemplo 1.2 (Un ARS)**.** Sea $\to$ definida por:

$$\begin{array}{llll} a \to b & a \to d & b \to c & b \to e \\ c \to a & c \to e & d \to d & d \to e \end{array}$$

o, de manera gráfica:



No hay ninguna otra restricción sobre un ARS, y por lo tanto cualquier relación binaria define un ARS. Sin embargo, en el estudio de la reescritura nos interesarán propiedades relacionadas a la idea de reducción, como son la *normalización* (Definiciones 1.7 y 1.8) o la *confluencia* (Definición 1.12)

En este capítulo, brindamos una breve introducción al área que debería ser autocontenida y suficiente para nuestros propósitos. Para un trato detallado, el lector puede referirse a los libros (TeReSe 2003) o (Baader y Nipkow 1998). En general, usaremos las convenciones notacionales de (Baader y Nipkow 1998).

Operaciones sobre relaciones

Dadas dos relaciones $R, S \subseteq A \times A$ podemos formar nuevas relaciones a partir de ellas. La *composición* se define como:

$$R \cdot S = \{(a, c) \mid \exists b.\, a \to_R b \wedge b \to_S c\}$$

Intuitivamente, $a \to_{R \cdot S} c$ cuando a puede ir a c reescribiendo vía R y luego vía S. También, podemos definir nuevas relaciones variando la cantidad de pasos o la dirección.

$\to^0$	$=$	$\{(x,x) \mid x \in A\}$	relación identidad
$\to^{n+1}$	$=$	$\to^n \cdot \to$	composición de $(n+1)$-pasos
$\to^{-1}$	$=$	$\{(b,a) \mid a \to b\}$	inversa
$\leftarrow$	$=$	$\to^{-1}$	inversa
$\to^=$	$=$	$\to \cup \to^0$	clausura reflexiva
$\to^+$	$=$	$\cup_{i \in \mathbb{N}^+} \to^i$	clausura transitiva
$\to^*$	$=$	$\to^+ \cup \to^0$	clausura reflexiva-transitiva
$\leftrightarrow$	$=$	$\to \cup \leftarrow$	clausura simétrica

La clausura reflexiva transitiva $\to^*$ también puede definirse inductivamente por las siguientes reglas de inferencia:

$$\frac{}{a \to^* a} \qquad \frac{a \to b \quad b \to^* c}{a \to^* c}$$

Ambas definiciones de $\to^*$ son equivalentes, y usaremos la más conveniente según el caso. Además, por razones de brevedad, solemos decir simplemente "clausura" para referirnos a la clausura reflexiva-transitiva.

Cuando $a \to^* b$ decimos que b es un *reducto* de a y a una *expansión* de b. Si $a \to^+ b$ decimos reducto *propio* y expansión *propia*, respectivamente.

Todas estas operaciones son *monótonas* respecto a la inclusión de conjuntos. Es decir, que si $\to_R \subseteq \to_S$ entonces tenemos $\to_R^n \subseteq \to_S^n$, $\leftrightarrow_R \subseteq \leftrightarrow_S$, etc. Notar que la inversión no es el complemento de la relación, sino su simetría.

Razonando sobre clausuras

Muchas veces trabajaremos con clausuras, y los lemas siguientes serán útiles. Cuando una relación $\to_R$ está contenida en $\to_S^*$, decimos que $\to_S$ *simula* a $\to_R$, en el sentido de que cualquier paso $\to_R$ puede hacerse vía $\to_S$ (en alguna cantidad de pasos). Esto implica que cualquier cantidad de pasos $\to_R$ puede hacerse vía $\to_S$.

Lema 1.3. *Si* $\to_R \subseteq \to_S^*$ *entonces* $\to_R^* \subseteq \to_S^*$

Demostración. Por nuestra hipótesis y la monotonía de la clausura reflexiva-transitiva tenemos $\to_R^* \subseteq \to_S^{**}$. Luego, dado que $\to_S^{**} = \to_S^*$, tenemos nuestro resultado. ■

Es fácil ver, entonces, que dos relaciones tienen la misma clausura cuando cada una simula a la otra.

Lema 1.4. $\to_R^* = \to_S^*$ *si y sólo si* $\to_R \subseteq \to_S^*$ *y* $\to_S \subseteq \to_R^*$

Demostración. **Ida.** Trivial. **Vuelta.** Usando el Lema 1.3. ■

Además, la igualdad entre clausuras se mantiene al extender las relaciones.

Lema 1.5. *Si* $\to_R^* = \to_S^*$, *entonces* $(\to_R \cup \to_T)^* = (\to_S \cup \to_T)^*$.

Demostración. Por doble contención usando el Lema 1.4. Para la ida, tenemos

$$
\begin{array}{ll}
\to_R & \to_T \\
\subseteq \{ \textit{ Def. clausura } \} & \subseteq \{ \textit{ Def. clausura } \} \\
\to_R^* & \to_T^* \\
= \{ \textit{ Hipótesis } \} & \subseteq \{ \textit{ Monotonía de la clausura } \} \\
\to_S^* & (\to_S \cup \to_T)^* \\
\subseteq \{ \textit{ Monotonía de la clausura } \} & \\
(\to_S \cup \to_T)^* &
\end{array}
$$

Entonces, $\to_R \cup \to_T \subseteq (\to_S \cup \to_T)^*$. Con el mismo razonamiento tenemos la otra inclusión y llegamos al resultado. ■

Relaciones de equivalencia

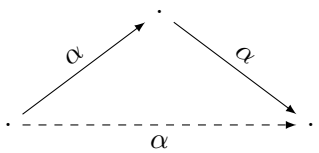
Llamamos a una relación *de equivalencia* cuando cumple las propiedades usuales de reflexividad, transitividad y simetría. Notar que $\leftrightarrow^*$ es siempre una relación de equivalencia (y de hecho, es la menor que contiene a $\to$).

Dada una relación de equivalencia $\approx$, decimos que $\to$ es *compatible* con $\approx$ si siempre que $a' \approx a \to b \approx b'$, también tenemos $a' \to b'$. Esto puede expresarse como $\approx \cdot \to \cdot \approx \quad \subseteq \quad \to$. Con esa idea, definimos $\to \,/\, \approx$ ($\to$ *módulo* $\approx$) cómo $\approx \cdot \to \cdot \approx$. Puede demostrarse que es la menor relación compatible con $\approx$ que contiene a $\to$. No nos adentraremos mucho en el área de "reescritura módulo". El lector interesado puede referirse a (Sethi 1974) o (TeReSe 2003, Capítulo 14).

Diagramas

A veces, la definición de una propiedad, o incluso una demostración, es más clara al usar un *diagrama*: una representación gráfica de las reducciones relevantes.

Por ejemplo, para expresar que la relación α es transitiva, podemos dar el siguiente diagrama, donde el estilo de flecha tiene un significado preciso:



Las flechas con líneas enteras están cuantificadas universalmente y las punteadas lo están existencialmente. Un diagrama es válido si para toda instanciación de las flechas enteras, las flechas punteadas existen.

Formas normales

Además de las propiedades usuales sobre relaciones, en el estudio de los ARS nos interesarán otras propiedades más relacionadas a la noción de ejecución. Serán de interés los elementos que no puedan reducir, y les daremos un nombre.

Definición 1.6. Un elemento b tal que no existe b' con $b \to b'$ se llama *forma normal*. Si $a \to^* b$ con b forma normal se dice que b es *una forma normal de* a.

Podemos decir a *es* $\to$*-normal* cuando queramos hacer énfasis en la relación. Los elementos que tengan formas normales también serán de interés.

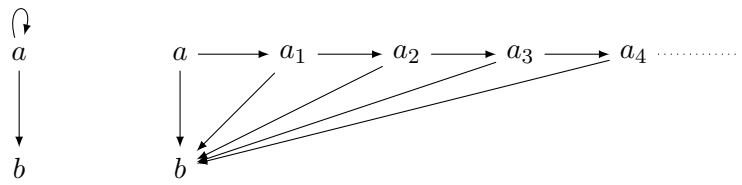
Definición 1.7. Si a tiene una forma normal, lo llamaremos *normalizable* o *débilmente normalizante* (notado $\mathrm{WN}(a)$).

La normalización débil indica que un elemento puede (pero no necesariamente debe) normalizar. Más aún, hay elementos que siempre normalizan.

Definición 1.8. Si no existe una cadena de reducción infinita $a \to a_1 \to a_2 \cdots$, diremos que a es *fuertemente normalizante* (notado $\mathrm{SN}(a)$).

Claramente, un elemento fuertemente normalizante es normalizable, pero la inversa no vale, como lo muestran los siguientes dos ejemplos.

Ejemplo 1.9. Ejemplos de $\mathrm{WN}(a) \not\Longrightarrow \mathrm{SN}(a)$.



Tenemos $\mathrm{WN}(a)$ en ambos, pero existen las cadenas infinitas $a \to a \to a \to \cdots$ y $a \to a_1 \to a_2 \to \cdots$ respectivamente, y por lo tanto $\neg\mathrm{SN}(a)$.

Como vimos, una reducción $\to$ puede ser no determinista, es decir, dado un a podemos tener $a \to b$ y $a \to c$ con $b \neq c$. En ese caso, decimos que

tenemos una *elección* de cómo reducir a a. Una propiedad deseable es que distintas elecciones no influyan en el resultado final.

Definición 1.10. Si para todas formas normales b_1, b_2 de a ocurre que $b_1 = b_2$, diremos que a tiene *formas normales únicas* (notado $\mathrm{UN}(a)$).

En los diagramas del Ejemplo 1.9, tenemos $\mathrm{UN}(a)$, dado que b es su única forma normal.

Llamamos *estrategia de reducción* a la elección que hacemos al reducir los elementos de un ARS. Es decir es una función que elige, para un elemento no normal a, un único sucesor de a. La propiedad UN, entonces, implica que la estrategia no influye en el resultado final.

Para cada propiedad $P \in \{\mathrm{WN}, \mathrm{SN}, \mathrm{UN}\}$, diremos que un ARS $\mathcal{A}$ *tiene la propiedad* P cuando para cada elemento a, tengamos $P(a)$. En ese caso, lo notamos con $\mathcal{A} \models P$, o con $\to\!\models P$ si queremos hacer énfasis en la relación. Usaremos la misma notación para propiedades futuras.

De estas tres propiedades, nos interesará en particular la unicidad de formas normales. En el contexto de lenguajes de programación, indica que el programa puede evaluar a un único valor (o divergir), lo cual es una propiedad de suma importancia al asegurar que el resultado de todo programa (si existe) está bien definido.

Inducción bien fundada

Una consecuencia importante de la normalización fuerte es que vale el principio de *inducción bien fundada* (WFI). Para demostrar una propiedad P sobre todo un sistema, basta con demostrarla para cada elemento asumiendo que es cierta para todos sus sucesores. La ausencia de cadenas infinitas implica la consistencia de este argumento.

El principio puede expresarse con la siguiente regla, válida siempre que $\mathcal{A} \models \mathrm{SN}$ (si bien simple, omitimos la demostración).

$$\frac{\forall a.\ (\forall b.\ a \to^+ b \implies P(b)) \implies P(a)}{\forall a.\ P(a)}\ \mathrm{WFI}$$

Ejemplo 1.11. Si se considera el ARS $\mathbb{N}_r$ de la forma

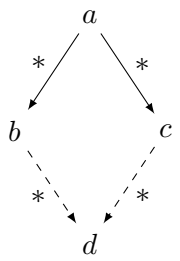
$$0 \leftarrow 1 \leftarrow 2 \leftarrow 3 \leftarrow 4 \leftarrow 5 \cdots\cdots$$

La inducción bien fundada coincide con la inducción usual ("fuerte" o "completa") en los naturales.

1.2. Confluencia

Esta tesina se enfoca en la propiedad de *confluencia*. En esta sección la definimos y mostramos algunas consecuencias útiles de dicha propiedad. Luego mostramos criterios abstractos (es decir, independientes de un ARS en particular) para demostrarla. También damos una prueba concreta de confluencia en el λ-cálculo.

La propiedad de confluencia es simple de presentar como diagrama:



O, formalmente:

Definición 1.12. Un elemento a es *confluente* (notado $\mathrm{CR}(a)$[1]) si para todo b, c tales que $a \to^* b$ y $a \to^* c$, existe un d tal que $b \to^* d$ y $c \to^* d$.

Intuitivamente, toda "divergencia" desde a puede "converger". Notar que ambos sistemas del Ejemplo 1.9 son confluentes. Además, notar que la confluencia puede verse como una propiedad de la clausura de una relación.

Como discutiremos mucho sobre confluencia en este trabajo, vale la pena dar alguna terminología específica. Al subdiagrama formado por a, b y c lo llamaremos la *parte superior*. Análogamente, el subdiagrama con b, c y d será la *parte inferior*. Entonces, diremos que una parte superior puede *cerrarse* cuando exista una parte inferior para la misma. En el diagrama de ejemplo, a es la raíz. A una parte superior con raíz a lo llamaremos un *a-diagrama*. Un elemento a es entonces confluente cuando todo a-diagrama puede cerrarse.

Una de las razones más importantes por las cuales nos interesa la confluencia es porque implica, fácilmente, la unicidad de formas normales.

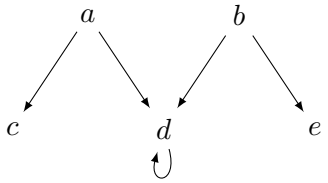
Lema 1.13. $\mathcal{A} \models \mathrm{CR} \implies \mathcal{A} \models \mathrm{UN}$

Demostración. Sean b, c dos formas normales de un elemento a. Dado que $\mathrm{CR}(a)$, existe un d tal que $b \to^* d \leftarrow^* c$. Sin embargo, b y c son formas normales y no pueden reducir vía $\to$, por lo que concluimos $b = d = c$. ■

[1] Las siglas son en honor a Alonzo Church y John Barkley Rosser, quienes estudiaron y demostraron la propiedad para el λ-cálculo en (Church y Rosser 1936).

Demostrar confluencia suele ser, además, más fácil que demostrar UN directamente. Sin embargo, CR es una propiedad estrictamente más fuerte que UN.

Ejemplo 1.14 (UN $\not\Longrightarrow$ CR)**.** En el siguiente ARS:



ocurre que todo elemento es UN, pero claramente a y b no son confluentes.

Entonces, además de dar la unicidad de formas normales, la confluencia implica que toda divergencia es salvable incluso si nunca se llega a una forma normal.

Teoría ecuacional y consistencia

Todo ARS induce una noción ecuacional, donde la equivalencia se define como $\leftrightarrow^*$: la menor relación de equivalencia que contiene a $\rightarrow$. Intuitivamente, a es equivalente a b cuando hay un camino no dirigido entre ellos.

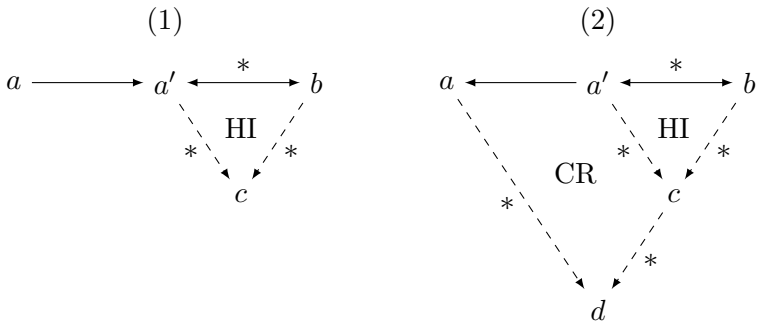
Definición 1.15. Decimos que a es *equivalente* a b cuando $a \leftrightarrow^* b$.

En un sistema confluente, la equivalencia puede decidirse por la existencia de un reducto común. Entonces, si la reducción es fuertemente normalizante, la equivalencia es decidible.

Definición 1.16. Un ARS $\mathcal{A}$ tiene la propiedad *Church-Rosser* cuando todo par de elementos equivalentes tienen un reducto común. Es decir, $a \leftrightarrow^* b$ implica que existe c tal que $a \rightarrow^* c \leftarrow^* b$.

Lema 1.17. *Un ARS $\mathcal{A}$ es confluente si y sólo si es Church-Rosser.*

Demostración. La vuelta es trivial. Demostramos la ida por inducción en la cantidad de pasos en $a \leftrightarrow^* b$. El caso $n = 0$ es trivial ya que $a = b$. Asumimos $a \leftrightarrow a' \leftrightarrow^* b$. Por la HI, existe c tal que $a' \rightarrow^* c \leftarrow^* b$. Procedemos por casos, representados en los siguientes diagramas



(1) Si $a \to a'$, c es un reducto común de a y b.
(2) Si $a \leftarrow a'$, usamos la confluencia de a' para encontrar d tal que $a \to^* d \leftarrow^* c$. Entonces, d es un reducto común de a y b. ■

Una propiedad deseable de toda teoría ecuacional es que sea *consistente*. Es decir, que no todos los elementos sean equivalentes, lo cual trivializa la teoría.

La propiedad Church-Rosser provee entonces una forma muy directa para demostrar la consistencia de una teoría (provisto que existen al menos dos formas normales distintas).

Lema 1.18. *Si* $\mathcal{A} \models \mathrm{CR}$ *y* b_1, b_2 *son formas normales, tenemos* $b_1 \leftrightarrow^* b_2 \iff b_1 = b_2$.

Demostración. La vuelta es trivial. Para la ida, como $b_1 \leftrightarrow^* b_2$, deben tener un reducto común (por la propiedad Church-Rosser). Sin embargo, al ser normales no tienen ningún reducto propio, y debe ser que $b_1 = b_2$. ■

Notar que en el Ejemplo 1.14, tenemos c y e equivalentes, siendo formas normales distintas. La confluencia garantiza que estas situaciones indeseables no pueden ocurrir.

Entonces, ciertamente la confluencia es una propiedad deseable, y queremos demostrarla para nuestros sistemas.

Demostrando confluencia

En general, tratar de cerrar la parte superior de un diagrama no es trivial, principalmente porque las divergencias pueden darse en cualquier cantidad de pasos. Por eso buscamos criterios que simplifiquen la tarea.

Debe notarse que, ya que la confluencia es una propiedad de la clausura, podemos razonar sobre cualquier otra relación con la misma clausura. Entonces, el Lema 1.4 provee un criterio para cambiar de la relación:

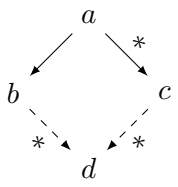
Lema 1.19. *Si* $\to_R \subseteq \to_S^*$ *y* $\to_S \subseteq \to_R^*$, *entonces* $\to_S \models \mathrm{CR} \iff \to_R \models \mathrm{CR}$.

Demostración. Por el Lema 1.4, las clausuras $\to_R^*$ y $\to_S^*$ coinciden. Al ser la confluencia una propiedad de la clausura, ambas propiedades son equivalentes. ■

Semi-confluencia

Una primera simplificación a los diagramas es restringir una de las reducciones de la parte superior a exactamente un paso.

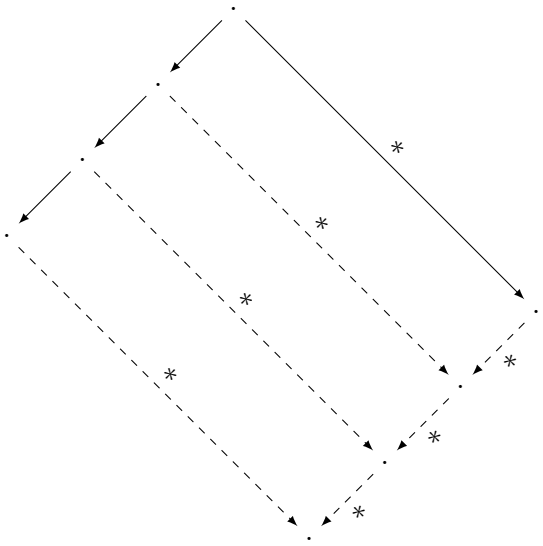
Definición 1.20. Un elemento a es *semi-confluente* (notado SCR(a)) cuando $b \leftarrow a \to^* c$ implica que existe d tal que $b \to^* d \leftarrow^* c$. Gráficamente:



La semi-confluencia de un sistema implica confluencia.

Lema 1.21. $\mathcal{A} \models \text{SCR} \implies \mathcal{A} \models \text{CR}$.

Demostración. La prueba es fácil de expresar con un diagrama:

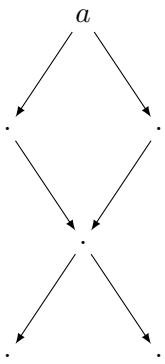


■

Además, $\mathcal{A} \models \text{CR}$ implica $\mathcal{A} \models \text{SCR}$ (trivialmente), con lo cual es una propiedad equivalente sobre un sistema. Sin embargo, pueden no ser equivalentes

elemento a elemento. En el siguiente ejemplo, a es SCR, pero claramente no es CR.

Ejemplo 1.22 ($\mathrm{SCR}(a) \not\Rightarrow \mathrm{CR}(a)$)**.**

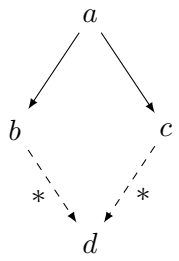


De todas maneras, por lo general nos interesa que la propiedad se cumpla para todo el sistema, y no para un elemento individual. Sin embargo, puede ser contraintuitivo.

Confluencia local

Yendo más allá, podríamos pensar en restringir ambas reducciones de la parte superior a un sólo paso.

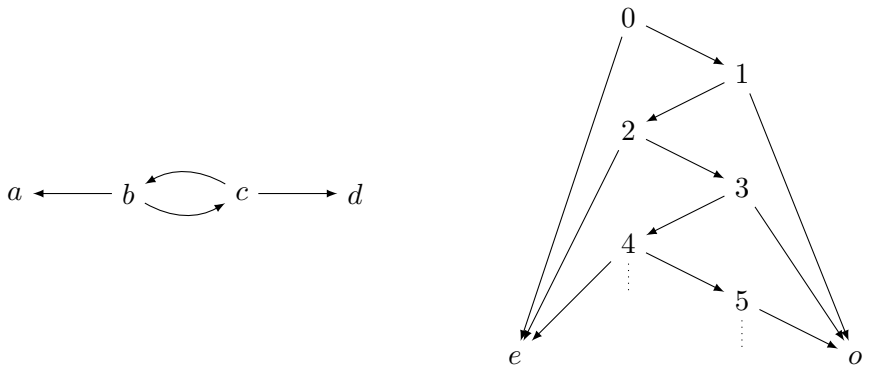
Definición 1.23. Un elemento a es *localmente confluente* (notado $\mathrm{LC}(a)$) cuando $b \leftarrow a \rightarrow c$ implica que existe d tal que $b \rightarrow^* d \leftarrow^* c$. Gráficamente:



Habiendo visto que $\mathcal{A} \models \mathrm{SCR} \iff \mathcal{A} \models \mathrm{CR}$, se podría esperar que lo mismo ocurra con LC. Sin embargo, esto falla, como se ve en los siguientes dos ejemplos clásicos[2].

[2]Curiosamente, como es notado por Huet, ambos sistemas pueden verse cómo proyecciones distintas de un mismo objeto tridimensional.

Ejemplo 1.24 (LC $\not\Rightarrow$ CR)**.**

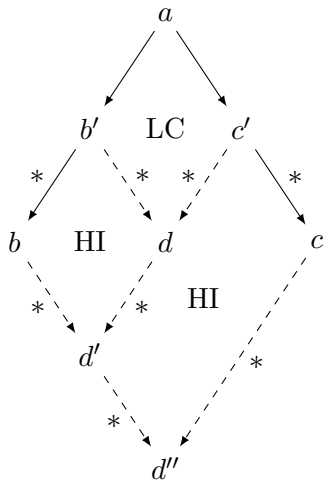


Entonces, en general LC no implica CR. Pero, ambos contraejemplos dados contienen reducciones infinitas (tanto con ciclos como sin). En (Newman 1942) se demuestra que en la ausencia de cadenas infinitas la implicancia sí vale, es decir:

Lema 1.25 (Lema de Newman)**.** *Si* $\mathcal{A} \models$ SN *y* $\mathcal{A} \models$ LC*, entonces* $\mathcal{A} \models$ CR*.*

Reproducimos la demostración de (Huet 1980), quien usa inducción bien fundada para obtener el resultado con facilidad.

Demostración. Por inducción bien fundada sobre $\rightarrow$. Supongamos que $b \leftarrow^* a \rightarrow^* c$. Debemos cerrar b y c. Si alguna de las reducciones es de cero pasos, es trivialmente cerrable. Si no, estamos en el caso del diagrama siguiente.



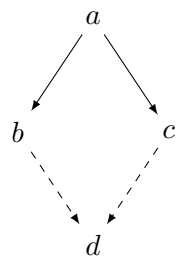
Usando la hipótesis de confluencia local encontramos d. Por las hipótesis inductivas para c' y b', son confluentes y podemos encontrar d' y d''. ■

Propiedad diamante

Vimos que, en general, la confluencia local no implica confluencia. Al intentar demostrar la implicancia uno observa que no se puede hacer una inducción correcta en la cantidad de pasos. En efecto, para cerrar una divergencia de un sólo paso pueden hacer falta arbitrariamente muchos.

Pero, ¿qué sucede si no variamos la cantidad de pasos? Esto da lugar a una nueva propiedad, estrictamente más fuerte que todas las otras, llamada *propiedad diamante*[3].

Definición 1.26. Un elemento a cumple la *propiedad diamante* (notado $\Diamond(a)$) cuando $b \leftarrow a \rightarrow c$ implica que existe d tal que $b \rightarrow d \leftarrow c$. Gráficamente:



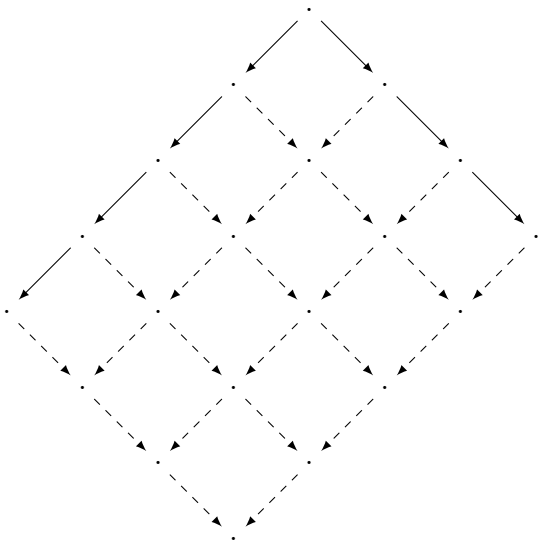
Notar que $\rightarrow \models \mathrm{CR} \iff \rightarrow^* \models \Diamond$.

Es claro que $\mathcal{A} \models \mathrm{CR} \not\Longrightarrow \mathcal{A} \models \Diamond$ (los sistemas del Ejemplo 1.9 son testigo). Pero podemos demostrar la inversa de manera simple.

Lema 1.27. $\rightarrow \models \Diamond \implies \rightarrow \models \mathrm{CR}$.

Demostración. La prueba está expresada por el siguiente diagrama:

[3] La propiedad lleva ese nombre no sólo por la forma del diagrama, sino también porque es preciosa y muy difícil de conseguir.



■

La propiedad diamante tiene más consecuencias útiles aparte de la confluencia, pero no nos adentraremos en ellas ya que escapan al objetivo de esta tesina.

Diamante débil

Si bien útiles, los diamantes son difíciles de encontrar en la naturaleza debido a la restricción de cerrar en exactamente un paso. Eso parece muy fuerte: ¿qué ocurre si cerramos en a lo sumo un paso? Intuitivamente, demostrar confluencia por inducción debería funcionar en este caso.

Demostraremos que si podemos cerrar los diagramas en 0 o 1 pasos, también tenemos confluencia. De hecho, implica la propiedad diamante sobre la clausura reflexiva.

> **Lema 1.28.** *Si siempre que* $b \leftarrow a \rightarrow c$ *existe* d *tal que* $b \rightarrow^{=} d \leftarrow^{=} c$, *entonces* $\rightarrow^{=} \models \Diamond$.

Demostración. Debemos cerrar $b \leftarrow^{=} a \rightarrow^{=} c$ en exactamente un paso de $\rightarrow^{=}$. Si ambos pasos son nulos, tenemos $b = a = c$, y reducimos $b \rightarrow^{0} a \leftarrow^{0} d$. Si $a \rightarrow^{0} b$, tomamos $d = c$ con $b \rightarrow d \leftarrow^{0} c$. Si $a \rightarrow^{0} c$, tomamos $d = b$ con $b \rightarrow^{0} d \leftarrow c$. En otro caso, tenemos por la hipótesis que existe d tal que $b \rightarrow^{=} d \leftarrow^{=} c$, como se necesita. ■

Entonces, usando el Lema 1.27 y notando que $(\rightarrow^{=})^{*} = \rightarrow^{*}$, tenemos que

$$\rightarrow^{=} \models \Diamond \implies \rightarrow^{=} \models \mathrm{CR} \implies \rightarrow \models \mathrm{CR}$$

Hemos introducido a los sistemas de reescritura abstracta junto a sus propiedades usuales. Le prestamos especial atención a la confluencia (por razones obvias), y dimos varios criterios simplificados para demostrarla. Con este preámbulo, estamos listos para analizar un ARS concreto.

1.3. λ-cálculo y β-reducción

Un uso estándar de los sistemas de reescritura es modelar la ejecución de los términos de un lenguaje de programación. Ciertamente, el ejemplo más común es el λ-cálculo, donde la relación de reescritura es la *β-reducción*.

El conjunto Λ de λ-términos está dado por la gramática

$$M, N ::= x \mid \lambda x.M \mid MN$$

donde $x, y, \ldots$ representan variables tomadas de un conjunto infinito numerable. Identificaremos a todos los términos α-equivalentes, y no nos preocuparemos por las capturas de variables.

Entonces, pensaremos en un ARS $\Lambda_r = (\Lambda, \longrightarrow_\beta)$ donde $\longrightarrow_\beta$ representa a la β-reducción, la cual tiene la siguiente definición inductiva (la sustitución tiene su definición usual).

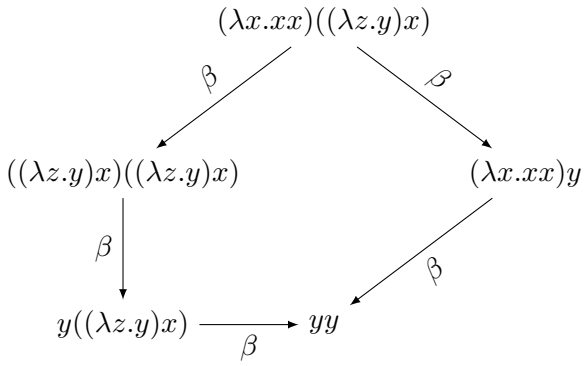
$$\frac{}{(\lambda x.M)N \longrightarrow_\beta M[N/x]}\ \text{R-}\beta \qquad \frac{M \longrightarrow_\beta M'}{\lambda x.M \longrightarrow_\beta \lambda x.M'}\ \text{R-}\lambda$$

$$\frac{M \longrightarrow_\beta M'}{MN \longrightarrow_\beta M'N}\ \text{R-AppL} \qquad \frac{N \longrightarrow_\beta N'}{MN \longrightarrow_\beta MN'}\ \text{R-AppR}$$

Las reglas R-λ, R-AppL y R-AppR, que permiten reducir arbitrariamente cualquier subtérmino, son llamadas *reglas de congruencia* (o simplemente *congruencias*).

Una expresión de la forma $(\lambda x.M)N$ se denomina *β-redex*. Cuando un término reduce, es por la *contracción* de exactamente un β-redex presente en el mismo. Entonces, la estrategia de reducción es equivalente a elegir un β-redex para cada término (que no esté en forma normal).

Ejemplo 1.29. En Λ_r tenemos las siguientes reducciones



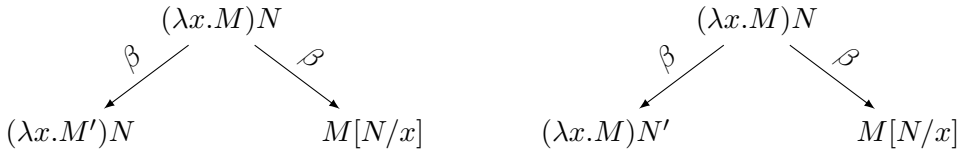
Es sabido que el λ-cálculo no es ni fuertemente ni débilmente normalizante ($\Omega = (\lambda x.xx)(\lambda x.xx)$ es el ejemplo clásico). Por el lado positivo, la β-reducción es confluente. Veremos cómo demostrar esto.

Demostrando confluencia para la β-reducción

Analizar el comportamiento de una secuencia de pasos es difícil (aunque no imposible), ya que los términos pueden variar de gran manera. Intentemos entonces restringirnos a un único paso en la parte superior.

Primero, notamos que demostrar confluencia local no sería útil, ya que existen cadenas infinitas.

Una posibilidad es intentar demostrar la propiedad diamante para $\longrightarrow_\beta$. La mayoría de los casos son triviales aplicando congruencias y/o la hipótesis inductiva. Los únicos dos casos interesantes (llamados *pares críticos*) son las interacciones de las congruencias con la β-reducción.

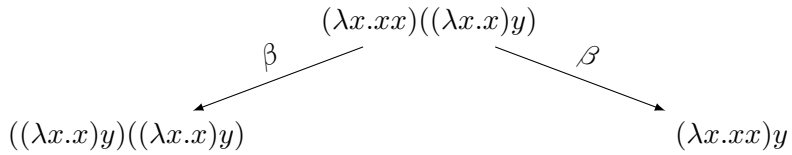


Podemos cerrar el primero, en exactamente un paso, en el término $M'[N/x]$. Para eso necesitaremos el siguiente lema.

Lema 1.30. *Si $M \longrightarrow_\beta M'$, entonces $M[N/x] \longrightarrow_\beta M'[N/x]$.*

Demostración. Por inducción en $M \longrightarrow_\beta M'$. ■

Sin embargo no podemos, en general, cerrar el otro diagrama. Consideremos la divergencia:



La única posibilidad para cerrar es confluir a yy en un paso. Podemos hacer esto a la derecha, pero no a la izquierda. Necesitamos exactamente 2 pasos, para reducir cada copia de $(\lambda x.x)y$. Análogamente, podemos formar términos que copien su argumento n veces, y necesitaremos n pasos.

Entonces, las copias de un elemento impiden que consigamos nuestro diamante y $\longrightarrow_\beta \not\models \Diamond$. Aun así, podremos demostrar confluencia cambiando de relación.

Reducción paralela

¿Qué pasa si usamos una relación de reescritura que puede reescribir todas las copias a la vez? Ciertamente no pensamos en $\longrightarrow_\beta^*$ (que no simplificaría el problema), sino en alguna noción de reducción *paralela* que pueda actuar sobre varios subárboles de un término a la vez.

Para esto, seguiremos la demostración en (Barendregt 1981) (atribuida a Tait y Martin-Löf) pero omitiendo las pruebas sintácticas, y dando sólo la idea de alto nivel. El lector puede referirse a dicho libro para un desarrollo completo.

Una forma de definir dicha reescritura paralela es la siguiente:

$$\frac{M \twoheadrightarrow_\beta M' \quad N \twoheadrightarrow_\beta N'}{(\lambda x.M)N \twoheadrightarrow_\beta M'[N'/x]}\ \textsc{Rp-}\beta \qquad \frac{M \twoheadrightarrow_\beta M'}{\lambda x.M \twoheadrightarrow_\beta \lambda x.M'}\ \textsc{Rp-}\lambda$$

$$\frac{M \twoheadrightarrow_\beta M' \quad N \twoheadrightarrow_\beta N'}{MN \longrightarrow_\beta M'N'}\ \textsc{Rp-App} \qquad \frac{}{M \twoheadrightarrow_\beta M}\ \textsc{Rp-Refl}$$

Vemos el carácter paralelo de la reducción en las reglas Rp-β y Rp-App. Con esta relación conseguimos un lema análogo al anterior, pero que permite variar también el término siendo sustituido.

Lema 1.31 (Sustitución)**.** *Si* $M \twoheadrightarrow_\beta M'$ *y* $N \twoheadrightarrow_\beta N'$*, entonces* $M[N/x] \twoheadrightarrow_\beta M'[N'/x]$.

Demostración. Por inducción en $M \twoheadrightarrow_\beta M'$. ■

Armados con ese lema podemos demostrar que $\twoheadrightarrow_\beta$ cumple la propiedad diamante.

Lema 1.32. $\twoheadrightarrow_\beta \models \Diamond$

Demostración. Por inducción en la forma de las reducciones. ■

Además, ya que $\twoheadrightarrow_\beta$ es reflexiva, siempre podemos no reducir un subtérmino. Entonces, puede mostrarse que $\longrightarrow_\beta \subseteq \twoheadrightarrow_\beta$. Una simple prueba inductiva muestra que podemos simular $\twoheadrightarrow_\beta$ con la relación original $\longrightarrow_\beta$.

Lema 1.33. $\twoheadrightarrow_\beta \subseteq \longrightarrow_\beta^*$

Demostración. Por inducción en la forma de la reducción. ■

Y entonces, $\twoheadrightarrow_\beta^* = \longrightarrow_\beta^*$. Finalmente concluimos que la β-reducción es confluente por el siguiente argumento:

$$\twoheadrightarrow_\beta \models \Diamond \implies \twoheadrightarrow_\beta \models \mathrm{CR} \implies \longrightarrow_\beta \models \mathrm{CR}$$

De alguna manera, el cambio de relación fue para buscar una forma de hacer inducción apropiada sobre los pasos de $\longrightarrow_\beta$. Existen maneras más directas, pero son considerablemente más complejas. Otras opciones, en general, son descomponer la relación de alguna forma ($\to = \to_1 \cup \to_2$, $\to = \to_1 \cdot \to_2$, ...) en subrelaciones confluentes e intentar componer sus confluencias (lo cual no siempre es trivial).

1.4. Conmutación secuencial

Una propiedad no tan usual de las relaciones de reescritura, pero que será muy usada en los capítulos siguientes, es la *conmutación secuencial*.

Definición 1.34. Decimos que una relación α *conmuta secuencialmente* con β (o que conmuta *sobre* β), y lo notamos $\alpha \dashv \beta$, si

$$\beta \cdot \alpha \subseteq \alpha \cdot \beta$$

Intuitivamente, esto implica que las reducciones de α pueden moverse sobre las de β, y realizarse primero. Notar que esta propiedad no es simétrica.

Es importante no confundir esta propiedad con la conmutación usual (o paralela, que podría notarse $\alpha \parallel \beta$) de α y β, que es equivalente a $\alpha^{-1} \cdot \beta \subseteq \alpha \cdot \beta^{-1}$ (y es una propiedad simétrica)[4]. La diferencia se puede ver gráficamente en la Figura 1.1, donde usamos nuestra convención notacional usual.

La conmutación secuencial se mantiene al aumentar las cantidades de pasos.

[4]Paradójicamente, la conmutación paralela es muy usada en estudios de confluencia, pero no le daremos uso.

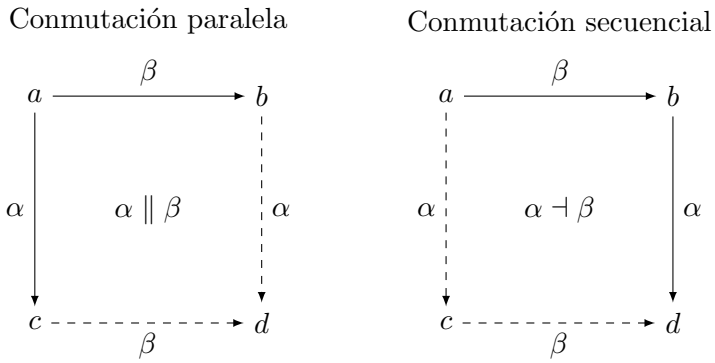
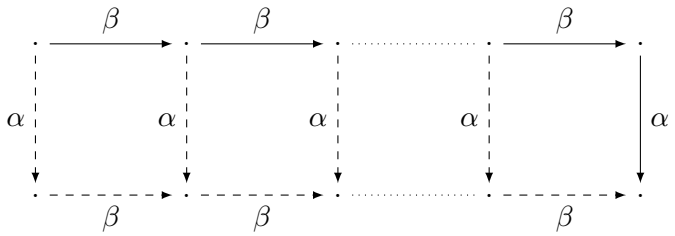


Figura 1.1: Diferencia entre conmutación paralela y conmutación secuencial

Lema 1.35. *Si $\alpha \dashv \beta$, entonces $\alpha \dashv \beta^m$ y $\alpha^n \dashv \beta$.*

Demostración. Para el primer resultado, damos una prueba gráfica en el siguiente diagrama, donde cada cuadrado está justificado por la hipótesis.



Para el segundo, la prueba es análoga. (O por inducción en m y n). ■

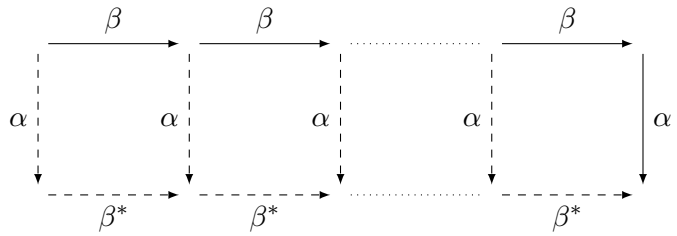
Lema 1.36. *Si $\alpha \dashv \beta$, entonces $\alpha^* \dashv \beta$ y $\alpha \dashv \beta^*$.*

Demostración. Análoga al Lema 1.35. ■

Cuando queramos demostrar una conmutación con una clausura, podemos dar un criterio (equivalente) simplificado.

Lema 1.37. *Si $\beta \cdot \alpha \subseteq \alpha \cdot \beta^*$, entonces $\alpha \dashv \beta^*$.*

Demostración. Damos una prueba gráfica en el siguiente diagrama, donde cada cuadrado está justificado por la hipótesis, y usamos la transitividad de β^*:



(O por inducción en la cantidad de pasos) ■

Análogamente tenemos un lema dual:

Lema 1.38. *Si* $\beta \cdot \alpha \subseteq \alpha^* \cdot \beta$*, entonces* $\alpha^* \dashv \beta$.

Demostración. Análoga al Lema 1.37. ■

Cuando las clausuras de dos relaciones conmutan, la clausura de su unión puede descomponerse fácilmente:

Lema 1.39. *Si* $\alpha^* \dashv \beta^*$*, entonces* $(\alpha \cup \beta)^* = \alpha^* \cdot \beta^*$.

Demostración. Por doble contención. La dirección $\supseteq$ es obvia. Para la otra dirección, tenemos que $a \to^*_{\alpha\cup\beta} c$. Procedemos por inducción en la cantidad de pasos. Si la reducción es vacía, el resultado es trivial. Si no, existe un b tal que $a \to_{\alpha\cup\beta} b \to^*_{\alpha\cup\beta} c$. Por la hipótesis inductiva sabemos que existe un b' tal que $b \to^*_\alpha b' \to^*_\beta c$. Hay dos casos para la primera reducción:

- $a \to_\alpha b$
  Componiendo con la hipótesis inductiva es claro que $a \to^*_\alpha b' \to^*_\beta c$.

- $a \to_\beta b$
  Tenemos $a \to_\beta b \to^*_\alpha b'$. Por la conmutación de α^* sobre β^*, existe un b'' tal que $a \to^*_\alpha b'' \to^*_\beta b'$. Componiendo con $b' \to^*_\beta c$ llegamos a nuestra meta. ■

Este último lema será de utilidad, ya que indica una forma de descomponer una relación cuando encontremos una conmutación "interna".

Capítulo 2

La necesidad de probabilidad

"Todo lo que pueda ir mal, irá mal."

—Ley de Murphy

Existen sistemas en donde la noción de reducción tiene un comportamiento probabilista, y no puede saberse con certeza el resultado de las mismas. Los ARS tradicionales no pueden expresar a estos sistemas. En este capítulo se describen los *sistemas de reescritura probabilista (PARS)* que sí pueden.

Motivados por nuestro interés en los lenguajes de programación, analizamos la expresividad de esta noción. Descubrimos que no pueden expresar congruencias, ni ninguna noción de elección no determinista.

Finalmente, mostramos un cálculo cuántico existente que sí permite elecciones no deterministas, y no puede ser modelado como un PARS. Concluimos que hace falta una noción más rica de reescritura probabilista, la cual será provista en el siguiente capítulo.

2.1. Sistemas de reescritura probabilista — PARS

En la naturaleza existen procesos probabilistas, los cuales queremos modelar y estudiar. Ejemplos son el tirar una moneda o un dado, las caminatas al azar y los lenguajes de programación con primitivas aleatorias. Un ejemplo de particular interés es la computación cuántica, en donde el proceso de medición es inherentemente probabilista.

En dichos procesos, un elemento puede tener distintas formas de reducir las cuales no controlamos (comúnmente, esto se llama *elección demoníaca*). Además, cada posible resultado tiene una probabilidad asociada, convirtiendo a la reducción en una noción cuantitativa.

Modelaremos estos comportamientos como sistemas de reescritura. En un ARS, la reducción es cualitativa (un elemento a o bien reduce a b o bien no lo hace) y consideramos poder elegir cuál reducción ocurre (*elección angelical*). Entonces, esta noción de sistema de reescritura no es adecuada, y se necesita una nueva.

Para conseguirla, será necesario cambiar la representación de la relación de reducción. Claramente, el tipo $\mathcal{P}(A \times A)$ no da a lugar a ninguna noción cuantitativa.

Notamos que una relación de ese tipo es isomorfa a una función de tipo $A \to A \to \{0,1\}$, es decir, un predicado que para cada a y b indique si a reduce a b o no. Para generalizar esta noción, podemos variar el codominio: en vez de tener sólo dos opciones, deberíamos poder permitir un continuo de "grados" de reducción.

Entonces podemos modelar nuestra reducción con una función de tipo $A \to A \to [0,1]$, donde $[0,1]$ representa al intervalo cerrado de todos los reales entre 0 y 1, inclusive.

Pero no toda función de ese tipo tiene significado probabilista. Dado un elemento a, debe o bien no reducir o bien sus sucesores tener probabilidades que sumen exactamente 1. En definitiva, cada elemento a tiene una *distribución sucesor* asociada (posiblemente nula), y podemos tomar una función de tipo $A \to \mathscr{D}_{0,1}(A)$.

Implícito en el hecho de usar listas es que cada distribución tiene un soporte finito. Es decir, si pensamos a d como una distribución matemática, el conjunto $\{a \in A \mid d(a) > 0\}$ es finito. No pensamos que esto sea una limitación importante para modelar lenguajes de programación, y lo mismo es cierto en otros trabajos (Selinger y Valiron 2005; Di Pierro, Hankin y Wiklicky 2005).

Una definición de sistema de reescritura probabilista que sigue este espíritu se encuentra en (Bournez y Kirchner 2002). La exponemos aquí, adaptada a la representación de listas.

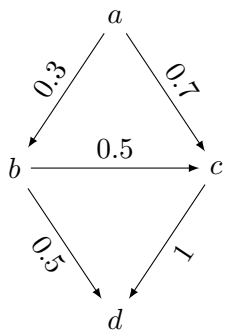
Definición 2.1. Un PARS (*probabilistic abstract rewriting system*) $\mathcal{A}$ es un par $(A, \to)$ donde A es un conjunto llamado *portador*, y $\to$ es una función de tipo $A \to \mathscr{D}_{0,1}(A)$, llamada *función de reducción probabilista*.

Llamaremos *terminal* a un elemento a con distribución sucesor nula. Asumiremos que no hay elementos repetidos en las distribuciones sucesor. Cuando (p, b) aparezca en $\to(a)$, decimos que a *reescribe a* b *con probabilidad* p y lo notamos $a \to_p b$.

Ejemplo 2.2 (Ejemplo de PARS)**.** Sea $\mathcal{A}$ el PARS dado por

$$a \to_{0.3} b \qquad a \to_{0.7} c \qquad c \to_1 d$$
$$b \to_{0.5} c \qquad b \to_{0.5} d$$

El elemento d es el único terminal. Gráficamente, podemos dar la siguiente representación.



Considerando secuencias de muchos pasos, decimos que *a reescribe a b en* $n \geq 0$ *pasos con probabilidad* p (notado $a \to_p^n b$) cuando exista una secuencia de reducciones válidas

$$a \to_{p_1} t_1 \to_{p_2} t_2 \to_{p_3} \cdots \to_{p_n} b$$

con $p = \Pi p_i$. Como se espera, notamos $a \to_p^* b$ cuando exista un n tal que $a \to_p^n b$. Cuando sólo queramos indicar que existe una probabilidad p mayor a cero tal que $a \to_p b$, notamos $a \to b$, y análogamente para $a \to^n b$ y $a \to^* b$.

Ejemplo 2.3 (Monedas)**.** Podemos formar un PARS que modele lanzar monedas. Nuestro conjunto portador son los términos generados por L en la siguiente gramática.

$$\begin{array}{rcl} L & ::= & C; L \mid \epsilon \\ C & ::= & \bigstar \mid v \\ v & ::= & h \mid t \end{array}$$

donde h y t representan cara y cruz (los llamamos *valores*), ϵ es la cadena vacía y $\bigstar$ una moneda lanzada. L representa entonces una secuencia de lances o valores. La reducción irá de izquierda a derecha. Inductivamente, la definimos como:

$$\frac{}{\bigstar; C \to_{1/2} h; C} \qquad \frac{}{\bigstar; C \to_{1/2} t; C} \qquad \frac{C \to_p C'}{v; C \to_p v; C'}$$

Notamos con $\bigstar^n$ a la secuencia de n lances $\bigstar$. Podemos ver entonces que para cualquier n, $\bigstar^n \to_{1/2^n}^n v^n$, para cualquier secuencia de n valores v^n. Además tenemos:

$$\begin{array}{llll} \bigstar;\bigstar \to_{1/2} h;\bigstar & \bigstar;\bigstar \to_{1/4}^2 h;h & \bigstar;\bigstar \to_{1/4}^2 t;h \\ \bigstar;\bigstar \to_{1/2} t;\bigstar & \bigstar;\bigstar \to_{1/4}^2 h;t & \bigstar;\bigstar \to_{1/4}^2 t;t \end{array}$$

Notar que $\to^*$ no forma un PARS, salvo casos triviales. Como los sucesores inmediatos de a ya suman una probabilidad de exactamente 1, no podemos

agregar más sucesores. Esto ocurre con cualquier propiedad, y concluimos que no podemos tomar clausuras de ningún tipo.

En otras palabras, un elemento no terminal de un PARS está *saturado* y no puede extenderse.

Árboles de computación

Una ejecución en un PARS no puede representarse como una secuencia. Debido a la incerteza en la ejecución necesitamos una estructura de árbol que tenga en cuenta los efectos probabilistas (que no controlamos).

Definición 2.4. Dado un elemento a de un PARS, definimos los *árboles de computación con raíz* a (notado $\mathcal{T}(a)$) inductivamente de la siguiente manera:

$$\frac{}{a \in \mathcal{T}(a)} \qquad \frac{\sum_{i=1}^{n} p_i = 1 \quad a \to_{p_i} a_i \quad t_i \in \mathcal{T}(a_i)}{[a, (p_1, t_1), \ldots, (p_n, t_n)] \in \mathcal{T}(a)}$$

Definición 2.5. Llamamos *maximal* a un árbol $T \in \mathcal{T}(a)$ tal que toda hoja es terminal. Puede formalizarse restringiendo a elementos terminales la construcción de hojas.

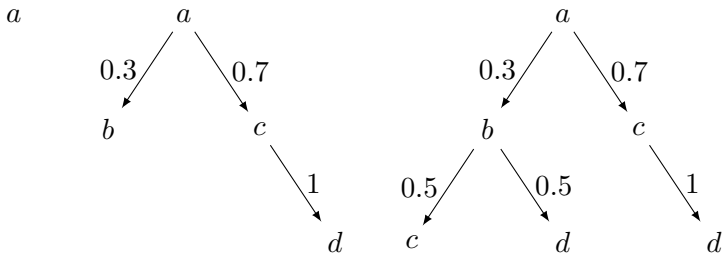
Entonces, un árbol está formado por hojas (de elementos) o por nodos internos de *evoluciones* de un elemento, considerando todas las posibilidades según el PARS.

Notamos con $\mathcal{T}$ a $\bigcup_{a \in A} \mathcal{T}(a)$. En el PARS del Ejemplo 2.2 tenemos los siguientes árboles de computación.

Ejemplo 2.6 (Árboles de computación)**.**

$a \quad [a, (0.3, b), (0.7, [c, (1, d)])] \quad [a, (0.3, [b, (0.5, c), (0.5, d)]), (0.7, [c, (1, d)]]$

Gráficamente, podemos representarlos como:



Todo árbol induce una distribución de probabilidad de peso 1, donde los elementos son las hojas del árbol asociados a las probabilidades asociadas a su rama.

Definición 2.7. Definimos $\mathrm{supp} : \mathcal{T} \to \mathscr{D}_1(A)$ por las siguiente ecuaciones.

$$\begin{array}{rcl} \mathrm{supp}(a) & = & [(1, a)] \\ \mathrm{supp}([a, (p_1, a_1), \ldots, (p_n, a_n)]) & = & p_1\ \mathrm{supp}(a_1) \mathbin{+\!\!+} \cdots \mathbin{+\!\!+} p_n\ \mathrm{supp}(a_n) \end{array}$$

Los árboles del Ejemplo 2.6 tienen los siguientes soportes respectivos.

$$[(1, a)] \qquad [(0.3, b), (0.7, d)] \qquad [(0.15, c), (0.15, d), (0.7, d)]$$

Hay un orden parcial natural entre árboles, similar a la contención. Intuitivamente, nos da una noción de cuando un árbol está "más desarrollado".

Definición 2.8. Definimos el orden parcial $\sqsubseteq$ por las siguientes reglas inductivas

$$\frac{T \in \mathcal{T}(a)}{a \sqsubseteq T} \qquad \frac{\forall i.\ a_i \sqsubseteq a'_i}{[a, (p_1, a_1), \ldots, (p_n, a_n)] \sqsubseteq [a, (p_1, a'_1), \ldots, (p_n, a'_n)]}$$

Cuando $T \sqsubseteq T'$, decimos que T es una *subcomputación* de T', o que T' es una *extensión* de T. Los tres árboles del Ejemplo 2.6 están en orden creciente.

Dado un árbol, podemos extraer la probabilidad asociada a un elemento con la siguiente función, que sólo inspecciona las hojas.

Definición 2.9. Definimos $\mathrm{Prob} : \mathcal{T} \times A \to [0, 1]$ por las siguientes ecuaciones.

$$\begin{array}{rcll} \mathrm{Prob}(b, b) & = & 1 & \\ \mathrm{Prob}(a, b) & = & 0 & \text{con } a \neq b \\ \mathrm{Prob}([a, (p_1, a_1), \ldots, (p_n, a_n)], b) & = & p_1\ \mathrm{Prob}(a_1, b) + \cdots + p_n\ \mathrm{Prob}(a_n, b) & \end{array}$$

En el tercer árbol del Ejemplo 2.6 tenemos $\mathrm{Prob}(T, d) = 0.85$. Notar que Prob no es monótona respecto a $\sqsubseteq$. Llamamos a dos árboles equivalentes cuando asignen la misma probabilidad a cada elemento.

Definición 2.10. Dados $T_1, T_2 \in \mathcal{T}(A)$, decimos que son *equivalentes* si para todo elemento c tenemos $\mathrm{Prob}(T_1, c) = \mathrm{Prob}(T_2, c)$.

2.2. Programas probabilistas

Nuestro interés en la reescritura probabilista es modelar lenguajes de programación con la misma cualidad, de los cuales existen varios ejemplos (Selinger y Valiron 2005; Dal Lago, Masini y Zorzi 2011; Díaz-Caro y Dowek 2016).

En los ARS, vimos que la confluencia es una propiedad muy deseable al implicar que los resultados de los programas están bien definidos, independientemente de la estrategia.

En un lenguaje probabilista, no podemos tomar la semántica de un programa como el valor al que reduce, debido a que puede no ser único, y debemos considerar el conjunto de todos los valores posibles. Además, debemos diferenciar programas que devuelvan los mismos valores con distintas probabilidades.

Entonces, para modelar apropiadamente el efecto probabilista, tomamos la semántica de un programa como la distribución de sus resultados.

Traspasando nuestra intuición sobre confluencia, esperamos que, en un lenguaje correcto, cualquier elección que tomemos al reducir no tenga efecto en esta distribución. Esto es vacuamente cierto para los lenguajes modelados como PARS, dado que no podemos tener la capacidad de expresar una elección a la hora de reducir.

2.3. La falta de elección

Volvamos al Ejemplo 2.3, donde las monedas se lanzan de izquierda a derecha. Intentemos modelar un sistema que permita elegir el orden de lanzamiento.

Supongamos que tenemos dos monedas, y nuestro estado inicial es $\bigstar^2$. Si podemos elegir lanzar la primera, deberíamos tener exactamente el comportamiento anterior.

$$\bigstar^2 \to_{1/2} h;\bigstar \qquad\qquad \bigstar^2 \to_{1/2} t;\bigstar$$

Además deberíamos poder empezar por la segunda moneda, necesitando tener

$$\bigstar^2 \to_{1/2} \bigstar;h \qquad\qquad \bigstar^2 \to_{1/2} \bigstar;t$$

Aquí llegamos a un problema: las probabilidades salientes de $\bigstar^2$ suman 2, y no 1, haciendo que estas reducciones no puedan formar un PARS. El elemento $\bigstar^2$ ya estaba saturado en el sistema original, y no podemos extender sus comportamientos.

El mismo problema ocurre en un lenguaje de programación si queremos tomar congruencias para la aplicación o cualquier otra construcción binaria (o n-aria con $n > 1$).

Una opción para poder entrar en la definición de un PARS es normalizar las probabilidades de reducción para conseguir un total de 1, obteniendo algo

como:

$$\bigstar^2 \to_{1/4} h;\bigstar \qquad \bigstar^2 \to_{1/4} t;\bigstar$$
$$\bigstar^2 \to_{1/4} \bigstar;h \qquad \bigstar^2 \to_{1/4} \bigstar;t$$

Sin embargo, este no es el comportamiento que se quiere modelar: las monedas no se lanzan espontáneamente, sino que nosotros queremos elegir cómo lanzarlas. Además, queremos conservar las reducciones del PARS original de manera exacta, y aquí estamos cambiando su comportamiento cuantitativo.

Resumiendo, esta elección no es consecuencia del efecto probabilista, y es erróneo considerarla como tal.

Por esta limitación, los lenguajes de programación probabilistas (que son modelados como PARS) deben tener un orden de reducción fijo. Entonces, al no haber elección posible, la distribución inducida por un término es bien definida y única. Esto puede formalizarse fácilmente.

Lema 2.11. *Sean $T_1, T_2 \in \mathcal{T}(a)$. Entonces existen T_1', T_2' extensiones equivalentes de T_1, T_2.*

Demostración. Por inducción en los árboles T_1, T_2. Si $T_1 = a$, tomar $T_1' = T_2' = T_2$. Análogamente, si $T_2 = a$, tomar $T_1' = T_2' = T_1$. En otro caso, tenemos $T_1 = [a, (p_1, t_1), \ldots, (p_n, t_n)]$ y $T_2 = [a, (q_1, u_1), \ldots, (q_n, u_n)]$. Por inversión de la formación de árboles, todos los sucesores de a deben estar presentes en t_i y u_i. Asumamos (sin pérdida de generalidad) que están en el mismo orden dando $p_i = q_i$ y que las raíces de cada t_i, u_i son iguales. Por la hipótesis inductiva, existen extensiones t_i', u_i' equivalentes dos a dos. Entonces, tomemos

$$\begin{aligned} T_1' &= [a, (p_1, t_1'), \ldots, (p_n, t_n')] \\ T_2' &= [a, (q_1, u_1'), \ldots, (q_n, u_n')] \end{aligned}$$

extensiones de T_1 y T_2. Tenemos entonces

$$\begin{aligned} \mathrm{Prob}(T_1', c) &= p_1 \ \mathrm{Prob}(t_1', c) + \cdots + p_n \ \mathrm{Prob}(t_n', c) \\ &= q_1 \ \mathrm{Prob}(u_1', c) + \cdots + q_n \ \mathrm{Prob}(u_n', c) = \mathrm{Prob}(T_2', c) \end{aligned}$$

para cada elemento c. ■

Entonces, en particular, si ambos árboles son maximales tenemos que son equivalentes.

Lema 2.12 (Unicidad de distribuciones en PARS)**.** *Si todas las hojas de $T_1, T_2 \in \mathcal{T}(a)$ son terminales, entonces T_1 es equivalente a T_2.*

Demostración. Por el Lema 2.11, existen extensiones equivalentes para T_1, T_2. Como toda hoja es terminal, los árboles no tienen extensiones propias, y son por lo tanto equivalentes. ■

Este resultado no es positivo ya que queremos tener la posibilidad de elegir, y por sobre ello poder demostrar que podemos ser liberales en nuestra elección.

Concluimos que los PARS no son una buena noción para un lenguaje probabilista, y necesitamos una que permita expresar distintas elecciones.

2.4. El cálculo Q^*

El cálculo cuántico Q^* (Dal Lago, Masini y Zorzi 2011) permite distintas estrategias junto al comportamiento probabilista. Debido al comentario que cierra la sección anterior, no debería ser sorprendente que Q^* no pueda ser modelado como un PARS, como los autores notan en el artículo.

En esta sección describiremos el cálculo y el criterio de confluencia que los autores demuestran. Daremos una descripción incompleta, centrándonos en el comportamiento probabilista y no determinista y siendo informales con los efectos cuánticos. La definición completa se encuentra en el Apéndice A.

En el Capítulo 3 daremos una extensión a los PARS en la que Q^* puede modelarse. Primero, damos una descripción breve de la computación cuántica.

Descripción breve e informal de la computación cuántica

La computación cuántica es un modelo de la física cuántica, sobre la cual no entraremos en mucho detalle[1]. Simplificando, podemos decir que existen los *qubit base* $|0\rangle$ y $|1\rangle$. Todo sistema cuántico (de un qubit) está formado por alguna *superposición* de los mismos, es decir, un vector normalizado de la forma $\alpha|0\rangle + \beta|1\rangle$ (con $\alpha, \beta \in \mathbb{C}$ tales que $|\alpha|^2 + |\beta|^2 = 1$). Los valores α, β se denominan *amplitudes*.

Podemos obtener estados base con facilidad. Para transformarlos y obtener estados más interesantes hay dos opciones: *aplicar una compuerta* o *medir*.

Las compuertas son necesariamente transformaciones lineales que preservan el módulo (llamadas *unitarias*). Por ejemplo, la compuerta de Hadamard H transforma a cada qubit base a una superposición de ambos con igual amplitud.

$$\begin{aligned} H|0\rangle &= \tfrac{1}{\sqrt{2}}|0\rangle + \tfrac{1}{\sqrt{2}}|1\rangle \\ H|1\rangle &= \tfrac{1}{\sqrt{2}}|0\rangle - \tfrac{1}{\sqrt{2}}|1\rangle \end{aligned}$$

Un qubit también puede *medirse* lo cual (además de dar un resultado) hace *colapsar* las superposiciones (modificándolas) a algún qubit base, con ciertas probabilidades bien definidas. Las amplitudes definen la probabilidad: en la superposición $\alpha|0\rangle + \beta|1\rangle$, con probabilidad $|\alpha|^2$ la medición resultará en $|0\rangle$ (con resultado 0) y con $|\beta|^2$ en $|1\rangle$ (con resultado 1).

[1] El lector interesado puede referirse a (Nielsen y Chuang 2011).

Para modelar lenguajes de programación cuánticos (que incluyan a la medición), debe usarse entonces una reducción probabilista. En general, tendremos reglas como:

$$\mathbf{meas}(\alpha\,|0\rangle + \beta\,|1\rangle) \to_{|\alpha|^2} 0 \qquad \mathbf{meas}(\alpha\,|0\rangle + \beta\,|1\rangle) \to_{|\beta|^2} 1$$

Dijimos que los sistemas de un qubit son representados por dos números complejos, o equivalentemente, un vector en $\mathbb{C}^2$. En sistemas de n qubits, el estado es un vector normalizado en el producto tensorial de los espacios individuales $\mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2$ que es (isomorfo a) $\mathbb{C}^{2^n}$.

El espacio tensorial permite comportamientos no locales, conocidos como *enredos cuánticos*. Formalmente, podemos tener un 2-qubit (en $\mathbb{C}^4$) que no puede ser descompuesto como $q_1 \otimes q_2$. Un ejemplo es:

$$\frac{1}{\sqrt{2}}(|0\rangle \otimes |0\rangle) + \frac{1}{\sqrt{2}}(|1\rangle \otimes |1\rangle)$$

Estos estados pueden conseguirse por la aplicación de compuertas unitarias. En los lenguajes, esto suele resultar en que todos los datos cuánticos (qubits) se modelan como algo indivisible (Selinger y Valiron 2005; Dal Lago, Masini y Zorzi 2011), aunque no es estrictamente necesario (van Tonder 2004; Díaz-Caro y col. 2011; Díaz-Caro y Dowek 2016).

Una propiedad interesante es que es imposible duplicar una superposición arbitraria. Es decir, la función

$$\psi \otimes |0\rangle \mapsto \psi \otimes \psi$$

es imposible de construir, debido a que no es lineal en ψ. Esto es conocido como el *teorema de no clonado*.

Sintaxis de Q*

Presentamos un subconjunto simplificado de la sintaxis, dado por la siguiente gramática (obviamos booleanos y pares).

$$\begin{array}{rcll} x & ::= & x_0 \mid x_1 \mid \dots & \textit{variables} \\ r & ::= & r_0 \mid r_1 \mid \dots & \textit{variables cuánticas} \\ U & ::= & U_0 \mid U_1 \mid \dots & \textit{operadores unitarios} \\ C & ::= & 0 \mid 1 \mid U & \textit{constantes} \\ M, N & ::= & x \mid r \mid C \mid M_1 M_2 \mid \lambda x.M & \\ & \mid & \mathbf{new}(M) \mid \mathbf{meas}(M) \mid !M \mid \lambda !x.M & \textit{términos} \end{array}$$

Las variables que representan qubits están diferenciadas de las estándar. Sobre ellas, se impone una restricción de *linealidad*: no pueden aparecer libres más de una vez. Esta restricción está formalizada con un juicio de buena formación, detallado en el Apéndice A.

Debido al teorema de no clonado, distinguimos dos tipos de abstracciones: *lineales* y *no lineales*. Las lineales usan sus argumentos exactamente una vez, siendo consistentes con el teorema de no clonado.

Las no lineales sí pueden duplicar sus argumentos, pero sus argumentos deben estar suspendidos como *thunks* ($!M$), aproximando una evaluación CBN (los términos $!M$ no reducen). Además, los thunks no pueden contener ninguna variable cuántica. De alguna forma, estos argumentos representan cómputos (que son duplicables) y no valores.

Las construcciones **new** y **meas** permiten crear y medir qubits, y las compuertas U nos permiten transformarlos. Notar que los qubits no están explícitos en los términos, sino que serán parte de un único estado fuera de los mismos, siguiendo el modelo de control clásico y datos cuánticos (Selinger 2004).

Semántica operacional

Definimos las *configuraciones* como ternas de la forma $[\mathcal{Q}, \mathcal{QV}, M]$, donde $\mathcal{Q}$ es un estado cuántico, $\mathcal{QV}$ es un mapeo de las variables cuánticas en M hacia el estado, y M un término.

La semántica operacional es una reducción probabilista entre configuraciones, de la forma $C \to^p_\alpha C'$ donde p es la probabilidad de transición[2] y α una *etiqueta*. La reducción permite no determinismo variando la etiqueta usada ($\mathsf{I}.\beta$, $\mathbf{meas}_r$, etc.). Podemos pensar que la elección de etiqueta modela cómo hacemos avanzar al sistema, o cómo pensamos su ejecución.

Algunas de las reglas interesantes son:

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, (\lambda x.M)N] \to^1_{\mathsf{I}.\beta} [\mathcal{Q}, \mathcal{QV}, M[N/x]]}$$

$$\frac{c \in \{0,1\} \quad p_c = \ldots}{[\mathcal{Q}, \mathcal{QV}, \mathbf{meas}(r)] \to^{p_c}_{\mathbf{meas}_r} [\mathcal{Q}', \mathcal{QV} - \{r\}, !c]}$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, M']}{[\mathcal{Q}, \mathcal{QV}, MN] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, M'N]} \qquad \frac{[\mathcal{Q}, \mathcal{QV}, N] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, N']}{[\mathcal{Q}, \mathcal{QV}, MN] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, MN']}$$

(No son relevantes aquí los detalles de $\mathcal{Q}'$ y p_c.)

De alguna forma (aproximada), las etiquetas son usadas para distinguir los distintos redexes en un término. Usando etiquetas distintas, podemos medir cualquiera de los n qubits presentes en el término, obteniendo una probabilidad total saliente de n. Debido a esto, no podemos modelar Q^* como un PARS.

Podríamos pensar que cada etiqueta forma un PARS por sí sola. Sin embargo, aun fijando la etiqueta, una configuración puede tener sucesores con probabilidad total mayor a 1.

[2] Usamos un superíndice para p, siguiendo la notación original.

Ejemplo 2.13 (Las etiquetas no forman un PARS)**.**

$$[\mathcal{Q}, \mathcal{QV}, (\lambda x.x)((\lambda y.fy)z)] \to^1_{\mathsf{l}.\beta} [\mathcal{Q}, \mathcal{QV}, (\lambda y.fy)z]$$
$$[\mathcal{Q}, \mathcal{QV}, (\lambda x.x)((\lambda y.fy)z)] \to^1_{\mathsf{l}.\beta} [\mathcal{Q}, \mathcal{QV}, (\lambda x.x)(fz)]$$

Esto podría evitarse dando alguna forma a las etiquetas. Por ejemplo, una opción que evitaría el problema es tomar al contexto en donde ocurre la reducción como la etiqueta, aunque no es algo muy elegante. Sin embargo, esto no es estrictamente un problema.

Árboles de computación Q^*

Para modelar el comportamiento probabilista, también usamos una noción de árbol de computación. En este caso, la expansión de un nodo es no determinista, ya que podemos elegir la etiqueta.

Definición 2.14. Dada una configuración C en Q^*, definimos los *árboles de computación con raíz* C (notado $\mathcal{T}^*(C)$) inductivamente[3] de la siguiente manera:

$$\frac{}{C \in \mathcal{T}^*(C)} \qquad \frac{\sum_{i=1}^n p_i = 1 \quad C \to^{p_i}_{\alpha} C_i \quad t_i \in \mathcal{T}^*(C_i)}{[C, (p_1, t_1), \ldots, (p_n, t_n)] \in \mathcal{T}^*(C)}$$

La definición es similar a los árboles anteriores, pero permitimos variar la etiqueta α en los nodos internos. Definimos análogamente $\mathrm{Prob}(T, C)$.

Notamos que todo nodo interno tiene exactamente 1 o 2 hijos. Para $\mathbf{meas}_r$ hay exactamente dos sucesores debido a que la variable cuántica r debe aparecer exactamente una vez en el término. En otro caso, si bien podemos tener muchos sucesores por la misma etiqueta, toda reducción tiene probabilidad 1, forzándonos a tomar uno sólo.

Observación 2.15. Sólo por este último hecho sobre $\mathbf{meas}_r$ es que tiene sentido la definición. Si una variable r pudiera aparecer dos veces podríamos expandirla en puntos distintos, o incluso sólo considerando el caso de resultado 0 y no el de 1.

En el siguiente capítulo daremos una noción de sistema de reescritura que puede modelar a Q^*, y tiene un enfoque algo más elegante en cuanto a las etiquetas y conjuntos sucesor. En dicho sistema podremos simular a todos los árboles Q^*.

[3] Los autores también razonan sobre árboles infinitos, usando una definición coinductiva. Seremos informales con respecto a los mismos.

La confluencia de Q^*

Debido a que en Q^* recuperamos una elección a la hora de reducir, una noción de confluencia vuelve a tener importancia.

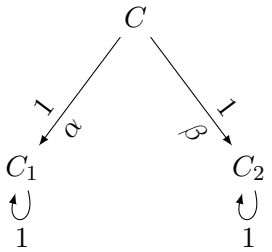
Los autores demuestran una propiedad de ese estilo que llaman *confluencia fuerte*. En esencia, dice que todos los árboles maximales con la misma raíz son equivalentes.

> **Teorema 2.16** (Dal Lago, Masini y Zorzi 2011)**.** *Sean $T_1, T_2 \in \mathcal{T}^*(C)$ maximales. Entonces para toda computación en forma normal D tenemos* $\mathrm{Prob}(T_1, D) = \mathrm{Prob}(T_2, D)$. *Además, la cantidad de hojas D es igual a cada lado.*

Por el momento, obviemos el comentario sobre la cantidad de hojas.

Vemos que el resultado es más similar a unicidad de formas normales que confluencia, por el hecho de limitarse a árboles maximales.

En efecto, el teorema no dice nada si tuviéramos algo como:



(las etiquetas omitidas no son relevantes) debido a que los dos árboles maximales con raíz C (necesariamente infinitos) no tienen elementos terminales. Sin embargo, son irreconciliablemente distintos, y nos gustaría rechazar estos comportamientos.

De hecho, por sobre el teorema, estos comportamientos no son posibles en Q^*. En los capítulos siguientes modelaremos Q^* con otro formalismo y demostraremos una confluencia estrictamente más fuerte, de manera simple.

Capítulo 3

No determinismo y probabilidad

> *"It should be observed first that the whole concept of a category is essentially an auxiliary one; our basic concepts are essentially those of a functor and of natural transformation."*
>
> —Eilenberg & Mac Lane, 1945

Como hemos visto, la noción de un PARS no permite elegir cómo reducir los elementos. Sin embargo, queremos expresar sistemas que sí permitan una elección y combinen probabilidad con no determinismo como nociones fundamentalmente distintas.

Para esto, definimos una noción de sistema de reescritura con ambas cualidades a los que llamamos MPARS y que generalizan tanto a los ARS como a los PARS. Interesado en definir lenguajes de programación, damos ejemplos de cómo pueden expresarse congruencias en un MPARS.

Al recuperar una elección no determinista, vuelve a tener importancia una noción análoga a la unicidad de formas normales. Para simplificar la tarea de demostrarla, usaremos una relación de reescritura entre distribuciones, más liberal que la evolución de árboles, inducida por cada MPARS. Sobre esa relación, definiremos una noción de *confluencia de distribuciones* que implica la unicidad buscada de manera directa.

Luego analizamos algunas consecuencias de la misma, posponiendo un análisis detallado sobre cómo demostrarla al siguiente capítulo. Además, mostramos cómo Q^* puede modelarse como MPARS y demostramos su confluencia de manera directa a partir de un lema sintáctico existente.

3.1. Sistemas multiprobabilistas — MPARS

La falta de no determinismo en los PARS se ve reflejada en el tipo de la función que lo define: la distribución sucesor para un elemento (si existe) es única. Aquí descartamos esa limitación y permitimos cualquier cantidad natural (o infinita) de ellas, para cada elemento.

De esta manera, el no determinismo y la probabilidad convivirán como conceptos distintos. Concretamente, la probabilidad modela la evolución incierta de los elementos mientras que el no determinismo será la elección de cómo razonamos o actuamos sobre el sistema.

A esta noción de reescritura con ambas cualidades la llamamos *sistemas de reescritura abstracta multiprobabilista (MPARS)*.[1]

Definición 3.1. Un MPARS $\mathcal{M}$ es un par $(A, \longmapsto)$ donde A es el conjunto portador y $\longmapsto$ una función de tipo $A \to \mathcal{P}(\mathscr{D}_1(A))$ llamada *función de evolución puntual*.

La función de evolución de un MPARS devuelve, para cada elemento a del portador, el conjunto (posiblemente infinito) de distribuciones sucesor a las que a puede evolucionar. De esta manera, un MPARS es una clara generalización a un PARS. Los elementos *terminales* son ahora aquellos a tales que $\longmapsto(a) = \emptyset$.

Observación 3.2. Dado que $\mathcal{P}(A \times A)$ es isomorfo a $A \to \mathcal{P}(A)$, tenemos la siguiente correspondencia (aproximada) entre los tipos que definen a cada sistema de reescritura.

ARS	$A \to \mathcal{P}(A)$
PARS	$A \to \mathscr{D}_1(A)$
MPARS	$A \to \mathcal{P}(\mathscr{D}_1(A))$

Observación 3.3. Un MPARS, en efecto, pone al no determinismo antes de o sobre la elección probabilista, como puede observarse en el tipo de $\longmapsto$. Otra opción podría haber sido tomarla de tipo $A \to \mathscr{D}_1(\mathcal{P}(A))$, donde las probabilidades de cada conjunto sucesor están fijas, pero uno puede luego elegir no determinísticamente dentro de ellos. La definición actual puede emular a esta alternativa, y es estrictamente más general.

La notación $a \to_p b$ para expresar que "a reescribe a b con probabilidad p" pierde sentido en el contexto de un MPARS. Un elemento a puede tener *distintas* probabilidades (o incluso la misma) de reescribir a b tomando distintas distribuciones sucesor. En otras palabras, p no es funcional en (a, b), ni tampoco lo es la distribución sucesor en (a, b, p). De todas maneras, y como veremos más adelante, no nos enfocaremos demasiado en esta relación entre elementos individuales.

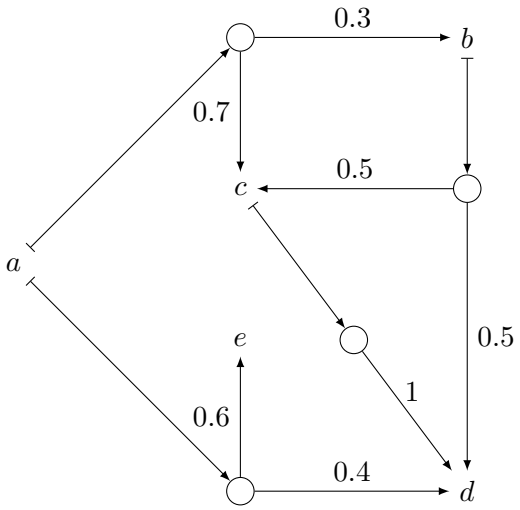
Usamos la notación $a \longmapsto A$ para expresar que $A \in \longmapsto(a)$. Podemos usarla para definir un MPARS por extensión.

[1] Un concepto similar fue definido en Bournez y Garnier 2005 y usado para estudiar una noción de normalización.

Ejemplo 3.4. Sea $\mathcal{M}$ el MPARS dado por

$$a \longmapsto \big[(0.3, b), (0.7, c)\big] \qquad a \longmapsto \big[(0.4, d), (0.6, e)\big]$$
$$b \longmapsto \big[(0.5, c), (0.5, d)\big] \qquad c \longmapsto \big[(1, d)\big]$$

En este caso, a es el único elemento que presenta una elección no determinista, y d y e son los elementos terminales. También podemos dar una representación gráfica:



donde cada círculo representa una distribución sucesor, compuestas por los elementos y probabilidades obvios implicados por el diagrama.

3.2. Permitiendo distintas estrategias

Con un MPARS, podemos dar una solución satisfactoria a la imposibilidad de elección en los PARS. La limitación anterior deja de ser un problema por el hecho de que puede haber cualquier cantidad de distribuciones sucesor, haciendo que un MPARS nunca esté saturado.

Ejemplo 3.5 (Continuación del Ejemplo 2.3)**.** Sea $\mathcal{M}$ el MPARS sobre los términos de L con $\longmapsto$ definida por:

$$\frac{}{\bigstar, L \longmapsto \big[(\tfrac{1}{2}, h; L), (\tfrac{1}{2}, t; L)\big]} \qquad \frac{L \longmapsto \big[(p_i, L_i)\big]_i}{C; L \longmapsto \big[(p_i, C; L_i)\big]_i}$$

Entonces tenemos:

$$\bigstar^2 \longmapsto \left[(\frac{1}{2}, h; \bigstar), (\frac{1}{2}, t; \bigstar)\right] \qquad \bigstar^2 \longmapsto \left[(\frac{1}{2}, \bigstar; h), (\frac{1}{2}, \bigstar; t)\right]$$

La estrategia de reducción será entonces equivalente a elegir una distribución sucesor para cada elemento no terminal. Notar que, de alguna forma, una estrategia es un "sub-PARS".

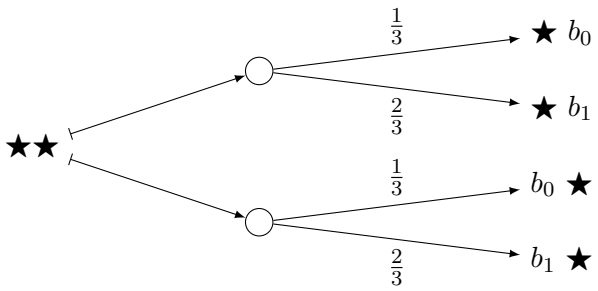
Yendo al contexto de un lenguaje de programación, no hay ningún inconveniente en definir congruencias, permitiendo liberalizar la semántica. A continuación se muestran las reglas de congruencia de λ_1, un cálculo multiprobabilista que será introducido en la Sección 5.2.

Ejemplo 3.6 (Congruencias en λ_1)**.**

$$\frac{M \longmapsto \left[(p_i, M_i)\right]_i}{MN \longmapsto \left[(p_i, M_i N)\right]_i} \text{ R-AppL} \qquad \frac{N \longmapsto \left[(p_i, N_i)\right]_i}{MN \longmapsto \left[(p_i, M N_i)\right]_i} \text{ R-AppR}$$

De esta manera, si M (resp. N) reduce a m (resp. n) distribuciones distintas, entonces MN tiene (al menos) $m + n$ distribuciones sucesor.

En λ_1, podemos formar un término $\bigstar$ tal que $\bigstar \longmapsto \left[(\frac{1}{3}, b_0), (\frac{2}{3}, b_1)\right]$, con b_0 y b_1 formas normales distintas. Si formamos el término $\bigstar\bigstar$, hay dos reducciones posibles.



Para modelar la ejecución de un MPARS también usaremos árboles de computación. Debido al no determinismo, debemos permitir elegir distintas distribuciones sucesor en los nodos internos (similarmente a Q^*).

Definición 3.7. Dado un elemento a de un MPARS, definimos los *árboles de computación con raíz* a (notado $\mathcal{T}(a)$) inductivamente de la siguiente manera:

$$\frac{}{a \in \mathcal{T}(a)} \qquad \frac{a \longmapsto \left[(p_i, a_i)\right]_i \quad t_i \in \mathcal{T}(a_i)}{[a, (p_1, t_1), \ldots, (p_n, t_n)] \in \mathcal{T}(a)}$$

Las nociones previas de árboles maximales y equivalentes se extienden directamente.

Generalizando ARS y PARS

La noción de MPARS generaliza tanto a los ARS como a los PARS. Esto no es muy sorprendente dada la comparación de tipos en la Observación 3.2, ya que $\mathcal{P}$ y $\mathscr{D}_1$ tienen inyecciones ($x \mapsto \{x\}$ y $x \mapsto [(1, x)]$). Veremos cómo, dado un ARS o PARS $\mathcal{A} = (A, \to)$, podemos embeberlos en un MPARS $\mathcal{M}$ equivalente.

ARS Definimos $\longmapsto$ como:

$$\frac{a \to b}{a \longmapsto [(1, b)]}$$

Luego, es fácil ver que todo árbol de computación $T \in \mathcal{T}(a)$ es una secuencia de reducciones $a \to a_1 \to \cdots \to a_n$ (en efecto, no hay ramificaciones probabilistas en un ARS) y viceversa.

PARS Similarmente, para los PARS, podemos definir la evolución de un elemento como su única distribución sucesor (o como el conjunto vacío, si el elemento es terminal). Formalmente, definimos

$$\frac{\sum_i p_i = 1 \quad a \to_{p_i} a_i}{a \longmapsto [(p_i, a_i)]_i}$$

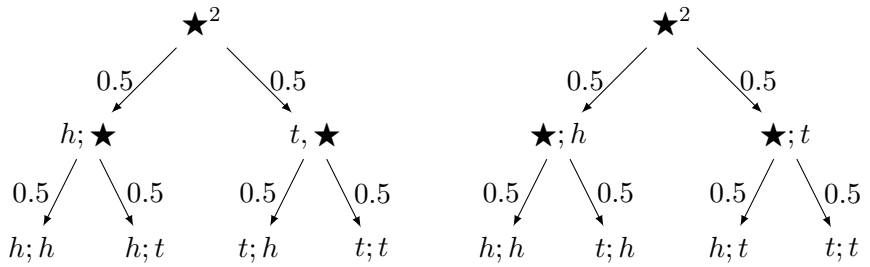
En este caso, los árboles de computación $T \in \mathcal{T}(a)$ son exactamente los árboles del PARS (notar que tomamos todas las permutaciones posibles para el lado derecho de $\longmapsto$).

Concluimos que podemos modelar a los ARS y PARS como MPARS.

Unicidad de distribuciones terminales

Debido a la elección no determinista, no es siempre cierto que un MPARS induce una única distribución. Esto es fácil de ver al tomar cualquier ARS que no cumpla UN y embeberlo como MPARS. Sin embargo, hay sistemas interesantes que sí cumplen la propiedad. Damos un ejemplo de una divergencia interesante.

> **Ejemplo 3.8.** Los siguientes árboles equivalentes modelan ejecuciones del Ejemplo 3.5.



Notar que expandimos el nodo $\bigstar^2$ de maneras distintas.

En un lenguaje probabilista, discutimos que un programa representa una distribución, y no un valor puntual. Por lo tanto, tener unicidad de distribuciones terminales garantiza que la semántica los programas es bien definida y única.

Entonces, demostrar la unicidad de distribuciones para un MPARS es de gran interés. En el caso tradicional, demostrarla de manera directa resulta a menudo difícil, por lo cual se suelen usar pruebas indirectas vía confluencia. Como los MPARS subsumen a los ARS, esperamos tener la misma dificultad. Por sobre UN, vimos otras propiedades favorables de la confluencia.

Necesitamos entonces una noción de confluencia adecuada a los MPARS. Una posibilidad sería tomar una noción de confluencia análoga a la del Lema 2.11. Sin embargo, razonar sobre los árboles de computación resulta rígido y difícil.

Entonces, la confluencia propuesta no será una propiedad sobre los árboles, sino sobre una reescritura entre distribuciones, definida en la sección siguiente.

3.3. Reduciendo distribuciones

Dado un MPARS $\mathcal{M} = (A, \longmapsto)$, podremos conseguir un ARS tradicional $\mathrm{Det}(\mathcal{M})$ cuyo soporte son las distribuciones de probabilidad sobre A. La reducción en $\mathrm{Det}(\mathcal{M})$, notada $\twoheadrightarrow$, está definida en base al MPARS.

Para reducir una distribución, usaremos a la función de evolución puntual para, como su nombre sugiere, evolucionar puntos individuales de la misma. Informalmente, los puntos (p, a) pueden transformarse en pA cuando $a \longmapsto A$.

Veremos luego (Capítulo 6) que $\mathrm{Det}(-)$ puede representar a todos los sistemas que reescriben distribuciones, en un sentido preciso.

Definición 3.9. Sea $\mathcal{M} = (A, \longmapsto)$ un MPARS. Definimos el ARS $\mathrm{Det}(\mathcal{M}) = (\mathscr{D}(A), \twoheadrightarrow)$, (llamado la *determinización* de $\mathcal{M}$) donde la relación $\twoheadrightarrow$ (llamada *evolución de distribuciones*) está dada por la siguiente definición

inductiva:

$$\frac{a \longmapsto A}{(p,a) : ds \twoheadrightarrow pA +\!\!+ ds}\ \textsc{Evolve} \qquad \frac{ds \twoheadrightarrow ds'}{(p,a) : ds \twoheadrightarrow (p,a) : ds'}\ \textsc{Tail}$$

$$\frac{}{(p_1,a_1) : (p_2,a_2) : ds \twoheadrightarrow (p_2,a_2) : (p_1,a_1) : ds}\ \textsc{Flip}$$

$$\frac{}{(p_1,a) : (p_2,a) : ds \twoheadrightarrow (p_1+p_2,a) : ds}\ \textsc{Join}$$

$$\frac{p = p_1 + p_2}{(p,a) : ds \twoheadrightarrow (p_1,a) : (p_2,a) : ds}\ \textsc{Split}$$

En la regla Split, está implícito que $p_1, p_2 \in \mathbb{R}^+$. Modelamos las equivalencias explícitamente para evitar informalidad (ya que es incómodo caracterizar a los sucesores de $\left[(1,a)\right]$ razonando módulo equivalencia). Sin embargo, muchas veces no seremos completamente formales en cuanto al orden de una distribución (sin que esto cause riesgo alguno).

Llamaremos *terminal* a una distribución compuesta solamente de elementos terminales. Notar que, debido a las equivalencias, no son formas normales en $\mathrm{Det}(\mathcal{M})$

Notamos con E ($\twoheadrightarrow_E$ de forma infija) y llamamos *evolución propia* a la subrelación que sólo hace uso de la regla Evolve y la congruencia Tail. Definimos a S, F y J de manera análoga, siempre incluyendo a Tail. También, usamos la notación (SFJ) para $S \cup F \cup J$, y análogamente para cualquier combinación. Notamos con $\sim$ a la relación (SFJ).

Al ser Tail la única congruencia, cada reducción es formada por n aplicaciones de esa regla y luego una aplicación de cualquier otra. Cuando queramos explicitar exactamente la reducción, usaremos la sintaxis $T^n R$ donde $R \in \{E, S, F, J\}$, con su significado obvio.

Notar que $\sim$ es una relación simétrica, ya que Flip se cancela a sí misma y Join y Split son inversas. Entonces, su clausura reflexiva transitiva es una relación de equivalencia, la cual notaremos con $\approx$. Esta relación captura nuestra noción de equivalencia de distribuciones. Notar que $\twoheadrightarrow^*$ es compatible con $\approx$.

Es claro que todo paso $\twoheadrightarrow$ es o bien una evolución propia o bien una equivalencia, es decir, $\twoheadrightarrow = \twoheadrightarrow_E \cup \sim$.

Ejemplo 3.10. Usando el MPARS del Ejemplo 3.4, podemos dar la si-

guiente secuencia de evolución de distribuciones

$$\begin{array}{rcl} \left[(1,a)\right] & \twoheadrightarrow_E & \left[(0.3,b),(0.7,c)\right] \\ & \twoheadrightarrow_E & \left[(0.15,c),(0.15,d),(0.7,c)\right] \\ & \sim & \left[(0.15,c),(0.7,c),(0.15,d)\right] \\ & \sim & \left[(0.85,c),(0.15,d)\right] \end{array}$$

Debe notarse que la regla SPLIT nos permite realizar *evoluciones parciales* de un punto de la distribución. Es decir, dado $p \in (0,1)$ y $a \longmapsto A$ podemos hacer

$$\left[(1,a)\right] \twoheadrightarrow \left[(1-p,a),(p,a)\right] \twoheadrightarrow (1-p,a) : pA$$

que corresponde a sólo evolucionar "una parte" del elemento a. En efecto, la evolución parcial nos da una reducción más liberal en comparación a los árboles.

Notar que $\mathrm{Det}(\mathcal{M})$ puede reescribir distribuciones no normalizadas (es decir, de peso total distinto a 1). Esto es intencional: si bien para modelar ejecuciones de un sistema sólo usaremos distribuciones normalizadas, no requerirlo será conveniente para poder razonar composicionalmente. Sin embargo, y como se espera, todas las reglas preservan el peso de una distribución.

Lema 3.11 (Preservación del peso)**.** *Si $D \twoheadrightarrow E$, entonces $w(D) = w(E)$.*

Demostración. Por inducción en $D \twoheadrightarrow E$, usando el hecho de que si $a \longmapsto A$ entonces $w(A) = 1$. ■

3.4. Simulando árboles en $\mathrm{Det}(\mathcal{M})$

Los soportes de los árboles de computación son distribuciones. En $\mathrm{Det}(\mathcal{M})$ podemos simular la evolución de (los soportes de) árboles de computación con evoluciones propias. Podemos formalizar esto con el siguiente lema:

Lema 3.12. *Dado un MPARS $\mathcal{M}$, tenemos en su determinización* $\mathrm{Det}(\mathcal{M})$ *que:*

$$\frac{T \in \mathcal{T}(a)}{\left[(1,a)\right] \twoheadrightarrow_E^* \mathrm{supp}(T)}$$

Demostración. Por inducción en el árbol T, usando que la evolución es *multiplicativa* y *composicional*, propiedades que serán formalizada en el siguiente capítulo (Corolario 4.8, Corolario 4.10). ■

La regla es invertible, en el sentido siguiente:

Lema 3.13. *En* $\mathrm{Det}(\mathcal{M})$ *tenemos.*

$$\frac{\left[(1,a)\right] \twoheadrightarrow_E^* D}{\exists T \in \mathcal{T}(a) \;/\; D = \mathrm{supp}(T)}$$

Demostración. Por inducción en los pasos de $\twoheadrightarrow_E$, usando que la evolución es *local* propiedad que será formalizada en el siguiente capítulo (Lema 4.11). ■

Sin embargo, la relación $\twoheadrightarrow$ permite más comportamientos, causados por la evolución parcial (es decir, por Split).

Ejemplo 3.14. Si traducimos el ARS $\mathbb{N}_r$ (Ejemplo 1.11) a un MPARS $\mathcal{M}$, tenemos en $\mathrm{Det}(\mathcal{M})$ que

$$\left[(1,5)\right] \twoheadrightarrow^* \left[(\tfrac{1}{3},5),(\tfrac{1}{3},5),(\tfrac{1}{3},5)\right] \twoheadrightarrow^* \left[(\tfrac{1}{3},4),(\tfrac{1}{3},3),(\tfrac{1}{3},2)\right]$$

Aun así, cuando embebemos un ARS en un MPARS, todo elemento de la distribución es sucesor del elemento original.

Lema 3.15. *Para la traducción de un ARS* $(A,\to)$ *a MPARS, tenemos que:*

$$\frac{\left[(1,a)\right] \twoheadrightarrow^* \left[(p_i,b_i)\right]_i}{a \to^* b_i}$$

Demostración. Por inducción en la cantidad de pasos y analizando la regla aplicada. ■

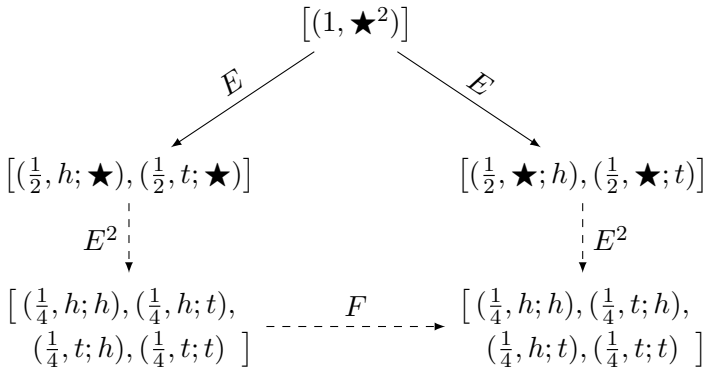
3.5. Confluencia de distribuciones

La noción de confluencia de distribuciones propuesta para un MPARS es la confluencia tradicional sobre su determinización. Es decir:

Definición 3.16 (Confluencia de distribuciones)**.** Dado un MPARS $\mathcal{M}$, decimos que una distribución D es *confluente* cuando sea confluente en $\mathrm{Det}(\mathcal{M})$ (en el sentido usual) y lo notamos $\mathrm{CR}(D)$. Llamamos al MPARS $\mathcal{M}$ *confluente en distribuciones* (o simplemente *confluente*, notado $\mathcal{M} \models \mathrm{CR}$) cuando $\mathrm{Det}(\mathcal{M})$ sea confluente.

Ejemplo 3.17. El MPARS del Ejemplo 3.5 es confluente en distribuciones. Esto es fácil de demostrar con un criterio que daremos luego, dado que el MPARS cumple una propiedad diamante (Teorema 4.29). Un ejemplo

de una divergencia interesante es:



Ejemplo 3.18. El MPARS del Ejemplo 3.4 no es confluente, ya que tenemos

$$[(1, a)] \twoheadrightarrow [(0.3, b), (0.7, c)] \qquad [(1, a)] \twoheadrightarrow [(0.4, d), (0.6, e)]$$

donde la distribución de la derecha es terminal y es claro que b y c nunca pueden evolucionar para dar e.

Demostrar que un MPARS es confluente parece (a priori) aún más difícil que el caso tradicional. En el capítulo siguiente daremos varios criterios que simplifican en gran manera la tarea, mostrando que no es fundamentalmente más complejo.

Por ahora, veremos algunas consecuencias de esta definición (análogas al caso tradicional) dando fe de que es una definición útil.

La confluencia implica unicidad de distribuciones terminales

Dado un MPARS confluente, es fácil demostrar que cumple con la unicidad de distribuciones terminales. Primero, damos una definición de la propiedad algo más liberal que la correspondiente a árboles.

Definición 3.19. Decimos que un MPARS $\mathcal{M}$ tiene *distribuciones terminales únicas* (y lo notamos $\mathcal{M} \models \mathrm{UTD}$) si siempre que $D_2 \twoheadleftarrow^* D \twoheadrightarrow^* D_1$ con D_1, D_2 terminales, entonces $D_1 \approx D_2$.

Por la simulación de árboles, es claro que UTD implica que los árboles maximales tienen soportes equivalentes (notar que T_1, T_2 son equivalentes si y sólo si $\mathrm{supp}(T_1) \approx \mathrm{supp}(T_2)$).

Ahora, podemos recuperar la implicancia esperada.

Lema 3.20. *Si D es terminal y $D \twoheadrightarrow^* E$, entonces $D \approx E$ con E terminal.*

Demostración. Es claro que si $D \twoheadrightarrow E$, entonces $D \sim E$ ya que la reducción no puede ser una evolución propia. Además, E debe ser terminal, ya que tiene exactamente los mismos elementos. Haciendo inducción en la cantidad de pasos, conseguimos el resultado. ■

Teorema 3.21. *Dado un MPARS $\mathcal{M}$, $\mathcal{M} \models \text{CR} \implies \mathcal{M} \models \text{UTD}$.*

Demostración. Supongamos que D evoluciona a D_1 y D_2 terminales. Entonces, como el sistema es confluente, sabemos existe C tal que $D_1 \twoheadrightarrow^* C \twoheadleftarrow^* D_2$. Por el lema anterior, tenemos $D_1 \approx C \approx D_2$, dando la equivalencia buscada. ■

Entonces, tener UTD implica que dos estrategias cualesquiera computan (si normalizan) la misma distribución terminal. En otras palabras, todo "sub-PARS" es equivalente. Por sobre ello, la noción de confluencia rechaza divergencias aun si no normalizan, como en el caso tradicional. Podemos extender trivialmente el Ejemplo 1.14 para verlo.

Notar que podemos demostrar la unicidad sobre árboles razonando con una noción de evolución más liberal (permitiendo la evolución parcial). Entonces, si uno rechaza la evolución parcial, la confluencia de distribuciones es aun adecuada para demostrar la unicidad de distribuciones terminales.

Consistencia ecuacional

Otra consecuencia importante de la confluencia tradicional es que la teoría ecuacional no iguala formas normales distintas (y por lo tanto, si existen al menos 2, es consistente).

Este resultado también se traslada de manera directa a las distribuciones.

Teorema 3.22. *Si $\mathcal{M} \models \text{CR}$, entonces para todas D_1, D_2 distribuciones terminales tenemos $D_1 \twoheadleftarrow\!\!\!\!\twoheadrightarrow^* D_2 \iff D_1 \approx D_2$.*

Demostración. La vuelta es trivial. Para la ida, notar que $\text{Det}(\mathcal{M})$ tiene la propiedad Church-Rosser (el Lema 1.17 aplica), y entonces D_1 y D_2 deben tener un reducto común C. Por el lema Lema 3.20, debemos tener $D_1 \approx C \approx D_2$. ■

Entonces, si un lenguaje tiene dos formas normales distintas M y N, la teoría es consistente ya que $\big[(1, M)\big] \twoheadleftarrow\!\!\!\!\not\twoheadrightarrow^* \big[(1, N)\big]$.

3.6. Q^* como MPARS y su confluencia

Además, de poder simular ARS y PARS, un MPARS permite nuevos comportamientos interesantes. Como vimos en la Sección 2.4, Q^* no es modelable como un PARS pero veremos que sí lo es como MPARS.

La elección no determinista en Q^* estaba dada por las distintas etiquetas asignadas a las reducciones. Entonces, es natural pensar en una correspondencia entre etiquetas y las distribuciones sucesor (aunque no pueda ser uno a uno, como muestra el Ejemplo 2.13).

Embeberemos Q^* en un MPARS $\mathcal{Q}^*$ equivalente. Similarmente a los PARS, definimos $\longmapsto$ como:

$$\frac{C \to_\alpha^{p_i} C_i \quad \sum_i p_i = 1}{C \longmapsto \left[(p_i, C_i)\right]_i}$$

Observación 3.23. Podríamos tomar un enfoque inverso y "traducir" cada regla de reducción Q^* a una evolución puntual. Por ejemplo, traducir

$$\frac{c \in \{0,1\} \quad p_c = \ldots}{[\mathcal{Q}, \mathcal{QV}, \mathbf{meas}(r)] \to_{\mathbf{meas}_r}^{p_c} [\mathcal{Q}', \mathcal{QV} - \{r\}, !c]}$$

a la regla

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, \mathbf{meas}(r)] \longmapsto \left[\begin{array}{l}(p_0, [\mathcal{Q}', \mathcal{QV} - \{r\}, !0]),\\ (p_1, [\mathcal{Q}'', \mathcal{QV} - \{r\}, !1])\end{array}\right]} \text{ R-MEAS}$$

y la congruencia

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, M']}{[\mathcal{Q}, \mathcal{QV}, MN] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, M'N]}$$

a la regla

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \longmapsto \left[(p_i, [\mathcal{Q}_i, \mathcal{QV}_i, M_i])\right]_i}{[\mathcal{Q}, \mathcal{QV}, MN] \longmapsto \left[(p_i, [\mathcal{Q}_i, \mathcal{QV}_i, M_iN])\right]_i} \text{ R-APPL}$$

Aunque sería una definición más elegante del cálculo, nos es más conveniente usar la definición previa en esta sección.

Dada esta definición, los árboles de computación Q^* (con su definición previa) y los árboles del MPARS $\mathcal{Q}^*$ coinciden exactamente (la demostración es trivial). Por lo tanto, tenemos las siguientes simulaciones:

Corolario 3.24. Tenemos en $\mathrm{Det}(\mathcal{Q}^*)$ que:

$$\frac{T \in \mathcal{T}^*(a)}{\big[(1,a)\big] \twoheadrightarrow_E^* \mathrm{supp}(T)} \qquad \frac{\big[(1,a)\big] \twoheadrightarrow_E^* D}{\exists T \in \mathcal{T}^*(a) \;/\; \mathrm{supp}(T) = D}$$

Concluimos que Q^* es modelable como un MPARS.

Confluencia de Q^*

Podemos demostrar que el MPARS $\mathcal{Q}^*$ definido anteriormente es confluente de manera simple reusando el siguiente lema sobre sus reducciones demostrado en (Dal Lago, Masini y Zorzi 2011).

Primero, se particionan las etiquetas en tres conjuntos: $\mathcal{N}$, $\mathcal{K}$ y $\{\mathbf{meas}_r\}$. La definición exacta de cada conjunto no viene al caso, pero se encuentra en el Apéndice A. Notamos con $C \to^p_{\mathcal{N}} C'$ cuando existe $\alpha \in \mathcal{N}$ tal que $C \to^p_\alpha C'$, y análogamente para $\mathcal{K}$.

El lema afirma algo similar a una propiedad diamante, pero con distintos comportamientos según el *tipo* de reducción aplicada. Además, permite "triángulos" en donde uno de los lados no reduce.

Lema 3.25 (Quasi-One-step Confluence para Q^*)**.** *Sean C, D, E configuraciones con $C \to^p_\alpha D$, $C \to^s_\beta E$, entonces:*

1. *Si $\alpha \in \mathcal{K}$ y $\beta \in \mathcal{K}$,*
   entonces o bien $D = E$ o existe F tal que $D \to^1_{\mathcal{K}} F$ y $E \to^1_{\mathcal{K}} F$.
2. *Si $\alpha \in \mathcal{K}$ y $\beta \in \mathcal{N}$,*
   entonces o bien $D \to^1_{\mathcal{N}} E$ o existe F tal que $D \to^1_{\mathcal{N}} F$ y $E \to^1_{\mathcal{K}} F$.
3. *Si $\alpha \in \mathcal{K}$ y $\beta = \boldsymbol{meas}_r$,*
   entonces existe F tal que $D \to^s_{\boldsymbol{meas}_r} F$ y $E \to^1_{\mathcal{K}} F$.
4. *Si $\alpha \in \mathcal{N}$ y $\beta \in \mathcal{N}$,*
   entonces o bien $D = E$ o existe F tal que $D \to^1_{\mathcal{N}} F$ y $E \to^1_{\mathcal{N}} F$.
5. *Si $\alpha \in \mathcal{N}$ y $\beta = \boldsymbol{meas}_r$,*
   entonces existe F tal que $D \to^s_{\boldsymbol{meas}_r} F$ y $E \to^1_{\mathcal{N}} F$.
6. *Si $\alpha = \boldsymbol{meas}_r$ y $\beta = \boldsymbol{meas}_q$ (con $r \neq q$),*
   entonces existen $t, u \in [0,1]$ y un F tal que $pt = su$ y $D \to^t_{\boldsymbol{meas}_q} F$ y $E \to^u_{\boldsymbol{meas}_r} F$.

De este lema, y el hecho de que no hay secuencias infinitas $\mathcal{K}$, los autores pueden demostrar la unicidad de formas normales (Teorema 2.16).

En el contexto del MPARS $\mathcal{Q}^*$ y su determinización, este lema tiene una interpretación más concisa. En la siguiente definición usamos la relación $\twoheadrightarrow_P$ que es un tipo de evolución paralela que puede evolucionar a cualquier cantidad de puntos de una distribución (incluso ninguno, haciéndola reflexiva).

Lema 3.26. *Sea C una configuración $\mathcal{Q}^*$ y D y E distribuciones. Si $D \longleftarrow\!\shortmid C \shortmid\!\longrightarrow E$, entonces existen D', E' equivalentes tales que $D \twoheadrightarrow_P D'$ y $E \twoheadrightarrow_P E'$.*

Demostración. Por la construcción de $\mathcal{Q}^*$, existen etiquetas α, β tales que $C \to_\alpha^{p_i} D_i$ para cada (p_i, D_i) en D y análogamente para E. Analizamos por casos en α, β, aplicando el Lema 3.25.

1. $\alpha \in \mathcal{K}$ y $\beta \in \mathcal{K}$.

   Forzosamente, D y E deben tener un único elemento. Tomemos $D = \left[(1, C_D)\right]$ y $E = \left[(1, C_E)\right]$.

   Si $C_D = C_E$, no hacemos ninguna evolución ($\twoheadrightarrow_P$ es reflexiva), y claramente tenemos distribuciones equivalentes. En otro caso, sabemos que existe C_F con $C_D \to_{\mathcal{K}}^1 C_F$ y $C_E \to_{\mathcal{K}}^1 C_F$, y convergemos en un paso a $\left[(1, C_F)\right]$ en ambas ramas.

2. $\alpha \in \mathcal{K}$ y $\beta \in \mathcal{N}$.

   Análogo a (1), notando que si $C_D \to_{\mathcal{N}}^1 C_E$ entonces convergemos en $\left[(1, C_E)\right]$ (aprovechando la reflexividad a la derecha).

3. $\alpha \in \mathcal{K}$ y $\beta = \mathbf{meas}_r$.

   En este caso, E debe tener exactamente dos elementos, resultando

   $$D = \left[(1, C_D)\right] \qquad E = \left[(p_0, C_0), (p_1, C_1)\right]$$

   Aplicando el lema dos veces, existen F_0, F_1 tales que

   $$C_D \to_{\mathbf{meas}_r}^{p_0} F_0 \qquad C_0 \to_{\mathcal{K}}^1 F_0 \qquad C_D \to_{\mathbf{meas}_r}^{p_1} F_1 \qquad C_1 \to_{\mathcal{K}}^1 F_1$$

   Por la construcción de $\mathcal{Q}^*$, tenemos

   $$\begin{array}{rcl} C_D & \longmapsto & \left[(p_0, F_0), (p_1, F_1)\right] \\ C_0 & \longmapsto & \left[(1, F_0)\right] \\ C_1 & \longmapsto & \left[(1, F_1)\right] \end{array}$$

   Aprovechando que $\twoheadrightarrow_P$ puede evolucionar ambos puntos de E, convergemos en $\left[(p_0, F_0), (p_1, F_1)\right]$.

4. $\alpha \in \mathcal{N}$ y $\beta \in \mathcal{N}$. Análogo a (1).

5. $\alpha \in \mathcal{N}$ y $\beta = \mathbf{meas}_r$. Análogo a (3).

6. $\alpha = \mathbf{meas}_r$ y $\beta = \mathbf{meas}_q$ (con $r \neq q$).

   Ambas D y E deben tener exactamente dos elementos, resultando

   $$D = \big[(d_0, C_0), (d_1, C_1)\big] \qquad E = \big[(e_0, C'_0), (e_1, C'_1)\big]$$

   Aplicando cuatro veces el lema, deben existir F_{ij}, t_{ij}, u_{ij} tales que

   $$\begin{array}{llll} C_0 & \to^{t_{00}}_{\mathbf{meas}_q} & F_{00} & C'_0 \to^{u_{00}}_{\mathbf{meas}_r} F_{00} \\ C_0 & \to^{t_{01}}_{\mathbf{meas}_q} & F_{01} & C'_1 \to^{u_{01}}_{\mathbf{meas}_r} F_{01} \\ C_1 & \to^{t_{10}}_{\mathbf{meas}_q} & F_{10} & C'_0 \to^{u_{10}}_{\mathbf{meas}_r} F_{10} \\ C_1 & \to^{t_{11}}_{\mathbf{meas}_q} & F_{11} & C'_1 \to^{u_{11}}_{\mathbf{meas}_r} F_{11} \end{array}$$

   Por la forma de la relación, debemos tener $t_{i0} + t_{i1} = 1$ y $u_{0j} + u_{1j} = 1$. Luego, tenemos $C_0 \longmapsto \big[(t_{00}, F_{00}), (t_{01}, F_{01})\big]$ y análogamente para los otros casos. Entonces, podemos evolucionar vía $\twoheadrightarrow_P$ a

   $$\big[(d_0 t_{00}, F_{00}), (d_0 t_{01}, F_{01}), (d_1 t_{10}, F_{10}), (d_1 t_{11}, F_{11})\big]$$

   y

   $$\big[(e_0 u_{00}, F_{00}), (e_0 u_{10}, F_{10}), (e_1 u_{01}, F_{01}), (e_1 u_{11}, F_{11})\big]$$

   Debido a que $e_j u_{ij} = d_i t_{ij}$ (por cada aplicación del lema), ambas distribuciones son equivalentes (luego de hacer un FLIP).

7. $\alpha = \mathbf{meas}_r$ y $\beta = \mathbf{meas}_q$ ($r = q$).

   Ambas ramas son iguales, convergemos por reflexividad. ■

En el Capítulo 4 veremos que este resultado es de hecho una propiedad diamante e implica la confluencia de $\mathrm{Det}(\mathcal{Q}^*)$ (Teorema 4.29). Por el Teorema 3.21, entonces, tenemos unicidad de formas normales dando el resultado anterior sobre $\mathbb{Q}^*$ de manera directa usando la simulación de árboles.

Demostración alternativa del Teorema 2.16. Sean $T_1, T_2 \in \mathcal{T}^*(C)$ maximales. Por la simulación del Corolario 3.24, tenemos que

$$\mathrm{supp}(T_2) \twoheadleftarrow^* \big[(1, C)\big] \twoheadrightarrow^* \mathrm{supp}(T_1)$$

Es claro que estos soportes son distribuciones terminales, y entonces, por UTD, tenemos $\mathrm{supp}(T_1) \approx \mathrm{supp}(T_2)$. Por lo tanto, T_1 es equivalente a T_2. ■

Pensamos que esta demostración contesta afirmativamente a la conjetura planteada por los diseñadores de $\mathbb{Q}^*$ de que todo lenguaje que cumpla "Quasi-One-step Confluence" será confluente en el sentido que proponen.

Por sobre UTD, sabemos además que no puede haber divergencias irreconcialiables (como las que mostramos al final de la Sección 2.4) ya que al tener confluencia, debe existir D tal que $\left[(1, C_1)\right] \twoheadrightarrow^* D \twoheadleftarrow^* \left[(1, C_2)\right]$. Es decir, aun para las computaciones que no normalizan, la confluencia de distribuciones nos dice algo sobre sus árboles.

No hemos demostramos que la cantidad de hojas sea la misma, como en el teorema original. Notando que en la demostración nunca usamos las reglas Join ni Split puede demostrarse que la relación $\twoheadrightarrow_{EF}$ es confluente, dando una noción de equivalencia más fuerte que implica el resultado anterior (ya que las hojas a cada lado deben ser exactamente las mismas, en algún orden). No se piensa que esto tenga alguna consecuencia relevante, y no lo desarrollamos aquí.

Capítulo 4

Demostrando confluencia

> *"Explanations exist; they have existed for all time; there is always a well-known solution to every human problem — neat, plausible, and wrong."*
>
> —Henry Louis Mecken, 1917

Como vimos en el primer capítulo, demostrar confluencia es complicado en el escenario tradicional. Por eso dimos muchos criterios para simplificar la tarea. En este capítulo tenemos la misma meta para los MPARS.

Nuestros primeros resultados consisten en abstraernos lo más posible de la noción de equivalencia, que es invariante de un MPARS a otro, permitiendo un uso más liberal de la misma. Luego, usando una descomposición de la evolución de distribuciones, simplificamos la forma de los diagramas a analizar. Con dicha simplificación se generalizan de manera directa algunos resultados tradicionales a este nuevo escenario, incluyendo un análogo al lema de Newman. Dados estos teoremas, notamos que demostrar confluencia no parece ser fundamentalmente más difícil que el caso tradicional.

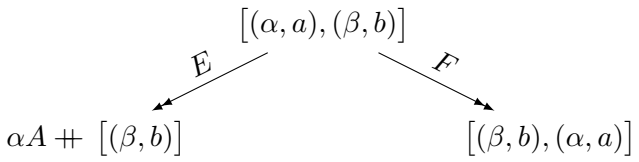
Daremos una prueba formal a cada afirmación, sacrificando algo de claridad expositiva. Como ventaja de este enfoque, esta sección debería ser mecanizable de una manera bastante directa.

A través del capítulo, se asume fijado un MPARS arbitrario $\mathcal{M}$ y razonamos sobre su determinización $\mathrm{Det}(\mathcal{M})$.

4.1. Ahorrando reducciones

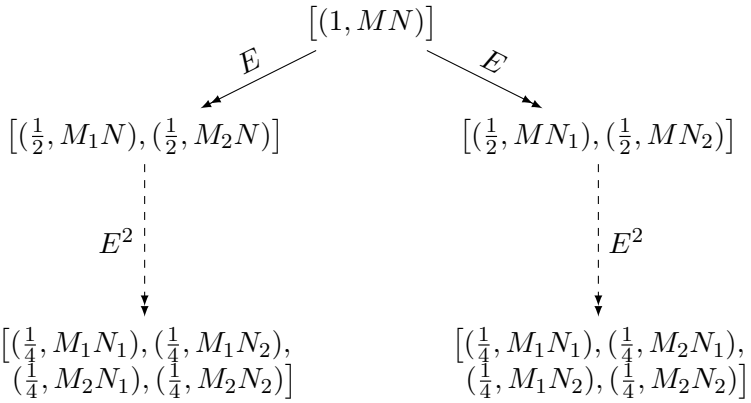
Al analizar confluencia de una relación, la cantidad de pasos es en ocasiones relevante sólo para permitir el razonamiento inductivo. El hecho de que la confluencia local no implique confluencia, y la propiedad diamante sí lo haga, es buen testigo de esto.

Contar las equivalencias como pasos parece poco eficiente. Si pensamos en la relación $\twoheadrightarrow$, veremos que rara vez podremos demostrar una propiedad diamante. Por ejemplo, intentemos cerrar el siguiente diagrama donde $a \longmapsto A$.



Podemos evolucionar a la derecha a $[(\beta, b)] + \alpha A$, pero la equivalencia con la rama izquierda nos llevará una cantidad arbitrariamente grande de pasos $\sim$ (dependiendo del tamaño de A).

Por otro lado, en el contexto de un cálculo con congruencias se da la siguiente situación si $M \longmapsto [(\frac{1}{2}, M_1), (\frac{1}{2}, M_2)]$ y similarmente para N.



La única forma, en general, de cerrar la parte superior será haciendo las *dos* evoluciones mostradas, en cada punto. Por sobre eso, las distribuciones están en orden distinto, y hace falta un paso F extra.

Ambas situaciones son decepcionantes ya que, intuitivamente, cerrar en dos distribuciones equivalentes debería ser suficiente. Además, deberíamos poder evolucionar los puntos de una distribución al mismo tiempo, ya que son alternativas distintas de una ejecución. Concretamente, las congruencias no eran un problema para la propiedad diamante y ahora lo son.

Resolveremos ambos problemas cambiando de relación.

Evolución paralela

Definiremos la relación $\twoheadrightarrow_P$ llamada *evolución paralela*. Esta idea es análoga en espíritu a la β-reducción paralela definida en el primer capítulo, y tiene un propósito similar: actuar sobre varias copias a la vez.

Podemos definirla inductivamente con las reglas:

$$\frac{}{[] \twoheadrightarrow_P []}\ \textsc{Par-}[] \qquad \frac{a \longmapsto A \quad ds \twoheadrightarrow_P ds'}{(p,a) : ds \twoheadrightarrow_P pA +\!\!+ ds'}\ \textsc{Par}_1$$

$$\frac{ds \twoheadrightarrow_P ds'}{(p,a) : ds \twoheadrightarrow_P (p,a) : ds'}\ \textsc{Par}_0$$

Intuitivamente, una reducción $\twoheadrightarrow_P$ realiza evoluciones en algunos de los elementos de la distribución (posiblemente todos o ninguno). Una primer propiedad a notar es que, como se espera, puede ser simulada por $\twoheadrightarrow_E$.

Lema 4.1. $\twoheadrightarrow_P \subseteq \twoheadrightarrow_E^*$

Demostración. Por inducción en la forma de $\twoheadrightarrow_P$.

- Par-[].

  Tenemos $[] \twoheadrightarrow_P []$. Concluimos por $[] \twoheadrightarrow_E^0 []$.

- $\textsc{Par}_1$.

  $(p,a) : ds \twoheadrightarrow_P pA +\!\!+ ds'$. Por inversión sabemos que $a \longmapsto A$ y $ds \twoheadrightarrow_P ds'$. Por HI tenemos $ds \twoheadrightarrow_E^* ds'$. Entonces hacemos $(p,a) : ds \twoheadrightarrow_E^* (p,a) : ds' \twoheadrightarrow_E pA +\!\!+ ds'$.

- $\textsc{Par}_0$.

  $(p,a) : ds \twoheadrightarrow_P (p,a) : ds'$. Por inversión sabemos que $ds \twoheadrightarrow_P ds'$. Por HI tenemos $ds \twoheadrightarrow_E^* ds'$. Entonces aplicamos repetidamente Tail para conseguir $(p,a) : ds \twoheadrightarrow_E^* (p,a) : ds'$. ■

Además, claramente un paso de $\twoheadrightarrow_E$ es un paso de $\twoheadrightarrow_P$

Lema 4.2. $\twoheadrightarrow_E \subseteq \twoheadrightarrow_P$

Demostración. Trivial. ■

Ambos lemas nos garantizan que sus clausuras reflexivas transitivas son exactamente iguales.

Corolario 4.3. $\twoheadrightarrow_E^* = \twoheadrightarrow_P^*$ y entonces $\twoheadrightarrow^* = (\twoheadrightarrow_E \cup \sim)^* = (\twoheadrightarrow_P \cup \sim)^*$.

Demostración. La primera afirmación se sigue por el Lema 1.4 usando los dos lemas previos. La segunda se sigue del Lema 1.5. ■

Equivalencia gratis

Dado el Corolario 4.3 podríamos analizar la confluencia de la relación $(\twoheadrightarrow_P \cup \sim)$. Sin embargo, seguimos teniendo el problema de que las equivalencias tienen un costo asociado. Cambiaremos a una relación en donde esto no ocurre.

Concretamente, demostraremos que podemos analizar la confluencia de la expansión módulo equivalencia: $(\twoheadrightarrow_P / \approx)$.

Lema 4.4. *La relación $\twoheadrightarrow_P$ es reflexiva*

Demostración. Por inducción sobre la distribución. Para el caso base usamos PAR-$[]$ y para el paso inductivo PAR_0. ■

Lema 4.5. *Las relaciones $\twoheadrightarrow^*$ y $(\twoheadrightarrow_P / \approx)^*$ coinciden.*

Demostración. Usando el Lema 1.4.

- $\twoheadrightarrow \subseteq (\twoheadrightarrow_P / \approx)^*$

  Analizamos por casos, dado que $\twoheadrightarrow = \twoheadrightarrow_E \cup \sim$.

  Por el Lema 4.2, $\twoheadrightarrow_E \subseteq \twoheadrightarrow_P$ y $\approx$ es reflexiva (por definición). Por lo tanto $\twoheadrightarrow_E \subseteq \approx \cdot \twoheadrightarrow_P \cdot \approx$. Además, $\sim \subseteq \approx$ y entonces $\sim \subseteq \approx \cdot \twoheadrightarrow_P \cdot \approx$ (porque $\approx$ y $\twoheadrightarrow_P$ son reflexivas). Entonces, la unión también está contenida.

- $(\twoheadrightarrow_P / \approx) \subseteq \twoheadrightarrow^*$

  Como $\sim \subseteq \twoheadrightarrow$, tenemos $\approx \subseteq \twoheadrightarrow^*$. Usando el Lema 4.1 y que $\twoheadrightarrow_E \subseteq \twoheadrightarrow$ tenemos

  $$\twoheadrightarrow_P \subseteq \twoheadrightarrow_E^* \subseteq \twoheadrightarrow^*$$

  Entonces,

  $$\approx \cdot \twoheadrightarrow_P \cdot \approx \subseteq \twoheadrightarrow^* \cdot \twoheadrightarrow^* \cdot \twoheadrightarrow^*$$

  Por transitividad de la clausura, llegamos al resultado buscado. ■

Por lo tanto, sus confluencias son equivalentes.

Corolario 4.6. $\twoheadrightarrow \models \text{CR} \iff \twoheadrightarrow_P / \approx \models \text{CR}$

De ahora en adelante, usando este corolario, consideraremos la confluencia de $\twoheadrightarrow_P / \approx$. Entonces podemos (en la parte inferior de un diagrama) realizar una equivalencia, evolucionar a todos los puntos de la distribución y luego realizar otra equivalencia en un único paso.

El problema ahora no parece necesariamente más fácil: seguimos teniendo equivalencias en la parte superior, y mezcladas arbitrariamente con las evoluciones. Las siguientes secciones eliminan este problema. Primero, demostramos algunos lemas auxiliares.

Lemas auxiliares

Notar que muchos de estos lemas también aplican a la evolución propia $\twoheadrightarrow_E$, dado el Corolario 4.3.

Lema 4.7 (Multiplicidad). *Para* $\alpha \in \mathbb{R}^+$, $D \twoheadrightarrow_P E \iff \alpha D \twoheadrightarrow_P \alpha E$.

Demostración. **Ida.** Por inducción en $D \twoheadrightarrow_P E$.

- Par-$[]$. Trivial.
- Par_0.

  Tenemos $D = (p, a) : ds \twoheadrightarrow_P (p, a) : ds' = E$. Por la premisa tenemos $ds \twoheadrightarrow_P ds'$. Aplicando la HI tenemos $\alpha ds \twoheadrightarrow_P \alpha ds'$. Entonces, por Par_0, tenemos $(\alpha p, a) : \alpha ds \twoheadrightarrow_P (\alpha p, a) : \alpha ds'$, como se buscaba.

- Par_1.

  Tenemos $D = (p, a) : ds \twoheadrightarrow_P pA +\!\!+ ds' = E$. Por la premisa tenemos $ds \twoheadrightarrow_P ds'$ y $a \longmapsto A$. Aplicando la HI tenemos $\alpha ds \twoheadrightarrow_P \alpha ds'$. Entonces, por Par_1, tenemos $(\alpha p, a) : \alpha ds \twoheadrightarrow_P (\alpha p)A +\!\!+ \alpha ds'$, como se buscaba (ya que $(\alpha p)A = \alpha(pA)$).

Vuelta. Aplicando la ida con α^{-1}. ■

Por inducción en la cantidad de pasos, el lema previo se extiende a la clausura.

Corolario 4.8. Para $\alpha \in \mathbb{R}^+$, $D \twoheadrightarrow_P^* E \iff \alpha D \twoheadrightarrow_P^* \alpha E$.

Lema 4.9 (Composicionalidad). *Si* $D \twoheadrightarrow_P D'$ *y* $E \twoheadrightarrow_P E'$, *entonces* $D +\!\!+ E \twoheadrightarrow_P D' +\!\!+ E'$.

Demostración. Por inducción en $D \twoheadrightarrow_P D'$.

- Par-$[]$.

  Tenemos $D = D' = []$ y concluimos por la hipótesis de $E \twoheadrightarrow_P E'$.

- Par_0.

  Tenemos $D = (p, a) : ds$ y $D' = (p, a) : ds'$ con $ds \twoheadrightarrow_P ds'$. Aplicando la HI a la premisa conseguimos $ds +\!\!+ E \twoheadrightarrow_P ds' +\!\!+ E'$. Por Par_0 concluimos $(p, a) : ds +\!\!+ E \twoheadrightarrow_P (p, a) : ds' +\!\!+ E'$, como se buscaba.

- $\textsc{Par}_1$.

  Tenemos $D = (p, a) : ds$ y $D' = pA +\!\!+ ds'$ con $ds \twoheadrightarrow_P ds'$ y $a \longmapsto A$. Aplicando la HI a la primer premisa conseguimos $ds +\!\!+ E \twoheadrightarrow_P ds' +\!\!+ E'$. Por $\textsc{Par}_1$ concluimos $(p, a) : ds +\!\!+ E \twoheadrightarrow_P pA +\!\!+ ds' +\!\!+ E'$, como se buscaba. ■

Corolario 4.10. Si $D \twoheadrightarrow_P^* D'$ y $E \twoheadrightarrow_P^* E'$, entonces $D +\!\!+ E \twoheadrightarrow_P^* D' +\!\!+ E'$.

Tanto Evolve como Split no alteran el orden de una distribución.

Lema 4.11. *Si $D_1 +\!\!+ D_2 \twoheadrightarrow_P E$ entonces existen E_1, E_2 tales que $E = E_1 +\!\!+ E_2$ y $D_i \twoheadrightarrow_P E_i$ para $i = 1, 2$.*

Demostración. Por inducción en D_1.

- $D_1 = \left[\right]$. Tomar $E_1 = \left[\right]$ y $E_2 = E$.
- $D_1 = (p, a) : D'$. Analizamos según la regla aplicada.
  - $\textsc{Par}_0$

    Tenemos $(p, a) : D' +\!\!+ D_2 \twoheadrightarrow_P (p, a) : E'$ con $D' +\!\!+ D_2 \twoheadrightarrow_P E'$. Por la HI, tenemos que $E' = E'_1 +\!\!+ E'_2$ con $D' \twoheadrightarrow_P E'_1$ y $D_1 \twoheadrightarrow_P E'_2$. Por $\textsc{Par}_0$ tenemos que $(p, a) : D' \twoheadrightarrow_P (p, a) : E'_1$. Entonces, tomamos $E_1 = (p, a) : E'_1$ y $E_2 = E'_2$.
  - $\textsc{Par}_1$

    Tenemos $(p, a) : D' +\!\!+ D_2 \twoheadrightarrow_P pA : E'$ con $D' +\!\!+ D_2 \twoheadrightarrow_P E'$ y $a \longmapsto A$. Por la HI, tenemos que $E' = E'_1 +\!\!+ E'_2$ con $D' \twoheadrightarrow_P E'_1$ y $D_1 \twoheadrightarrow_P E'_2$. Por $\textsc{Par}_1$ tenemos que $(p, a) : D' \twoheadrightarrow_P pA +\!\!+ E'_1$. Entonces, tomamos $E_1 = pA +\!\!+ E'_1$ y $E_2 = E'_2$. ■

Lema 4.12. *Si $D_1 +\!\!+ D_2 \twoheadrightarrow_S^= E$ entonces existen E_1, E_2 tales que $E = E_1 +\!\!+ E_2$ y $D_i \twoheadrightarrow_S^= E_i$ para $i = 1, 2$.*

Demostración. Si el paso es una reflexividad, el resultado es trivial. Si no, por inducción en D_1, similar a la anterior. ■

Notaremos con $\twoheadrightarrow_{SP}$ a la relación $(\twoheadrightarrow_S \cup \twoheadrightarrow_P)$.

Lema 4.13 (Localidad de Split y Evolve)**.** *Si $\left[(p_i, a_i)\right]_i \twoheadrightarrow_{SP}^* N$ entonces existen E_i tales que $E = p_1E_1 +\!\!+ \cdots +\!\!+ p_nE_n$ y $\left[(1, a_i)\right] \twoheadrightarrow_{SP}^* E_i$.*

Demostración. Por inducción en la cantidad de pasos, ambos lemas anteriores se extienden a las clausuras. Podemos aplicarlos iteradamente y obtener:

$$\left[(p_i, a_i)\right]_i = \left[(p_1, a_1)\right] + \ldots + \left[(p_n, a_n)\right] \twoheadrightarrow_{SP}^* E \implies E = E_1' + \cdots + E_n'$$

con $\left[(p_i, a_i)\right] \twoheadrightarrow_{SP}^* E_i'$ para cada i. Por multiplicidad tenemos que $\left[(1, a_i)\right] \twoheadrightarrow \frac{1}{p_i} E_i'$. Tomamos entonces $E_i = \frac{1}{p_i} E_i'$ ■

Cambiar uniformemente los elementos de una distribución preserva la equivalencia, como dicta el siguiente lema.

Lema 4.14. *Sea $f : A \to B$ función y $D \sim E$, entonces $\hat{f}(D) \sim \hat{f}(E)$.*

Demostración. Por inducción en $D \sim E$.

- Flip.
  Tenemos $(p, a) : (q, b) : ds \twoheadrightarrow (q, b) : (p, a) : ds$.
  Entonces $(p, f(a)) : (q, f(b)) : \hat{f}(ds) \twoheadrightarrow (q, f(b)) : (p, f(a)) : \hat{f}(ds)$.
- Join.
  Tenemos $(p_1, a) : (p_2, a) : ds \twoheadrightarrow (p_1 + p_2, a) : ds$.
  Entonces $(p_1, f(a)) : (p_2, f(a)) : \hat{f}(ds) \twoheadrightarrow (p_1 + p_2, f(a)) : \hat{f}(ds)$.
- Split. Similar al caso anterior.
- Tail. Aplicar HI. ■

Corolario 4.15. Si $D \approx E$ entonces $\hat{f}(D) \approx \hat{f}(E)$.

Demostración. Por inducción en la cantidad de pasos usando el Lema 4.14. ■

4.2. Descomponiendo la equivalencia

Dos distribuciones son equivalentes cuando tienen los mismos elementos, con exactamente el mismo peso total para cada uno. Esto nos indica que podríamos decidir la equivalencia entre dos distribuciones reduciéndolas a una forma canónica donde cada elemento aparece una única vez. Si bien cierto, nos será más útil un criterio de espíritu inverso.

Mostraremos que toda equivalencia puede hacerse dividiendo (con Split) los elementos de una distribución, y luego reordenándolos y reagrupándolos (con Flip y Join). Es decir, formalmente

$$\approx \quad = S^* \cdot (FJ)^*$$

Esto es consecuencia de una conmutación secuencial.

Lema 4.16. $S \dashv (FJ)^*$

Demostración. Usando el criterio simplificado del Lema 1.37 y por inducción en la forma de las reducciones. La demostración completa se encuentra en el Apéndice B.1. ■

Corolario 4.17. $(SFJ)^* = S^* \cdot (FJ)^*$

Demostración. Aplicando el Lema 4.16 y Lema 1.39. ■

Entonces, sabemos que para cada equivalencia podemos asumir que fue hecha en este orden. Se puede demostrar que, por sobre esto, los FLIP siempre pueden ponerse antes de los JOIN (es decir $F^* \dashv J^*$), pero no usaremos este resultado.

4.3. Evolución y equivalencia

Estudiaremos como interactúa la evolución con la equivalencia. Demostramos dos conmutaciones interesantes, que nos permitirán descomponer la relación $\twoheadrightarrow_P \,/\, \approx$ a una forma más manejable.

La primera dice que siempre podemos evolucionar antes de aplicar las reglas FLIP y JOIN. Más aún, usando la evolución paralela, mantenemos exactamente la cantidad de evoluciones.

Lema 4.18. $\twoheadrightarrow_P \dashv (FJ)^*$.

Demostración. Usando el criterio simplificado del Lema 1.37 y por inducción en la forma de las reducciones. La demostración completa se encuentra en el Apéndice B.1. ■

Además, SPLIT *casi* conmuta sobre la evolución paralela: podemos mover los pasos de SPLIT sobre la evolución paralela (manteniendo ambas cantidades), pero hacen falta algunos pasos de reordenamiento luego. Esto no será un problema, como veremos.

Lema 4.19. $\twoheadrightarrow_P \cdot S \subseteq S \cdot \twoheadrightarrow_P \cdot (FJ)^*$

Demostración. Analizando la forma de las distribuciones. La demostración completa se encuentra en el Apéndice B.1. ■

El lema anterior se generaliza a la clausura.

Lema 4.20. $\twoheadrightarrow_P \cdot S^* \subseteq S^* \cdot \twoheadrightarrow_P \cdot (FJ)^*$

Demostración. Por inducción en la cantidad de pasos S. El caso base es trivial. Para el paso inductivo, debemos mostrar que $\twoheadrightarrow_P \cdot S^{n+1} \subseteq S^{n+1} \cdot \twoheadrightarrow_P \cdot (FJ)^*$.

$\twoheadrightarrow_P \cdot S \cdot S^n$
$\subseteq$ { *Lema 4.19* }
$S \cdot \twoheadrightarrow_P \cdot (FJ)^* \cdot S^n$
$\subseteq$ { *Por el Lema 4.16 tenemos también* $S^n \dashv (FJ)^*$ }
$S \cdot \twoheadrightarrow_P \cdot S^n \cdot (FJ)^*$
$\subseteq$ { *HI* }
$S \cdot S^n \cdot \twoheadrightarrow_P \cdot (FJ)^* \cdot (FJ)^*$
$\subseteq$ { *Transitividad* }
$S \cdot S^n \cdot \twoheadrightarrow_P \cdot (FJ)^*$ ■

El siguiente teorema enuncia un resultado de suma utilidad, ya que acota las equivalencias a cada lado de una evolución.

Teorema 4.21. *La relación* $\twoheadrightarrow_P / \approx$ *coincide con* $S^* \cdot \twoheadrightarrow_P \cdot (FJ)^*$.

Demostración. La dirección $\supseteq$ es trivial. Demostraremos la otra dirección.

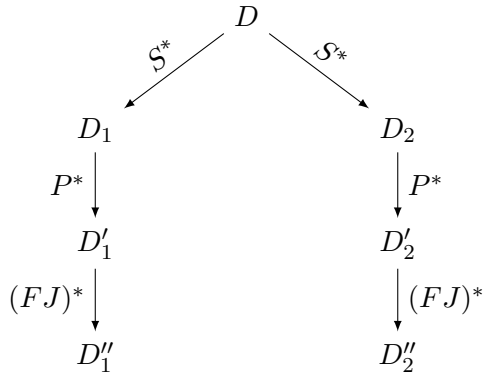
$\approx \cdot \twoheadrightarrow_P \cdot \approx$
$=$ { *Definición* }
$(SFJ)^* \cdot \twoheadrightarrow_P \cdot (SFJ)^*$
$\subseteq$ { *Corolario 4.17* }
$S^* \cdot (FJ)^* \cdot \twoheadrightarrow_P \cdot S^* \cdot (FJ)^*$
$\subseteq$ { $\twoheadrightarrow_P \dashv (FJ)^*$ }
$S^* \cdot \twoheadrightarrow_P \cdot (FJ)^* \cdot S^* \cdot (FJ)^*$
$\subseteq$ { $S^* \dashv (FJ)^*$ }
$S^* \cdot \twoheadrightarrow_P \cdot S^* \cdot (FJ)^* \cdot (FJ)^*$
$\subseteq$ { *Lema 4.20* }
$S^* \cdot S^* \cdot \twoheadrightarrow_P \cdot (FJ)^* \cdot (FJ)^* \cdot (FJ)^*$
$\subseteq$ { *Transitividad* }
$S^* \cdot \twoheadrightarrow_P \cdot (FJ)^*$ ■

Por inducción, y usando las conmutaciones anteriores, se puede demostrar lo mismo para mayor cantidad de pasos.

Corolario 4.22. La relación $(\twoheadrightarrow_P / \approx)^n$ coincide con $S^* \cdot \twoheadrightarrow_P^n \cdot (FJ)^*$.

Corolario 4.23. La relación $(\twoheadrightarrow_P / \approx)^*$ coincide con $S^* \cdot \twoheadrightarrow_P^* \cdot (FJ)^*$.

Este resultado será de mucha utilidad, ya que ahora al analizar una parte superior de un diagrama de confluencia, podemos asumir que las reducciones tienen esta forma. Para ilustrar el punto, podemos asumir que todos los diagramas de confluencia tienen la forma:



donde debemos cerrar D_1'' y D_2'' usando cualquier regla.

Más aún, la cantidad de evoluciones $\twoheadrightarrow_P$ se preserva exactamente, lo que nos habilitará a demostrar criterios análogos a la propiedad diamante o semiconfluencia.

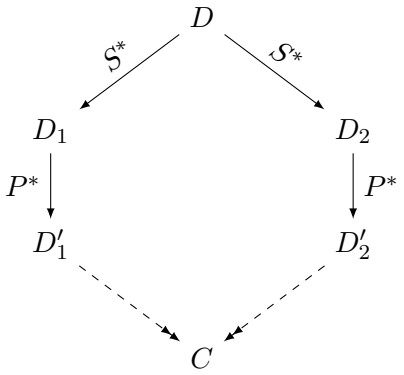
4.4. Simplificando diagramas

Aunque acotamos su forma, la equivalencia sigue apareciendo en la parte superior de los diagramas de confluencia. En esta sección, nos encargaremos de eliminarla por completo de allí, dejándola sólo como una herramienta que podemos usar para cerrar un diagrama. En otras palabras: "la equivalencia sólo ayuda, nunca molesta".

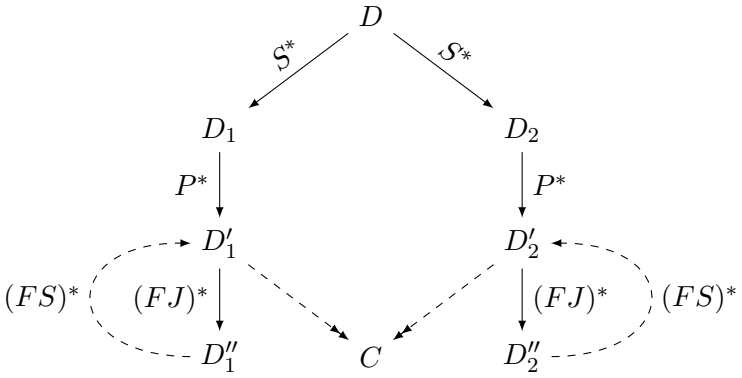
Veamos como simplificar un D-diagrama.

Cancelar Joins y Flips

Una primera simplificación a los diagramas anteriores es notar que Flip y Join son invertibles, con lo cual podemos deshacer sus efectos. Es decir, si podemos cerrar los diagramas de la forma:



ya es suficiente para cerrar todos los D-diagramas. Gráficamente, haríamos algo como:

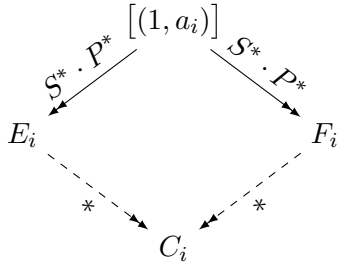


Singletons

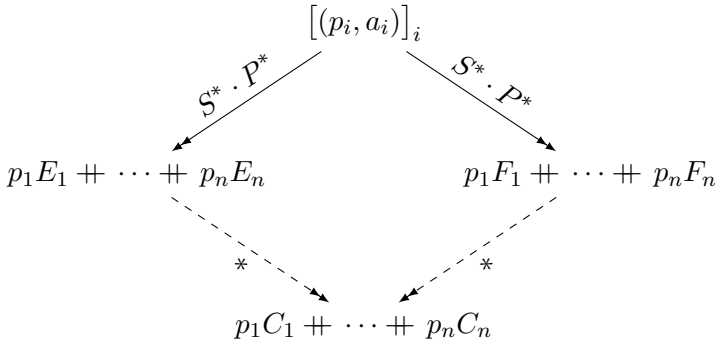
Como consecuencia de lo anterior, los únicos pasos relevantes en una parte superior son Split y la evolución paralela, en ese orden. Analizaremos la forma de D.

Tenemos $D = \big[(p_i, a_i)\big]_i$, con n elementos. Como $D \twoheadrightarrow^*_{SP} D'_1$ sabemos por el Lema 4.13 que $D'_1 = p_1 E_1 + \cdots + p_n E_n$ donde para cada i, $\big[(1, a_i)\big] \twoheadrightarrow^*_{SP} E_i$; y análogamente $D'_2 = p_1 F_1 + \cdots + p_n F_n$.

Supongamos que cada singleton $\big[(1, a_i)\big]$ es confluente. Es decir, que podemos cerrar los diagramas de la forma:



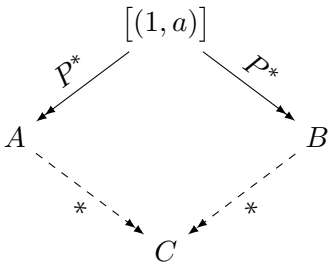
Por multiplicidad y composicionalidad (Lema 4.7 y Lema 4.9), podemos componer estas confluencias para cerrar D:



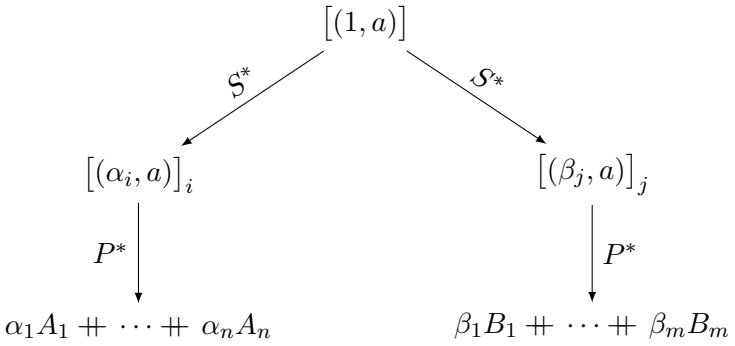
Por lo tanto, basta con demostrar que los elementos son confluentes para conseguir el resultado sobre la distribución.

Abajo con los Splits

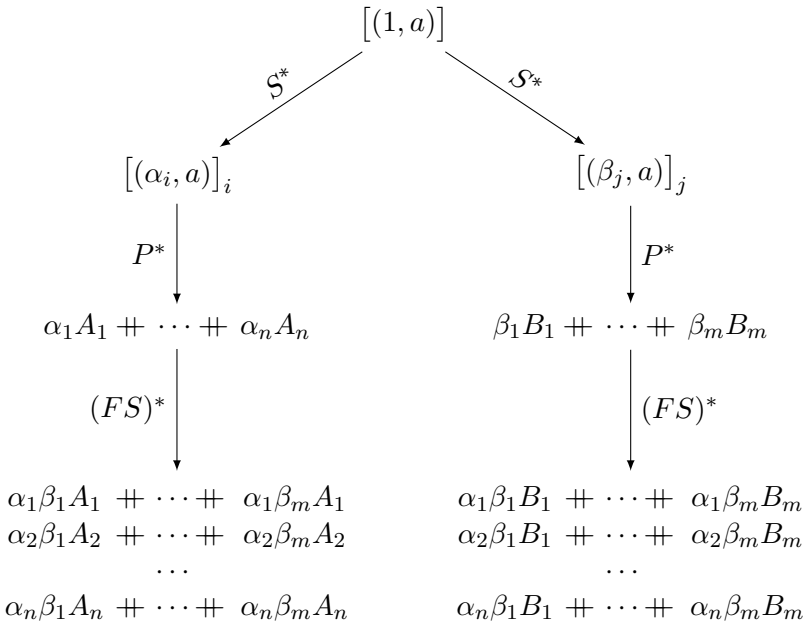
Los Splits en la parte superior siguen molestando. Veremos que es suficiente con cerrar los D-diagramas de sólo evolución. Supongamos que podemos cerrar todo diagrama sin Splits para un elemento a. Es decir, que tenemos:



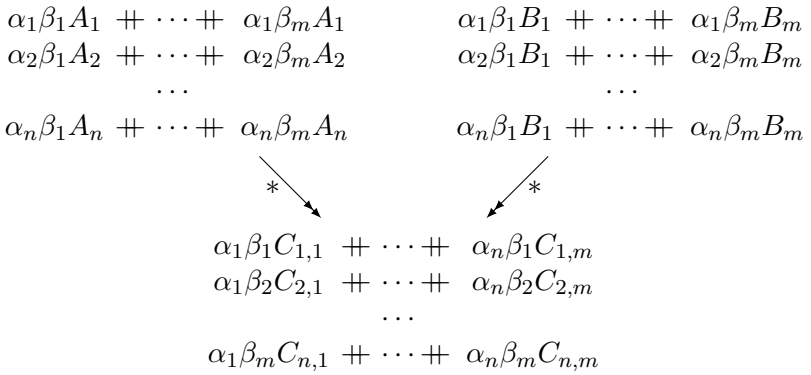
Si tenemos un a-diagrama con Splits entonces, por inversión, sabemos algo sobre la forma de cada rama:



Donde $a \twoheadrightarrow_P^* A_i$ y $a \twoheadrightarrow_P^* B_j$. Los α_i y β_j son arbitrariamente distintos, pero podemos usar Split para equiparar el terreno: dividiremos en la rama izquierda cómo se hizo en la derecha y viceversa.



Ahora, para i, j tenemos $\alpha_i\beta_jA_i$ por la izquierda y $\alpha_i\beta_jB_j$ por la derecha. Por la hipótesis sobre a, debe existir un $C_{i,j}$ tal que $A_i \twoheadrightarrow^* C_{i,j} \twoheadleftarrow^* B_j$. Entonces, por multiplicidad, $\alpha_i\beta_jA_i \twoheadrightarrow^* \alpha_i\beta_jC_{i,j} \twoheadleftarrow^* \alpha_i\beta_jB_j$. Usando composicionalidad podemos confluir en:



cerrando nuestro a-diagrama.

Resumiendo, hemos demostrado el siguiente teorema

Teorema 4.24. *Sea D una distribución. Si, para todo a elemento de D, $E \twoheadleftarrow_P^* [(1,a)] \twoheadrightarrow_P^* F$ implica que existe C tal que $E \twoheadrightarrow^* C \twoheadleftarrow^* F$,*

entonces D es confluente.

4.5. Criterios

Resumiendo los resultados de la sección anterior, vimos que para demostrar que una distribución $D = \left[(p_i, a_i)\right]_i$ es confluente alcanza con cerrar los a_i-diagramas, con peso 1, y sin considerar ninguna equivalencia en la parte superior. Es claro que al demostrarlo para todos los elementos a_i, se sigue la confluencia del sistema entero.

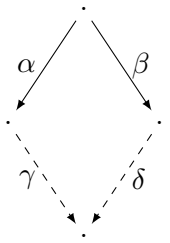
Corolario 4.25. Si siempre que $D \twoheadleftarrow_P^* \left[(1, a)\right] \twoheadrightarrow_P^* E$ existe C tal que $D \twoheadrightarrow^* C \twoheadleftarrow^* E$, entonces $\mathcal{M} \models \mathrm{CR}$.

Pero, ¿cómo acotar la cantidad de evoluciones en la parte superior?

La clave es notar que la sección anterior aplica incluso cambiando $\twoheadrightarrow_P^*$ por $\twoheadrightarrow_P$, en una o ambas ramas superiores e incluso para cerrar el diagrama. En definitiva, las únicas propiedades de $\twoheadrightarrow_P^*$ que usamos fueron la inversión, composicionalidad, y multiplicidad, que $\twoheadrightarrow_P$ también cumple.

Más aún, usamos localidad sólo para las partes superiores, y multiplicidad y composicionalidad sólo para las inferiores. Es decir, nuestra demostración previa es de hecho más fuerte, de la siguiente manera:

Definición 4.26. Decimos que un par de relaciones (γ, δ) *cierra a* otro par (α, β) si $\alpha^{-1} \cdot \beta \subseteq \gamma \cdot \delta^{-1}$. Gráficamente:



Teorema 4.27. *Sean $\alpha, \beta, \gamma, \delta$ relaciones tales que α, β son locales (en el sentido del Lema 4.13) y γ, δ son multiplicativas y composicionales (en el sentido del Lema 4.7 y Lema 4.9).*

Si $(\gamma/\approx, \delta/\approx)$ cierra a (α, β) para distribuciones singleton, entonces $(\gamma/\approx, \delta/\approx)$ cierra a $(S^* \cdot \alpha \cdot (FJ)^*, S^* \cdot \beta \cdot (FJ)^*)$ en cualquier distribución.

Entonces, el Teorema 4.24 es simplemente consecuencia de aplicar este teorema con $\alpha, \beta, \gamma, \delta = P^*$, notando que $S^* \cdot \twoheadrightarrow_P^* \cdot (FJ)^* = (\twoheadrightarrow_P / \approx)^*$ y que $\twoheadrightarrow_P^* / \approx = (\twoheadrightarrow_P / \approx)^* = \twoheadrightarrow^*$.

Sin embargo, tenemos más posibilidades. La relación $\twoheadrightarrow_P$ también tiene todas estas propiedades, y podemos conseguir un criterio para demostrar la propiedad diamante sobre $\twoheadrightarrow_P / \approx$.

Lema 4.28. *Si siempre que* $D_2 \twoheadleftarrow_P \left[(1,a)\right] \twoheadrightarrow_P D_1$ *existe* C *tal que* $D_2 \twoheadrightarrow_{P/\approx} C \twoheadleftarrow_{P/\approx} D_1$, *entonces* $\twoheadrightarrow_{P/\approx} \models \Diamond$.

Demostración. Consecuencia del Teorema 4.27 (con $\alpha, \beta, \gamma, \delta = P$) y de que $(\twoheadrightarrow_P / \approx) = S^* \cdot \twoheadrightarrow_P \cdot (FJ)^*$. ■

Esto puede simplificarse un poco más obviando las reducciones reflexivas de $\twoheadrightarrow_P$ en la parte superior, que siempre pueden cerrarse en un paso (similarmente al Lema 1.28).

Teorema 4.29. *Si siempre que* $D_2 \longmapsto a \longmapsto D_1$ *existe* C *tal que* $D_2 \twoheadrightarrow_{P/\approx} C \twoheadleftarrow_{P/\approx} D_1$, *entonces* $\twoheadrightarrow_{P/\approx} \models \Diamond$.

Demostración. Consecuencia del Lema 4.28. ■

Entonces, es claro que demostrar esta propiedad implica la confluencia del sistema. Este es el criterio que usamos para demostrarla en $\mathcal{Q}^*$ en la Sección 3.6.

Es trivial también obtener un criterio de semiconfluencia.

Lema 4.30. *Si siempre que* $D_2 \twoheadleftarrow_P \left[(1,a)\right] \twoheadrightarrow_P^* D_1$ *existe* C *tal que* $D_2 \twoheadrightarrow_{P/\approx}^* C \twoheadleftarrow_{P/\approx}^* D_1$, *entonces* $\twoheadrightarrow_{P/\approx} \models$ SCR.

Demostración. Consecuencia del Teorema 4.27 (con $\alpha = P$ y $\beta, \gamma, \delta = P^*$). ■

Cambiando de relación

Como vimos en el caso tradicional, a veces es conveniente cambiar la relación por otra mas amena al análisis de confluencia. También será conveniente hacerlo aquí, y damos un criterio simplificado para hacerlo.

Definición 4.31. Decimos que una relación (de un MPARS) $\longmapsto_1$ *simula* a $\longmapsto_2$ cuando siempre que $a \longmapsto_2 D$ se tiene $\left[(1,a)\right] \twoheadrightarrow_1^* D$.

Dada una simulación de ese estilo, tenemos una simulación en las distribuciones.

Lema 4.32. *Si* $\longmapsto_1$ *simula a* $\longmapsto_2$*, entonces* $\twoheadrightarrow_2 \subseteq \twoheadrightarrow_1^*$.

Demostración. Por inducción en $D \twoheadrightarrow_2 E$.

- Evolve

  Tenemos $(p, a) : ds \twoheadrightarrow_2 pA + ds$, dado que $a \longmapsto_2 A$. Por la simulación, tenemos $[(1, a)] \twoheadrightarrow_1^* A$. Por multiplicidad y composicionalidad de $\twoheadrightarrow_1^*$, tenemos $(p, a) : ds \twoheadrightarrow_1^* pA + ds$

- Flip, Join, Split

  La misma regla está presente en $\twoheadrightarrow_1$.

- Tail

  Directo desde la HI. ■

Como consecuencia, tenemos que

Corolario 4.33. Si $\longmapsto_1$ simula a $\longmapsto_2$ y $\longmapsto_2$ simula a $\longmapsto_1$, entonces

$$\longmapsto_1 \models \text{CR} \iff \longmapsto_2 \models \text{CR}$$

Demostración. Consecuencia de que, por el Lema 4.32, $\twoheadrightarrow_1^* = \twoheadrightarrow_2^*$. ■

4.6. Generalización del lema de Newman

En esta sección mostramos una generalización del lema de Newman (Lema 1.25) al contexto multiprobabilista. Dado que la reescritura de distribuciones es un ARS, el lema de Newman podría aplicar directamente a esta situación. Sin embargo, debido a la ausencia de formas normales (salvo la distribución vacía) esto no será posible.

Ni siquiera (en general) si uno fuerza a evolucionar considerando la relación módulo equivalencia $\twoheadrightarrow_P / \approx$, ya que si $a \longmapsto A$ uno puede formar la cadena infinita:

$$[(1, a)] \twoheadrightarrow [(\frac{1}{2}, a)] + \frac{1}{2}A \twoheadrightarrow [(\frac{1}{4}, a)] + \frac{1}{4}A + \frac{1}{2}A \twoheadrightarrow \cdots$$

Entonces buscaremos un lema análogo, con nociones apropiadas de normalización fuerte y confluencia local. Llamaremos *fuertemente normalizante* a una distribución cuando no tenga una evolución infinita, sin permitir equivalencias. Es decir:

Definición 4.34. Una distribución D se llama *fuertemente normalizante*

cuando no existe una cadena infinita de evoluciones propias, es decir, de la forma $D \twoheadrightarrow_E D_1 \twoheadrightarrow_E D_2 \twoheadrightarrow_E \cdots$.

Ciertamente, algunos MPARS cumplen esta condición. El criterio de confluencia local usa la evolución puntual.

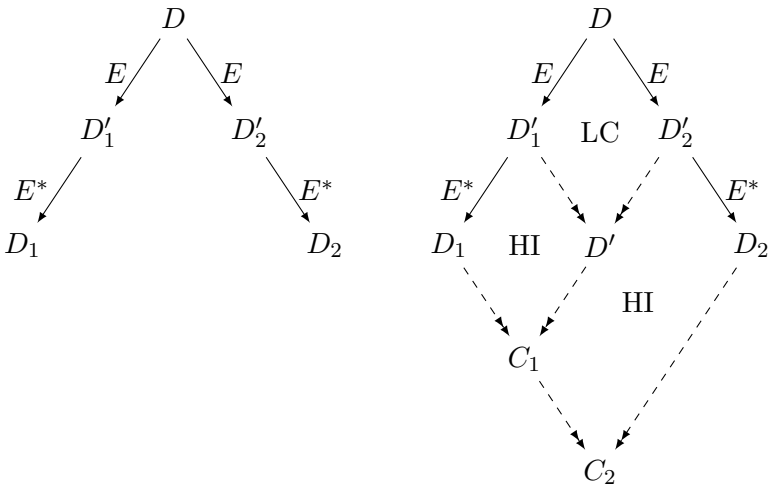
Definición 4.35. Una distribución D se llama *localmente confluente* (notado $\mathrm{LC}(D)$) cuando siempre que $E \twoheadleftarrow_E D \twoheadrightarrow_E F$ existe C tal que $E \twoheadrightarrow^* C \twoheadleftarrow^* F$.

Notamos que una distribución es LC precisamente cuando todos sus singletons lo son (si se expanden puntos distintos, podemos confluir en un paso).

Ahora, podemos extender la prueba de Huet a la reducción de distribuciones de una manera bastante directa.

Teorema 4.36 (Multi-Newman)**.** *Si un MPARS es localmente confluente y fuertemente normalizante, entonces es confluente.*

Demostración. Por inducción bien fundada sobre $\twoheadrightarrow_E$. Por el Teorema 4.24, podemos demostrar la confluencia de D sólo considerando cerrar diagramas de la forma $D_1 \twoheadleftarrow_P^* D \twoheadrightarrow_P^* D_2$[1]. O, equivalentemente, diagramas $D_1 \twoheadleftarrow_E^* D \twoheadrightarrow_E^* D_2$. Si alguna de las dos evoluciones es de cero pasos, el diagrama es trivial. En otro caso, tenemos el diagrama de la izquierda.



Usando la confluencia local, podemos encontrar D'. Usando las hipótesis inductivas para D'_1 y D'_2 encontramos C_1 y C_2, concluyendo la prueba. ■

[1] Notar que esto no es parte de la inducción.

Un detalle a notar es que no es necesaria ninguna hipótesis sobre D', al igual que en la demostración de Huet para el lema tradicional.

De esta forma, para los sistemas fuertemente normalizantes, se simplifica el chequeo de confluencia.

4.7. Confluencia de ARS y PARS

Demostramos que cuando un MPARS $\mathcal{M}$ es inducido por un ARS $\mathcal{A}$ sus confluencias son equivalentes.

Lema 4.37. *Sea $\mathcal{A}$ un ARS y $\mathcal{M}$ tal que $a \longmapsto \big[(1, b)\big]$ si y sólo si $a \to b$ (y no hay otras evoluciones en $\mathcal{M}$). Entonces, $\mathcal{A} \models \mathrm{CR} \iff \mathcal{M} \models \mathrm{CR}$.*

Demostración. **Ida.** Por el Corolario 4.25, basta con cerrar los diagramas $E \twoheadleftarrow_P^* \big[(1, a)\big] \twoheadrightarrow_P^* D$. Dada la definición de $\longmapsto$, debemos tener $D = \big[(1, d)\big]$ con $a \to^* d$ y similarmente para E. Por confluencia de $\mathcal{A}$, existe c tal que $e \to^* c \leftarrow^* d$. Entonces, podemos confluir por $\twoheadrightarrow_P$ en $\big[(1, c)\big]$.

Vuelta. Sea $b \leftarrow^* a \to^* c$. Tenemos entonces $\big[(1, b)\big] \twoheadleftarrow^* \big[(1, a)\big] \twoheadrightarrow^* \big[(1, c)\big]$. Por confluencia, existe D tal que $\big[(1, b)\big] \twoheadrightarrow^* D \twoheadleftarrow^* \big[(1, c)\big]$. Por el Lema 3.15, tenemos que cualquier elemento d_i de D debe ser reducto de b y c. Tomamos cualquiera de ellos y concluimos. ■

Además, podemos mostrar con facilidad que el MPARS generado por un PARS siempre es confluente (y de hecho cumple una propiedad diamante). Esto es consistente con el hecho de que no hay elección en un PARS.

Lema 4.38. *Sea $\mathcal{A} = (A, \to)$ un PARS y $\mathcal{M}$ definido como $a \longmapsto D$ si y sólo si $\to(a) \approx D$, entonces $\mathcal{M} \models \mathrm{CR}$.*

Demostración. Por el Teorema 4.29, basta con cerrar los diagramas $E \longleftarrow a \longmapsto D$ en un paso de $\twoheadrightarrow_{P/\approx}$. Por la definición del MPARS, debemos tener $E \approx \to(a) \approx D$. Dado que $\approx \subseteq \twoheadrightarrow_{P/\approx}$, cerramos en un paso. ■

Capítulo 5

Probabilidades y linealidad

> *"Some of the best things in life are free; and some are not.*
>
> *Truth is free. Having proved a theorem, you may use this proof as many times as you wish, at no extra cost.*
>
> *Food, on the other hand, has a cost. Having baked a cake, you may eat it only once.*
>
> *If traditional logic is about truth, then linear logic is about food."*
>
> —Philip Wadler, 1993

En este capítulo analizamos cuáles son las razones por las cuales un lenguaje de programación probabilista puede ser no confluente, obteniendo intuiciones interesantes y relacionándolas al λ-cálculo clásico.

Con esa intuición, definimos un λ-cálculo probabilista que evita las causas de no confluencia mediante restricciones de *linealidad*. Damos una prueba simple de su confluencia, la cual debería ser extensible al agregar nuevas funcionalidades al cálculo.

5.1. El porqué de la no confluencia

Supongamos que tenemos una extensión probabilista al λ-cálculo, donde podemos formar un término $\bigstar$ que reduce con igual probabilidad a dos formas normales distintas.

$$\bigstar \longmapsto \left[(\frac{1}{2}, n_0), (\frac{1}{2}, n_1)\right]$$

Supongamos además que hay algún no determinismo en la reducción (o la confluencia sería trivialmente cierta). En particular pensemos que tenemos congruencias para la aplicación. Si formamos el término $\Theta = (\lambda x.xx)\bigstar$, vemos que llegamos a un problema.

$$\left[(1, \Theta)\right] \twoheadrightarrow \left[(1, \bigstar\bigstar)\right] \twoheadrightarrow^* \left[(\tfrac{1}{4}, n_0n_0), (\tfrac{1}{4}, n_0n_1), (\tfrac{1}{4}, n_1n_0), (\tfrac{1}{4}, n_1n_1)\right]$$
$$\left[(1, \Theta)\right] \twoheadrightarrow \left[(\tfrac{1}{2}, (\lambda x.xx)n_0), (\tfrac{1}{2}, (\lambda x.xx)n_1)\right] \twoheadrightarrow^* \left[(\tfrac{1}{2}, n_0n_0), (\tfrac{1}{2}, n_1n_1)\right]$$

Estás distribuciones son terminales y no equivalentes, y por lo tanto el sistema no es confluente. Entonces, si queremos permitir congruencias arbitrarias, necesitamos una forma de prohibir estos comportamientos.

Notamos que es la interacción entre reducir el argumento y evaluar la función que da el problema, al igual que vimos en la Sección 1.3. En ese caso, el problema era causado por las copias del argumento, que aumentaban la cantidad de pasos a realizar.

También es el caso aquí que las copias son el problema (si x aparece libre una sola vez, el problema no ocurre), pero la diferencia no esta sólo en la cantidad de pasos, sino que existe una diferencia cuantitativa en el comportamiento de los términos, como se observa en las distribuciones finales. Esta diferencia es causada sólo por el comportamiento probabilista de $\bigstar$, ya que el λ-cálculo clásico mantiene la confluencia aun en este caso.

Concluimos que, en el contexto de un lenguaje funcional, podemos lograr un fallo de confluencia con tres ingredientes:

1. Poder, a la vez, reducir un argumento o aplicar la función.
2. Que las funciones puedan duplicar su argumento.
3. Que el argumento tenga comportamiento probabilista.

5.2. λ_1: un cálculo lineal

Teniendo en mente a la sección previa, diseñaremos un cálculo que evite esta mezcla explosiva de tres ingredientes. Demostraremos que el cálculo es confluente, dando evidencia de que sólo de esa manera puede romperse la confluencia en un lenguaje probabilista razonable.

En este cálculo, llamado λ_1, tendremos dos tipos de abstracciones: *lineales* y *no lineales*. Las abstracciones lineales no podrán duplicar sus argumentos, pero los mismos pueden ser probabilistas y reducir antes de aplicarse. Las no lineales sí pueden duplicar, pero sus argumentos (posiblemente probabilistas) estarán suspendidos como *thunks* y no reducirán antes de aplicarse[1].

El uso de linealidad es común en los cálculos cuánticos. Algunos ejemplos son (van Tonder 2004; Selinger y Valiron 2005; Dal Lago, Masini y Zorzi 2011; Atzemoglou 2014) y (Díaz-Caro y Dowek 2016). En esos casos, la justificación para la linealidad no es sólo la confluencia sino el hecho de que es imposible duplicar un qubit arbitrario (por el teorema de no clonado descripto en la Sección 2.4).

El cálculo λ_1 está basado en el λ-cálculo con reducción superficial de Alex Simpson (Simpson 2005), pero relajamos la linealidad a *afinidad*, es decir, permitimos que una función no use su argumento.

[1] Notamos que una tercera opción de función (que duplique y reduzca argumentos, forzando que no sean probabilistas) debería ser posible de agregar manteniendo confluencia, pero no hacemos el desarrollo.

Sintaxis y buena formación

El conjunto de *pretérminos* está dado por las expresiones de la gramática:

$$M, N ::= x \mid MN \mid \lambda x.M \mid \lambda!x.M \mid !M \mid M \oplus_p N$$

Aquí, $\lambda x.M$ representa una *abstracción lineal*, que no puede duplicar su argumento. Análogamente, $\lambda!x.M$ representa a las no lineales. Las abstracciones no lineales sólo pueden evaluar con argumentos de la forma $!M$. El símbolo $!$ suspende a su argumento como un "thunk", y no permite reducciones internas.

Como novedad, el término $M \oplus_p N$ representa una elección (*choice*) probabilista entre M y N, con probabilidades p y $(1-p)$ respectivamente.

Para asegurarnos de que las funciones lineales no duplican a sus argumentos, sólo consideraremos pretérminos que cumplan condiciones de buena formación. Damos el juicio de manera inductiva en la Figura 5.1. En esas reglas, Γ y Δ representan conjuntos de variables libres. Se asume que la unión sólo está definida para conjuntos disjuntos, y que Γ y Δ siempre son disjuntos. También identificamos los términos α-equivalentes. El juicio fuerza que las variables de abstracciones lineales aparezcan una única vez en su cuerpo.

$$\frac{x \in \Gamma}{\Gamma \mid \Delta \vdash x}\ \text{WF-Var} \qquad \frac{x \in \Delta}{\Gamma \mid \Delta \vdash x}\ \text{WF-Var!}$$

$$\frac{\Gamma_1 \mid \Delta \vdash M \quad \Gamma_2 \mid \Delta \vdash N}{\Gamma_1 \cup \Gamma_2 \mid \Delta \vdash MN}\ \text{WF-App} \qquad \frac{\Gamma_1 \mid \Delta \vdash M \quad \Gamma_2 \mid \Delta \vdash N}{\Gamma_1 \cup \Gamma_2 \mid \Delta \vdash M \oplus_p N}\ \text{WF-}\oplus$$

$$\frac{\Gamma, x \mid \Delta \vdash M}{\Gamma \mid \Delta \vdash \lambda x.M}\ \text{WF-Abs} \qquad \frac{\Gamma \mid \Delta, x \vdash M}{\Gamma \mid \Delta \vdash \lambda!x.M}\ \text{WF-Abs!}$$

$$\frac{\cdot \mid \Delta \vdash M}{\Gamma \mid \Delta \vdash !M}\ \text{WF-Bang}$$

Figura 5.1: Juicio de buena formación para λ_1

La posibilidad de no usar variables se ve en el uso de Γ en WF-Var, WF-Var! y WF-Bang.

De ahora en más, sólo consideramos pretérminos bien formados (aquellos M tales que $\Gamma \mid \Delta \vdash M$ para algún Γ y Δ) y los llamamos *términos*.

Cuando $\Gamma \mid \Delta \vdash M$ y $x \in \Gamma$, tenemos que x aparece libre a lo sumo una vez en M. Cuando aparezca exactamente una vez diremos que x es *lineal* en M.

Semántica operacional

La semántica operacional está dada por un MPARS. Fijaremos una semántica laxa, y luego demostraremos la confluencia de la misma.

La evaluación de abstracciones tiene su semántica usual, notando que ahora tenemos dos tipos de las mismas, cada una con su regla[2] .

$$\frac{}{(\lambda x.M)N \longmapsto \left[(1, M[N/x])\right]}\ \text{R-}\beta \qquad \frac{}{(\lambda !x.M)!N \longmapsto \left[(1, M[N/x])\right]}\ \text{R-}\beta!$$

La regla que introduce el comportamiento probabilista es claramente la reducción de un choice. Además, tenemos reglas de congruencia de los mismos.

$$\frac{}{M \oplus_p N \longmapsto \left[(p, M), (1-p, N)\right]}\ \text{R-}\oplus$$

$$\frac{M \longmapsto \left[(p_i, M_i)\right]_i}{M \oplus_p N \longmapsto \left[(p_i, M_i \oplus_p N)\right]_i}\ \text{R-}\oplus\text{-L} \qquad \frac{N \longmapsto \left[(p_i, N_i)\right]_i}{M \oplus_p N \longmapsto \left[(p_i, M \oplus_p N_i)\right]_i}\ \text{R-}\oplus\text{-R}$$

Como se espera, tenemos congruencias a izquierda y derecha de la aplicación, y una reducción fuerte dentro de los cuerpos de abstracciones.

$$\frac{M \longmapsto \left[(p_i, M_i)\right]_i}{MN \longmapsto \left[(p_i, M_i N)\right]_i}\ \text{R-AppL} \qquad \frac{N \longmapsto \left[(p_i, N_i)\right]_i}{MN \longmapsto \left[(p_i, MN_i)\right]_i}\ \text{R-AppR}$$

$$\frac{M \longmapsto \left[(p_i, M_i)\right]}{\lambda x.M \longmapsto \left[(p_i, \lambda x.M_i)\right]_i}\ \text{R-}\lambda \qquad \frac{M \longmapsto \left[(p_i, M_i)\right]}{\lambda !x.M \longmapsto \left[(p_i, \lambda !x.M_i)\right]_i}\ \text{R-}\lambda!$$

No hay regla que permita reducir bajo un término de la forma $!M$, con lo cual dichos términos están suspendidos. En efecto, esto implica que las abstracciones no lineales siguen aproximadamente una estrategia *call-by-name (CBN)*. Usar *call-by-value (CBV)* sería válido también: sólo debemos impedir la interacción entre reducir el argumento y evaluar la función.

Ejemplo 5.1. Sea $\Theta' = (\lambda !x.xx)!(a \oplus_{\frac{1}{2}} b)$ con a, b formas normales distintas. La única reducción posible es aplicar R-β! resultando

$$\Theta' \longmapsto \left[(1, (a \oplus_{\frac{1}{2}} b)(a \oplus_{\frac{1}{2}} b))\right]$$

Como se espera, la semántica es consistente con el juicio de buena formación.

[2] La sustitución tiene su definición usual, con λ y $\lambda!$ ligando variables y definida por componentes para $\oplus_p$.

Lema 5.2. *Si* $\Gamma \mid \Delta \vdash M$ *y* $M \longmapsto D = \left[(p_i, N_i)\right]_i$, *entonces* $\Gamma \mid \Delta \vdash N_i$ *para cada* i.

Demostración. Por inducción en $M \longmapsto D$ (los detalles se encuentran en el Apéndice B.2). ■

Analizando la reducción

Para analizar la confluencia, extenderemos algunos de los lemas clásicos sobre reducciones y sustitución. Primero, introduciremos alguna notación para hablar sobre sustituciones *de* y *con* distribuciones en lugar de términos.

Si $D = \left[(p_i, M_i)\right]_i$ es una distribución y N un término, notamos con $D[N/x]$ a la distribución $\left[(p_i, M_i[N/x])\right]_i$. Dualmente, notamos con $N[D/x]$ a la distribución $\left[(p_i, N[M_i/x])\right]_i$. También notamos con DN a la distribución $\left[(p_i, M_iN)\right]_i$, y análogamente para ND. Notar que entonces R-AppL puede expresarse como $M \longmapsto D \implies MN \longmapsto DN$.

Demostraremos dos lemas de sustitución para la reducción en λ_1. El primero es análogo al que teníamos para el λ-cálculo (Lema 1.30). El segundo es consecuencia de las restricciones de linealidad.

Lema 5.3. *Si* $M \longmapsto D$, *entonces* $M[N/x] \longmapsto D[N/x]$.

Demostración. Por inducción en $M \longmapsto D$.

- R-β y R-β!.

  Por la premisa $(\lambda y.P)Q \longmapsto \left[(1, P[Q/y])\right]$. Podemos hacer $(\lambda y.P[N/x])(Q[N/x]) \to \left[(1, P[N/x]\,[Q[N/x]/y])\right]$. Por propiedades de la sustitución, $P[N/x]\,[Q[N/x]/y] = P[Q/y][N/x]$, dando el resultado.

- R-AppL.

  Tenemos $M = PQ$ y por la premisa $P \longmapsto \left[(p_i, P_i)\right]_i$. Por la HI, tenemos $P[N/x] \longmapsto \left[(p_i, P_i[N/x])\right]_i$. Aplicando R-AppL tenemos $P[N/x]Q[N/x] \longmapsto \left[(p_i, P_i[N/x]Q[N/x])\right]_i$ como se buscaba.

- R-AppR. Análogo a R-AppL.

- R-λ y R-λ!. Directo por HI.

- R-$\oplus$.

  Tenemos $M = P \oplus_p Q$ con $M \longmapsto \left[(p, P), (1-p, Q)\right]$. Por la misma regla $P[N/x] \oplus_p Q[N/x] \longmapsto \left[(p, P[N/x]), (1-p, Q[N/x])\right]$ como se buscaba.

- R-$\oplus$-L y R-$\oplus$-R. Similares a R-AppL y R-AppR. ■

Lema 5.4. *Si $M \longmapsto D$, y x es lineal en N, entonces $N[M/x] \longmapsto N[D/x]$.*

Demostración. Por inducción en la buena formación de N. Sea $D = \left[(p_i, D_i)\right]_i$.

- WF-Var.

  Tenemos $N = x$ y concluimos trivialmente. Notar que no puede ser otra variable, ya que x debe ser lineal en N.

- WF-App.

  Tenemos $N = N_l N_r$. En ese caso, uno y sólo uno de N_l, N_r tiene a x libre. Supongamos que es N_l, entonces por la HI tenemos que $N_l[M/x] \longmapsto \left[(p_i, N_l[D_i/x])\right]_i$. Aplicando R-AppL tenemos $N_l[M/x]\ N_r \longmapsto \left[(p_i, N_l[D_i/x]\ N_r)\right]_i$. Dado que x no está libre en N_r, esto es nuestra meta. El razonamiento es análogo para N_r.

- WF-$\oplus$. Análogo al caso de la aplicación.

- WF-λ, WF-λ!. Directo por HI y R-λ o R-λ!.

- WF-Bang, WF-Var!. No puede ser que x esté libre. ■

Estos dos resultados son análogos al Lema 3.1 de (Simpson 2005), pero permitiendo comportamiento probabilista.

5.3. Confluencia de λ_1

Demostraremos, usando ambos lemas previos, que el cálculo cumple una propiedad diamante de la siguiente forma.

Teorema 5.5 (Prop. diamante para λ_1)**.** *Si $E \longleftarrow M \longmapsto D$ entonces existen C, C' tales que $D \twoheadrightarrow_P C$ y $E \twoheadrightarrow_P C'$ con $C \approx C'$.*

Demostración. Por inducción en la forma de ambas reducciones. Al ser $\twoheadrightarrow_P$ reflexiva, si $D \approx E$ podemos concluir inmediatamente.

1. $M = PQ$; R-AppL; R-AppL.

   Por inversión, tenemos $D_1 \longleftarrow P \longmapsto D_2$. Entonces, por la hipótesis inductiva, existen $\left[(p_i, C_i)\right]_i \approx \left[(p_i, C'_i)\right]_i$ tales que

   $$D_1 \twoheadrightarrow_P \left[(p_i, C_i)\right]_i \qquad D_2 \twoheadrightarrow_P \left[(p_i, C'_i)\right]_i$$

   Usando R-AppL, podemos hacer

   $$D_1 Q \twoheadrightarrow_P \left[(p_i, C_i Q)\right]_i \qquad D_2 Q \twoheadrightarrow_P \left[(p_i, C'_i Q)\right]_i$$

   Que, claramente, son equivalentes (por el Corolario 4.15).

2. $M = PQ$; R-AppR; R-AppR.

   Análogo al caso anterior.

3. $M = PQ$; R-AppL; R-AppR.

   Tenemos $M \longmapsto \left[(p_i, P_iQ)\right]_i$ y $M \longmapsto \left[(q_j, PQ_j)\right]_j$. Por las premisas obtenemos que $P \longmapsto \left[(p_i, P_i)\right]_i$ y $Q \longmapsto \left[(q_j, Q_j)\right]_j$. Podemos aplicar las congruencias para evolucionar de nuevo a cada lado y obtener:

   $$\left[(q_j, PQ_j)\right]_j \twoheadrightarrow_P \left[(p_iq_j, P_iQ_j)\right]_{j,i} \qquad \left[(p_i, P_iQ)\right]_i \twoheadrightarrow_P \left[(q_jp_i, P_iQ_j)\right]_{i,j}$$

   Como la multiplicación conmuta, estas dos distribuciones son permutaciones una de la otra y por lo tanto equivalentes.

4. $M = (\lambda x.P)Q$; R-AppL; R-β.

   Por R-AppL se reduce la abstracción, que sólo puede reducir vía R-λ. Entonces, tenemos

   $$M \longmapsto \left[(p_i, (\lambda x.P_i)Q)\right]_i \qquad M \longmapsto \left[(1, P[Q/x])\right]$$

   donde $P \longmapsto \left[(p_i, P_i)\right]_i$. Usando la regla R-$\beta$ y el Lema 5.3 respectivamente, obtenemos

   $$\begin{array}{rcl} \left[(p_i, (\lambda x.P_i)Q)\right]_i & \twoheadrightarrow_P & \left[(p_i, P_i[Q/x])\right]_i \\ \left[(1, P[Q/x])\right] & \twoheadrightarrow_P & \left[(p_i, P_i[Q/x])\right]_i \end{array}$$

   que son exactamente iguales.

5. $M = (\lambda x.P)Q$; R-AppR; R-β.

   Tenemos

   $$M \longmapsto \left[(p_i, (\lambda x.P)Q_i)\right]_i \qquad M \longmapsto \left[(1, P[Q/x])\right]$$

   Por la buena formación de $(\lambda x.P)$, sabemos que o bien x es lineal o bien no está libre en P.

   Si es lineal, usando R-β y el Lema 5.4 tenemos

   $$\begin{array}{rcl} \left[(p_i, (\lambda x.P)Q_i)\right]_i & \twoheadrightarrow_P & \left[(p_i, P[Q_i/x])\right]_i \\ \left[(1, P[Q/x])\right] & \twoheadrightarrow_P & \left[(p_i, P[Q_i/x])\right]_i \end{array}$$

   que son exactamente iguales.

   Si no aparece libre, tenemos $P[Q/x] = P$, y entonces usamos R-β y reflexividad, respectivamente, para obtener:

   $$\begin{array}{rcl} \left[(p_i, (\lambda x.P)Q_i)\right]_i & \twoheadrightarrow_P & \left[(p_i, P)\right]_i \\ \left[(1, P)\right] & \twoheadrightarrow_P & \left[(1, P)\right] \end{array}$$

   que son equivalentes al tener $\sum p_i = 1$.

6. $M = (\lambda x.P)Q$; R-β; R-β.

   Ambas distribuciones son exactamente iguales

7. $M = P \oplus_p Q$; R-$\oplus$; R-$\oplus$.

   Ambas distribuciones son exactamente iguales

8. $M = P \oplus_p Q$; R-$\oplus$; R-$\oplus$-L.

   Tenemos

   $$M \longmapsto \left[(p, P), (1-p, Q)\right] \qquad M \longmapsto \left[(p_i, P_i \oplus_p Q)\right]_i$$

   donde $P \longmapsto \left[(p_i, P_i)\right]_i$. A la izquierda, usamos $\twoheadrightarrow_P$ para reducir P y no Q. A la derecha, aplicamos R-$\oplus$-R a cada término. Obtenemos:

   $$\begin{array}{rcl} \left[(p, P), (1-p, Q)\right] & \twoheadrightarrow_P & \left[(p\ p_i, P_i)\right]_i + \left[(1-p, Q)\right] \\ \left[(p_i, P_i \oplus_p Q)\right]_i & \twoheadrightarrow_P & \left[(p_i\ p, P_i), (p_i\ (1-p), Q_i)\right]_i \end{array}$$

   Dado que la multiplicación conmuta y que $\sum p_i = 1$, estas distribuciones son equivalentes.

9. $M = P \oplus_p Q$; R-$\oplus$; R-$\oplus$-R.

   Análogo al caso anterior.

10. $M = P \oplus_p Q$; R-$\oplus$-L; R-$\oplus$-L.

    Directo por HI y congruencias (muy similar al caso de la aplicación)

11. $M = P \oplus_p Q$; R-$\oplus$-R; R-$\oplus$-R.

    Directo por HI y congruencias (muy similar al caso de la aplicación)

12. $M = P \oplus_p Q$; R-$\oplus$-L; R-$\oplus$-R.

    Directo por HI y congruencias (muy similar al caso de la aplicación)

13. $M = \lambda!x.P$; R-$\lambda!$; R-$\lambda!$.

    Ambas distribuciones son exactamente iguales

14. $M = PQ$; R-AppL; R-$\beta!$.

    Análogo al caso para R-AppL y R-β (la linealidad no es relevante aquí).

15. $M = PQ$; R-AppR; R-$\beta!$.

    Este caso no puede ocurrir. Si aplicamos R-$\beta!$, el lado derecho es un thunk $!Q$ y no puede reducir. ■

Con este resultado, y un criterio previo (Teorema 4.29), se obtiene de forma directa la confluencia de distribuciones del cálculo.

Corolario 5.6. $\lambda_1 \models \mathrm{CR}$.

Demostración. Por por el Teorema 5.5 y Teorema 4.29. ■

Capítulo 6

Generalidad

"Specialization is for insects."

—Robert A. Heinlein

Hemos discutido que la reescritura de distribuciones de un MPARS es, al menos, una buena herramienta para demostrar unicidad de distribuciones terminales. Por sobre ello, pensaremos en el significado propio de la reescritura de distribuciones.

Fijado un MPARS $\mathcal{A}$, ¿tiene sentido la evolución de distribuciones inducida? Es decir: ¿realmente tiene una interpretación como una evolución probabilista?

En esta sección postulamos algunos axiomas que deberían ser ciertos para cualquier relación entre distribuciones (sea inducida por un MPARS o no) que realmente represente la evolución probabilista de un sistema, dando argumentos intuitivos que los soportan.

Mostramos que la evolución de distribuciones inducida siempre cumple estos axiomas. Más aún, mostramos que cualquier evolución (sobre distribuciones de soporte finito) que los cumpla puede ser modelada como la determinización de un MPARS, dando apoyo a nuestra definición.

6.1. Distribuciones y evolución

Estudiaremos las evoluciones de distribuciones de manera genérica. Primero, volvemos a una definición matemática de las mismas, dándoles otro nombre para diferenciarlas de las listas en $\mathscr{D}(A)$.

Definición 6.1. Una *distribución matemática* sobre un conjunto (finito o infinito numerable) A es una función D de tipo $A \to \mathbb{R}^+$. Cada distribución tiene un *peso* no negativo asociado (notado con $w_{\mathbb{D}}$ para diferenciarlo

del peso sobre $\mathscr{D}(A)$), dado por

$$w_{\mathbb{D}}(D) = \sum_{a \in A} D(a)$$

Cuando el peso de una distribución sea exactamente 1, la llamamos *propia*. El conjunto de todas las distribuciones sobre A se nota con $\mathbb{D}(A)$.

Observación 6.2. Se podría rechazar el uso de distribuciones con pesos arbitrarios y usar sólo aquellas propias. El desarrollo de este capítulo aplica de la misma manera, sólo siendo algo menos directo.

Notamos con $D + E$ y αD (con $\alpha \in \mathbb{R}^+$) a la suma y escalamiento de distribuciones, respectivamente, con la definición obvia. La suma es asociativa y conmutativa, con elemento neutro 0_D, la distribución nula. Además, la suma es lineal sobre el producto escalar, es decir $\alpha D + \beta D = (\alpha + \beta)D$. Usamos la notación $\overset{\circ}{a}$ para la distribución D con $D(a) = 1$ y $D(x) = 0$ para todo $x \neq a$.

Para obtener una distribución matemática a partir de una distribución de lista, usamos la función $[\![-]\!] : \mathscr{D}_1(A) \to \mathbb{D}(A)$ definida por las ecuaciones:

$$\begin{array}{rcl} [\![\,[\,]\,]\!] & = & 0_D \\ [\![(p,a) : ds]\!] & = & p\overset{\circ}{a} + [\![ds]\!] \end{array}$$

Esta definición es consistente con nuestra noción de equivalencia, como lo establece siguiente lema.

Lema 6.3. $D \approx E \iff [\![D]\!] = [\![E]\!]$

Demostración. **Ida.** Si $D \sim E$, es claro que $[\![D]\!] = [\![E]\!]$, por conmutatividad y linealidad de la suma. Por inducción en la cantidad de pasos se sigue el resultado.

Vuelta. Sean $a_1, \ldots, a_n$ los elementos de A tales que $[\![D]\!](a_i) \neq 0$. (Como la lista D es finita, el conjunto es finito). Para cada a_i, la suma total de sus pesos es $[\![D]\!](a_i)$. Entonces, haciendo Flips y Joins podemos llevar a D a la distribución $\left[([\![D]\!](a_1), a_1), \ldots, ([\![D]\!](a_n), a_n)\right]$ y análogamente para E. Como $[\![D]\!] = [\![E]\!]$, estas distribuciones son iguales y tenemos $D \approx E$. ■

Además, la concatenación, escalamiento y peso de distribuciones también son consistentes.

Lema 6.4. $[\![D +\!\!+ E]\!] = [\![D]\!] + [\![E]\!]$

Demostración. Por inducción en D.

- $D = []$. Trivial.
- $D = (p, a) : ds$. Por la HI tenemos $[\![ds +\!\!+ E]\!] = [\![ds]\!] + [\![E]\!]$. Entonces,

$$
\begin{aligned}
& [\![((p, a) : ds) +\!\!+ E]\!] \\
= & \{ \ \textit{Def.} +\!\!+ \ \} \\
& [\![(p, a) : ds +\!\!+ E]\!] \\
= & \{ \ \textit{Def.} [\![-]\!] \ \} \\
& p\overset{\circ}{a} + [\![ds +\!\!+ E]\!] \\
= & \{ \ \textit{HI} \ \} \\
& p\overset{\circ}{a} + [\![ds]\!] + [\![E]\!] \\
= & \{ \ \textit{Def.} [\![-]\!] \ \} \\
& [\![(p, a) : ds]\!] + [\![E]\!]
\end{aligned}
$$

■

Lema 6.5. $[\![\alpha D]\!] = \alpha\, [\![D]\!]$

Demostración. Por inducción en D.

- $D = []$. Trivial.
- $D = (p, a) : ds$. Por la HI tenemos $[\![\alpha ds]\!] = \alpha\, [\![ds]\!]$. Entonces,

$$
\begin{aligned}
& [\![\alpha((p, a) : ds)]\!] \\
= & \{ \ \textit{Def. escalamiento} \ \} \\
& [\![(\alpha p, a) : \alpha ds]\!] \\
= & \{ \ \textit{Def.} [\![-]\!] \ \} \\
& \alpha p\overset{\circ}{a} + [\![\alpha ds]\!] \\
= & \{ \ \textit{HI} \ \} \\
& \alpha p\overset{\circ}{a} + \alpha\, [\![ds]\!] \\
= & \{ \ \textit{Distributividad} \ \} \\
& \alpha(p\overset{\circ}{a} + [\![ds]\!]) \\
= & \{ \ \textit{Def.} [\![-]\!] \ \} \\
& \alpha([\![(p, a) : ds]\!])
\end{aligned}
$$

■

Lema 6.6. $w(D) = w_{\mathbb{D}}([\![D]\!])$

Demostración. Por inducción en D.

- $D = [\,]$. Trivial.
- $D = (p, a) : ds$. Por la HI tenemos $w(ds) = w_{\mathbb{D}}([\![ds]\!])$ Entonces,

$$
\begin{aligned}
& w_{\mathbb{D}}([\![(p,a) : ds]\!] \\
= & \ \{\ \textit{Def. } [\![-]\!]\ \} \\
& w_{\mathbb{D}}(p\overset{\circ}{a} + [\![ds]\!]) \\
= & \ \{\ \textit{Def. } w_{\mathbb{D}}(-)\ \} \\
& \textstyle\sum_{b\in A}(p\overset{\circ}{a} + [\![ds]\!])(b) \\
= & \ \{\ \textit{Def. suma de distribuciones}\ \} \\
& \textstyle\sum_{b\in A} p\overset{\circ}{a}(b) + [\![ds]\!]\,(b) \\
= & \ \{\ \textstyle\sum a + b = \sum a + \sum b\ \} \\
& \textstyle\sum_{b\in A} p\overset{\circ}{a}(b) + \sum_{b\in A} [\![ds]\!]\,(b) \\
= & \ \{\ \overset{\circ}{a}(b) = 1 \iff a = b,\ 0\ \textit{en otro caso}\ \} \\
& p + \textstyle\sum_{b\in A} [\![ds]\!]\,(b) \\
= & \ \{\ \textit{Def. peso}\ \} \\
& p + w_{\mathbb{D}}([\![ds]\!]) \\
= & \ \{\ \textit{HI}\ \} \\
& p + w(ds) \\
= & \ \{\ \textit{Def. } w(-)\ \} \\
& w((p,a) : ds)
\end{aligned}
$$

■

6.2. Axiomas de composicionalidad

Manteniendo la generalidad, cualquier evolución posible entre distribuciones debe ser una relación (sin probabilidades asociadas), formando un ARS $(\mathbb{D}(A), \to)$. Sin embargo, el concepto de evolución probabilista debe cumplir algunas reglas.

Primero, para tener significado probabilista, la evolución debería mantener el peso de las distribuciones. Además, la evolución debería ser *composicional* respecto de la suma: si D está compuesta por otras distribuciones, los comportamientos de estas deberían seguir existiendo en D.

Formalizamos esto con el siguiente axioma:

$$\frac{D \to^* D' \quad E \to^* E'}{\alpha D + \beta E \to^* \alpha D' + \beta E'}\ \textsc{AxComp}$$

En general no nos interesará la cantidad de evoluciones, por lo que usualmente trabajaremos con $\to^*$. Notar que AxComp permite una variación arbitraria en la cantidad de pasos.

La composicionalidad permite razonar de una manera más simple sobre las evoluciones. Notar que los árboles de computación, si usamos sus soportes como la distribución que inducen, no cumplen este axioma. En efecto, el árbol $[a, (\frac{1}{2}, b), (\frac{1}{2}, b)]$ tiene más comportamientos que $[a, (1, b)]$.

La composicionalidad y el principio de que distribuciones iguales tengan los mismos comportamientos son naturales. Sin embargo, con ambos principios, se obtiene la evolución parcial que se discutió antes (Sección 3.3): supongamos que $D \to^* E$ y sea $\alpha \in (0, 1)$, luego podemos concluir que:

$$D = \frac{\overline{D \to^* D} \quad \overline{D \to^* E}}{(1-\alpha)D + \alpha D \to^* (1-\alpha)D + \alpha E}\ \textsc{AxComp}$$

Entonces, D puede evolucionar parcialmente. Dado que no queremos sacrificar la composicionalidad, aceptaremos la evolución parcial.

El axioma previo indica que los comportamientos de las partes deben estar presentes en el todo. Sin embargo, ¿puede un todo presentar comportamientos no presentes en sus partes?

Postulamos que no[1]: esto sólo sería posible si hay alguna interacción o interferencia entre ellas, pero una distribución de probabilidad representa un conjunto de posibles estados, de los cuales a lo sumo uno puede estar materializado, impidiendo interacciones.

Concretamente, si la distribución $\frac{1}{2}D + \frac{1}{2}E$ evoluciona, esta evolución debe poder ser modelada por una evolución de D, de E, o de ambos. No tiene sentido que la combinación de D y E sea necesaria para la reducción: una evolución es composicional y sólo composicional.

Descartamos las evoluciones de ese estilo incluyendo el siguiente axioma (con D', E' cuantificados existencialmente):

$$\frac{\alpha D + \beta E \to^* O}{O = \alpha D' + \beta E' \ \text{ con } \ D \to^* D' \ \text{ y } \ E \to^* E'}\ \textsc{AxInv}$$

Notar que tiene un espíritu inverso al axioma AxComp. Cuando una relación entre distribuciones cumple ambos axiomas, la llamamos *sound*.

6.3. Soundness

Dado un MPARS arbitrario, veremos que la evolución inducida por él es siempre sound. Usaremos la función $[\![-]\!]$ para obtener una reducción entre distribuciones matemáticas. Nuestra reducción entre distribuciones $\to$ está definida por:

$$[\![D]\!] \to [\![E]\!] \iff D \twoheadrightarrow^* E$$

[1] Con disculpas a Aristóteles.

Esta definición es correcta debido al Lema 6.3 y el hecho de $\twoheadrightarrow^*$ es compatible con $\approx$. En efecto, si $[\![D]\!] = [\![D']\!]$ y $[\![E]\!] = [\![E']\!]$ tenemos:

$$\dfrac{\dfrac{[\![D']\!] = [\![D]\!]}{\dfrac{D' \approx D}{D' \twoheadrightarrow^* D}} \quad D \twoheadrightarrow^* E \quad \dfrac{\dfrac{[\![E]\!] = [\![E']\!]}{E \approx E'}}{E \twoheadrightarrow^* E'}}{\dfrac{D' \twoheadrightarrow^* E'}{[\![D']\!] \to [\![E']\!]}}$$

Las operaciones de suma y escalamiento serán la concatenación y escalamiento de listas.

$$\begin{aligned} [\![D]\!] + [\![E]\!] &= [\![D +\!\!+ E]\!] \\ \alpha\,[\![D]\!] &= [\![\alpha D]\!] \end{aligned}$$

Es fácil ver que las definiciones son correctas y que cumplen los requerimientos listados anteriormente. Dado esto, podemos demostrar que nuestra relación es composicional.

Lema 6.7. *La relación $\to$ cumple* AxComp.

Demostración. Supongamos que $[\![D]\!] \to^* [\![D']\!]$ y $[\![E]\!] \to^* [\![E']\!]$. Por definición, tenemos $D \twoheadrightarrow^* D'$ y $E \to^* E'$. Por multiplicidad y composicionalidad, tenemos que $\alpha D +\!\!+ \beta E \twoheadrightarrow^* \alpha D' +\!\!+ \beta E'$, y entonces $[\![\alpha D +\!\!+ \beta E]\!] \to^* [\![\alpha D' +\!\!+ \beta E']\!]$. Entonces, llegamos a nuestra meta de $\alpha\,[\![D]\!] + \beta\,[\![E]\!] \to^* \alpha\,[\![D']\!] + \beta\,[\![E']\!]$. ■

Lema 6.8. *La relación $\to$ cumple* AxInv.

Demostración. Supongamos que $\alpha\,[\![D]\!] + \beta\,[\![E]\!] = [\![\alpha D +\!\!+ \beta E]\!] \to^* [\![O]\!]$. Debemos tener $\alpha D +\!\!+ \beta E \twoheadrightarrow^* O$. Por la descomposición de $\twoheadrightarrow$ (Teorema 4.21), debe existir un $O' \approx O$ tal que $\alpha D +\!\!+ \beta E \twoheadrightarrow^*_{SP} O'$. Por localidad (Lema 4.13) tenemos que $O' = \alpha D' +\!\!+ \beta E'$ con $D \twoheadrightarrow^*_{SP} D'$ y $E \twoheadrightarrow^*_{SP} E'$. Esto implica que $[\![O]\!] = [\![O']\!] = \alpha\,[\![D']\!] + \beta\,[\![E']\!]$ con $[\![D]\!] \to^* [\![D']\!]$ y $[\![E]\!] \to^* [\![E']\!]$, como buscábamos. ■

Entonces, cualquier reescritura de distribuciones generada por un MPARS es sound.

Teorema 6.9. *La reescritura $\to$ propuesta es sound.*

Demostración. Por los dos lemas previos. ■

6.4. Completitud

Hemos visto que toda evolución inducida por un MPARS es sound, garantizando que estamos razonando sobre sistemas que tienen significado. En esta sección veremos además que no existen evoluciones sound (de soporte finito) que no sean inducidas por un MPARS, y entonces no estamos excluyendo ninguna al restringirnos a los MPARS.

El hecho de que la evolución de una distribución deba seguir los axiomas de composicionalidad nos da la intuición que sólo las distribuciones "atómicas" son las que importan, y que la evolución está definida por ellas. Un MPARS define precisamente cuales son las distribuciones de elementos individuales, coincidiendo con esta intuición.

Pensando en esa equivalencia, podemos formalizar el argumento de la siguiente manera, restringiéndonos a distribuciones con soporte finito.

Definición 6.10. Una distribución D se llama *de soporte finito* (notado $\mathrm{FS}(D)$) si existen reales positivos α_i y elementos t_i tales que $D = \sum_{i=1}^{n} \alpha_i \overset{\circ}{t_i}$.

Notamos con $\mathbb{D}_f(A)$ al conjunto de distribuciones con soporte finito. Es decir,

$$\mathbb{D}_f(A) = \{D \in \mathbb{D}(A) \mid \mathrm{FS}(D)\}$$

Notar que toda distribución de soporte finito D tiene una *representación*, es decir, un $E \in \mathscr{D}(A)$ tal que $[\![E]\!] = D$. En otras palabras, $[\![-]\!]$ es sobreyectiva en $\mathbb{D}_f(A)$.

Tomemos una evolución de distribuciones $(\mathbb{D}_f(A), \to)$ sound. Construimos el MPARS $(A, \longmapsto)$ definido por[2]:

$$\frac{\overset{\circ}{t} \to^* [\![E]\!]}{t \longmapsto E}$$

Notar que debemos tener $E \in \mathscr{D}_1(A)$ ya que $\to$ preserva el peso, y $\overset{\circ}{t}$ es propia. Demostraremos que podemos caracterizar la evolución de distribuciones con el MPARS propuesto.

Teorema 6.11 (Completitud de MPARS)**.** *Dados $D, E \in \mathscr{D}_1(T)$, tenemos*

$$D \twoheadrightarrow^* E \iff [\![D]\!] \to^* [\![E]\!]$$

Demostración. **Ida.** Alcanza con tomar un sólo paso de $\twoheadrightarrow$, y el resultado se sigue por inducción. Sea $D = \left[(\alpha_i, t_i)\right]_i$ y $D \twoheadrightarrow E$. Si el paso es una equivalencia, tenemos $[\![D]\!] = [\![E]\!]$.

[2]Esta construcción podría ser más "eficiente" tomando una única representación. No nos interesa hacer la simplificación.

Si no, tenemos

$$D = D_1 + \left[(p, a)\right] + D_2 \twoheadrightarrow_E D_1 + pA + D_2 = E$$

con $a \longmapsto A$. Por la definición del MPARS, tenemos $\overset{\circ}{a} \to^* [\![A]\!]$. Entonces, usando AxComp, tenemos

$$[\![D]\!] = [\![D_1]\!] + p\overset{\circ}{a} + [\![D_2]\!] \to^* [\![D_1]\!] + p\,[\![A]\!] + [\![D_2]\!] = [\![E]\!]$$

Vuelta. Sea $D = \left[(\alpha_i, t_i)\right]_i$ y entonces $[\![D]\!] = \sum_{i=1}^{n} \alpha_i \overset{\circ}{t_i}$. Aplicando AxInv repetidamente tenemos que $[\![E]\!]$ debe ser igual a $\sum_{i=1}^{n} \alpha_i E_i$ con $\overset{\circ}{t_i} \to^* E_i$ para cada i. Por la construcción del MPARS y dado que $[\![-]\!]$ es sobre, tenemos $t_i \longmapsto E'_i$ con $[\![E'_i]\!] = E_i$. Evolucionando punto a punto conseguimos

$$D = \left[(\alpha_i, t_i)\right]_i \twoheadrightarrow_P \alpha_1 E'_1 + \cdots + \alpha_n E'_n$$

Además:

$$[\![\alpha_1 E'_1 + \cdots + \alpha_n E'_n]\!] = \sum_{i=1}^{n} \alpha_i \,[\![E'_i]\!] = \sum_{i=1}^{n} \alpha_i E_i = [\![E]\!]$$

Entonces, por el Lema 6.3 tenemos que son distribuciones equivalentes y entonces:

$$\alpha_1 E'_1 + \cdots + \alpha_n E'_n \twoheadrightarrow^* E$$

completando la prueba. ■

Hemos dado evidencia que los MPARS representan a todas las evoluciones (composicionales y de soporte finito) de distribuciones. Entonces, nuestro análisis de la propiedad de confluencia y cómo demostrarla aplica a cualquier sistema de reescritura de distribuciones que cumpla los axiomas.

Capítulo 7

Conclusiones

7.1. Resumen de aportes

Hemos introducido una noción de sistema de reescritura multiprobabilista (MPARS) que permite modelar una amplia variedad de lenguajes de programación. En particular, mostramos cómo Q^* y λ_1 pueden ser modelados como MPARS, brindando un soporte común a ambos.

Definimos una noción de reducción de distribuciones $\mathrm{Det}(\mathcal{M})$, que usamos para modelar evoluciones probabilistas (posiblemente parciales). En $\mathrm{Det}(\mathcal{M})$ definimos y analizamos la propiedad de confluencia de distribuciones, discutiendo que es apropiada al implicar la unicidad de distribuciones terminales y consistencia ecuacional sobre distribuciones. Notamos que aun si uno rechaza la evolución parcial, nuestra noción de confluencia es útil como método de prueba.

Luego, simplificamos la tarea de demostrar dicha confluencia. En particular, demostramos que la equivalencia entre distribuciones es "benigna", pudiendo obviarla en la parte superior de todo diagrama. Además, conseguimos múltiples criterios análogos al cambio de relación, la propiedad diamante, semiconfluencia, y el lema de Newman, dando evidencia de que la tarea no parece ser fundamentalmente más difícil al caso tradicional.

Usamos estos criterios para simplificar una prueba de confluencia existente para Q^* (incluso mejorando estrictamente el resultado y demostrando una conjetura planteada por sus autores) y para demostrar la confluencia de λ_1.

Luego, mostramos que los MPARS caracterizan a todas las evoluciones de distribuciones de probabilidad composicionales, dando evidencia de que la aplicación de los criterios demostrados es amplia.

7.2. Trabajo relacionado

Existen otros trabajos sobre confluencia en reescritura probabilista. Los exponemos uno por uno en esta sección comparándolos a nuestro desarrollo.

Confluencia de Bournez y Kirchner 2002

Los autores definen, en el contexto de los PARS, la siguiente noción de confluencia:

> **Definición 7.1** (BK-confluencia)**.** Un PARS $\mathcal{A}$ es *probabilísticamente confluente* si tenemos $P(\exists d.\ b \to^* d \leftarrow^* c) > 0$ dado que $b \leftarrow^* a \to^* c$, con ambos caminos elegidos independientemente.

donde $P(E)$ es la probabilidad de que E sea cierto. Por sobre esto, definen confluencia *casi certera* forzando a que la probabilidad sea exactamente 1, y una versión local de ambas propiedades.

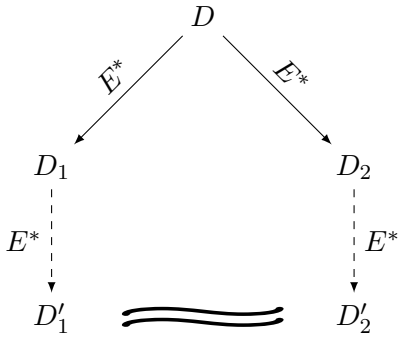
Como vemos, esta noción de confluencia habla sobre las divergencias probabilistas del sistema de reescritura (en efecto, no hay otras). De alguna forma, la propiedad indica que el PARS confluye "por sí sólo".

Estando motivados por los lenguajes probabilistas (y en particular cuánticos), esta noción no nos es de utilidad. En efecto, la medición cuántica o un choice probabilista nunca podrá atenerse a esta definición de confluencia, pero realmente queremos permitir distintos resultados finales como efecto de la reducción probabilista. Más aún, vimos que ciertos lenguajes no son siquiera expresables como un PARS.

Confluencia de Díaz-Caro y col. 2011

En este trabajo, se define una noción de confluencia similar. De hecho, nuestra reescritura de distribuciones y noción de confluencia están fuertemente basadas en este desarrollo previo. Con nuestra notación, esta confluencia puede expresarse como:

> **Definición 7.2.** Un MPARS $\mathcal{M}$ es *DC-confluente* si siempre que $D_1 \twoheadleftarrow^*_E D \twoheadrightarrow^*_E D_2$ existen D'_1, D'_2 equivalentes tales que $D_1 \twoheadrightarrow^*_E D'_1$ y $D_2 \twoheadrightarrow^*_E D'_2$.
>
> Gráficamente:



La DC-confluencia es estrictamente más fuerte a la confluencia de distribuciones. Por el Teorema 4.24 se sigue de forma directa que DC $\implies$ CR. Además, el siguiente MPARS es confluente pero no DC-confluente.

Ejemplo 7.3 (CR $\not\Longrightarrow$ DC)**.** Sea M dado por:

$$a \longmapsto \left[(\frac{1}{2}, b), (\frac{1}{2}, c)\right] \qquad a \longmapsto \left[(\frac{2}{3}, b), (\frac{1}{3}, c)\right] \qquad c \longmapsto \left[(\frac{1}{2}, b), (\frac{1}{2}, c)\right]$$

Entonces tenemos:

$$\left[(\frac{1}{2}, b), (\frac{1}{2}, c)\right] \twoheadleftarrow_E \left[(1, a)\right] \twoheadrightarrow_E \left[(\frac{2}{3}, b), (\frac{1}{3}, c)\right]$$

donde estas dos distribuciones no pueden hacerse equivalentes vía $\twoheadrightarrow_E$. Para convencerse de esto, notar que sólo c puede evolucionar. Por lo tanto las evoluciones de la izquierda siempre tienen peso $\frac{1}{2^n}$ para c y las de la derecha $\frac{1}{3\times 2^m}$. Dado que estas dos cantidades nunca pueden ser iguales, el sistema no es DC-confluente.

Sin embargo, el sistema es confluente, ya que a es el único elemento no determinista y podemos unir sus dos evoluciones en un paso. En efecto, tenemos:

$$\begin{array}{rcl} \left[(\frac{1}{2}, b), (\frac{1}{2}, c)\right] & \approx & \left[(\frac{1}{2}, b), (\frac{1}{3}, c), (\frac{1}{6}, c)\right] \\ & \twoheadrightarrow_P & \left[(\frac{1}{2}, b), (\frac{1}{6}, b), (\frac{1}{6}, c), (\frac{1}{6}, c)\right] \\ & \approx & \left[(\frac{2}{3}, b), (\frac{1}{3}, c)\right] \end{array}$$

y entonces

$$\begin{array}{rcl} \left[(\frac{1}{2}, b), (\frac{1}{2}, c)\right] & \twoheadrightarrow_{P/\approx} & \left[(\frac{2}{3}, b), (\frac{1}{3}, c)\right] \\ \left[(\frac{2}{3}, b), (\frac{1}{3}, c)\right] & \twoheadrightarrow_{P/\approx} & \left[(\frac{2}{3}, b), (\frac{1}{3}, c)\right] \end{array}$$

Por lo tanto, por el Teorema 4.29, tenemos la confluencia.

Aun siendo más débil, nuestra versión de la confluencia implica la deseada unicidad de distribuciones. Por sobre ello, esta versión es más fácil para razonar debido a la posibilidad de realizar evoluciones parciales como hemos visto.

Notar en el ejemplo previo que, intuitivamente, a debería ser considerado confluente ya que su evolución siempre "tiende" a $\left[(1, b)\right]$.

Por sobre la definición, en este trabajo se demuestra la (preservación de) confluencia de una extensión a λ_q, el $\lambda-$cálculo cuántico de André van Tonder (van Tonder 2004), con medición explícita.

Confluencia de Q^* en Dal Lago, Masini y Zorzi 2011

Ya hemos discutido bastante esta noción, notando que se corresponde con nuestra unicidad de formas normales. La discutiremos de nuevo a modo de resumen.

En este trabajo se define (y demuestra) confluencia de Q^* usando una noción de evolución basada en árboles de computación (posiblemente infinitos).

La noción de confluencia es más débil: se demuestra que todos los árboles *maximales* de un elemento son equivalentes. De esta manera, el resultado es más similar a unicidad de formas normales que a la confluencia.

Usando nuestra noción, y un lema sintáctico demostrado en el artículo, pudimos demostrar confluencia y por lo tanto recuperar este resultado de unicidad de manera más simple. De esta forma, contestamos afirmativamente a una conjetura planteada por los autores sobre la generalidad de su demostración de confluencia. La equivalencia considerada en el artículo es más fuerte, ya que también iguala la cantidad de hojas (para cada elemento) a cada lado. Podríamos también recuperar este resultado, usando una equivalencia de distribuciones que no permita Join ni Split (en efecto, considerando a $\frac{1}{2}D + \frac{1}{2}D \neq D$), pero no hicimos el desarrollo al no resultarnos relevante.

Resumen

Según nuestro conocimiento, la noción de confluencia propuesta en este trabajo es estrictamente más débil a todas las otras nociones relevantes. Es decir, se aceptan como confluentes sistemas que las otras rechazan. Aun así, tenemos las propiedades que esperamos de una noción de confluencia, como son la unicidad de formas normales y la consistencia ecuacional del cálculo.

Con los criterios demostrados, nuestra propiedad resulta ampliamente reusable. Para demostrarla en un cálculo concreto, muchas de las intuiciones sobre demostrar confluencia en un ARS se traspasan a este caso multiprobabilista.

7.3. Trabajo futuro

Detallamos algunas líneas de investigación parecen que prometedoras.

Normalización casi certera Se podría relacionar este estudio de confluencia con el concepto de *almost-sure termination* (Bournez y Garnier 2005; Dal Lago y Grellois 2017), otra noción de normalización en cálculos multiprobabilistas. Informalmente, la propiedad indica que la probabilidad de que un término diverja es exactamente 0, aun si pueden existir cadenas infinitas. Sería interesante relacionar la propiedad a la reescritura de distribuciones propuesta en nuestro trabajo.

Generalizar a un cuerpo arbitrario Durante todo este desarrollo hemos asumido que los pesos de una distribución son reales positivos, y que las distribuciones sucesor de un MPARS tienen peso 1. Si bien la justificación para introducir la reescritura de $\mathrm{Det}(\mathcal{M})$ fue simular distribuciones de probabilidad, no aplicamos leyes del área de probabilidad a nuestros razonamientos. Entonces, se piensa que muchos resultados demostrados son generalizables a un cuerpo arbitrario $\mathcal{K}$ (aunque no fue demostrado). En particular, permitir pesos negativos parece ser aceptable. Si bien no es claro como interpretar una reescritura de ese estilo, una posibilidad es investigar la relación con las probabilidades negativas que se discuten en (Feynman 1987).

Reescritura de términos Por sobre el estudio de reescritura abstracta (donde no se analizan los elementos) existe la reescritura de *términos* modelada por los *term rewriting systems (TRS)*. Sería interesante extender resultados de confluencia sobre TRS a los *MPTRS* (para alguna definición adecuada de los mismos). Paradójicamente, la reescritura entre λ-términos no forma, al menos de manera directa, un TRS (debido al efecto del binding y la sustitución). Entonces, los resultados sobre MPTRS pueden no extenderse a lenguajes de programación.

Mecanización Sería interesante mecanizar el desarrollo, sobre todo el contenido del Capítulo 4, en algún asistente de pruebas como Coq o Agda. De esta forma, la formalización podría servir de librería para demostrar confluencia de un lenguaje probabilista, o para razonar sobre la reescritura de distribuciones inducida por el mismo. Pensamos que esto es posible de una manera bastante directa.

Liberalizar λ_1 Debería ser posible extender λ_1 de varias maneras. Primero, dijimos que la elección de CBN era arbitraria, y CBV funcionaría también. De hecho, podrían coexistir con distintos símbolos para las abstracciones. Además, una tercer opción de abstracción debería ser posible en λ_1: aquellas que pueden duplicar y ser aplicada a cualquier término mientras no sea probabilista. Sería interesante extender el cálculo y la prueba de confluencia para el mismo. Para la segunda extensión, seguramente debería hacerse un cambio a una reducción paralela como en la prueba mostrada para el λ-cálculo.

Medición explícita Nuestro ejemplo de λ_1 da evidencia de que la linealidad es central para que un lenguaje con choice probabilista sea confluente. Sin embargo, en los lenguajes cuánticos, las superposiciones no reducen espontáneamente a un valor, sino que deben ser explícitamente medidas. Sería interesante diseñar un cálculo modelo que permita expresar superposiciones (no necesariamente cuánticas) y medirlas, y conseguir requerimientos para su confluencia.

Sobrepasar el soporte finito Una asunción que atraviesa todo el desarrollo es que las distribuciones tienen soporte finito, para poder modelarlas como listas. Además, asumimos que los conjuntos sobre los cuales formamos distribuciones son numerablemente infinitos. Debería ser posible extender el desarrollo para evitar ambas restricciones, tal vez usando una definición coinductiva.

Apéndice A

Definición de $\mathbf{Q}^*$

Tomado de (Dal Lago, Masini y Zorzi 2011), adaptado en notación.

A.1. Sintaxis

$$
\begin{array}{rcll}
x & ::= & x_0 \mid x_1 \mid \ldots & \textit{variables} \\
r & ::= & r_0 \mid r_1 \mid \ldots & \textit{variables cuánticas} \\
\pi & ::= & x \mid \langle x_1, \ldots x_n \rangle & \textit{patrones lineales} \\
\psi & ::= & \pi \mid !x & \textit{patrones} \\
B & ::= & 0 \mid 1 & \textit{constantes booleanas} \\
U & ::= & U_0 \mid U_1 \mid \ldots & \textit{operadores unitarios} \\
C & ::= & B \mid U & \textit{constantes} \\
M, N & ::= & x \mid r \mid C \mid M_1 M_2 \mid !M & \\
 & \mid & \mathbf{new}(M) \mid \mathbf{meas}(M) & \textit{términos} \\
 & \mid & \langle M_1, \ldots M_n \rangle \mid \lambda\psi.M &
\end{array}
$$

A.2. Buena formación

Debemos imponer las restricciones de linealidad.

Notamos con Γ a un conjunto de variables "lineales", tanto cuánticas como no, que sólo pueden ser (y deben ser) usadas una vez. Notamos con $!\Delta$ a un conjunto de variables clásicas con "bangs", es decir, a un conjunto de la forma $!x_i, !x_j, \ldots$. La unión de conjuntos se nota como Γ_1, Γ_2, donde se sobreentiende que Γ y Δ no comparten variables. Ambos conjuntos pueden ser vacíos.

La forma del juicio es: $\Gamma, !\Delta \vdash M$. Las siguientes reglas definen a la buena formación.

$$\frac{}{!\Delta \vdash C}\ \text{WF-Const} \qquad \frac{}{r, !\Delta \vdash r}\ \text{WF-QVar} \qquad \frac{}{x, !\Delta \vdash x}\ \text{WF-CVar}$$

$$\frac{}{!x, !\Delta \vdash x}\ \text{WF-Der} \qquad \frac{!\Delta \vdash M}{!\Delta \vdash !M}\ \text{WF-Prom}$$

$$\frac{\Gamma_1, !\Delta \vdash M \quad \Gamma_2, !\Delta \vdash N}{\Gamma_1, \Gamma_2, !\Delta \vdash MN}\ \text{WF-App} \qquad \frac{\Gamma_i, !\Delta \vdash M_i}{\Gamma_1, \ldots, \Gamma_n, !\Delta \vdash \langle M_i, \ldots M_n \rangle}\ \text{WF-Tens}$$

$$\frac{\Gamma, !\Delta \vdash M}{\Gamma, !\Delta \vdash \mathbf{new}(M)}\ \text{WF-New} \qquad \frac{\Gamma, x_1, \ldots x_n, !\Delta \vdash M}{\Gamma, !\Delta \vdash \lambda\langle x_1, \ldots x_n \rangle.M}\ \text{WF-Lam1}$$

$$\frac{\Gamma, x, !\Delta \vdash M}{\Gamma, !\Delta \vdash \lambda x.M}\ \text{WF-Lam2} \qquad \frac{\Gamma, !\Delta, !x \vdash M}{\Gamma, !\Delta \vdash \lambda !x.M}\ \text{WF-Lam3}$$

$$\frac{\Gamma, !\Delta \vdash M}{\Gamma, !\Delta \vdash \mathbf{meas}(M)}\ \text{WF-Meas} \qquad \frac{\Gamma \vdash M \quad !\Delta \vdash N \quad !\Delta \vdash O}{\Gamma, !\Delta \vdash \mathbf{if}\ M\ \mathbf{then}\ N\ \mathbf{else}\ O}\ \text{WF-If}$$

A.3. Semántica

Notamos con $\mathbf{Q}(M)$ al conjunto de variables cuánticas libres de M.

Definición A.1. Una preconfiguración es una terna $[\mathcal{Q}, \mathcal{QV}, M]$ donde:

- M es un término
- $\mathcal{QV}$ es un conjunto finito de variables cuánticas tal que $\mathbf{Q}(M) \subseteq \mathcal{QV}$
- $\mathcal{Q}$ es un *estado cuántico*: un vector normalizado sobre las variables en $\mathcal{QV}$. Por ejemplo, si $\mathcal{QV} = \{r, s\}$, $\mathcal{Q}$ toma la forma

$$\begin{aligned} & \alpha_{00} \left| r \mapsto 0 \right\rangle \otimes \left| s \mapsto 0 \right\rangle \\ + \quad & \alpha_{01} \left| r \mapsto 0 \right\rangle \otimes \left| s \mapsto 1 \right\rangle \\ + \quad & \alpha_{10} \left| r \mapsto 1 \right\rangle \otimes \left| s \mapsto 0 \right\rangle \\ + \quad & \alpha_{11} \left| r \mapsto 1 \right\rangle \otimes \left| s \mapsto 1 \right\rangle \end{aligned}$$

Sea $\Theta : \mathcal{QV} \to \mathcal{RV}$ biyectiva. Si $\mathbf{Q}(M) \subseteq \mathcal{QV}$ definimos $\Theta(M)$ de la manera obvia cambiando todas las variables cuánticas según Θ. Similarmente, podemos extender Θ a un mapeo entre estados cuánticos.

Definición A.2 (Configuración)**.** Dos preconfiguraciones $[\mathcal{Q}, \mathcal{QV}, M], [\mathcal{R}, \mathcal{RV}, N]$ se llaman *equivalentes* cuando exista $\Theta : \mathcal{QV} \to \mathcal{RV}$ biyectiva tal que $N = \Theta(M)$ y $\mathcal{R} = \Theta(\mathcal{Q})$.

Una *configuración* es una clase de equivalencia de preconfiguraciones

según la noción anterior.

Las reducciones son entre configuraciones. Está definida inductivamente por las siguientes reglas.

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, (\lambda x.M)N] \to^1_{\mathsf{l}.\beta} [\mathcal{Q}, \mathcal{QV}, M[N/x]]}$$

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, (\lambda !x.M)!N] \to^1_{\mathsf{c}.\beta} [\mathcal{Q}, \mathcal{QV}, M[N/x]]}$$

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, (\lambda \langle x_1, \dots x_n\rangle.M)\langle r_1, \dots, r_n\rangle] \to^1_{\mathsf{q}.\beta} [\mathcal{Q}, \mathcal{QV}, M[r_1/x_1, \dots, r_n/x_n]]}$$

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, \mathbf{if}\ 1\ \mathbf{then}\ M\ \mathbf{else}\ N] \to^1_{\mathsf{if}_1} [\mathcal{Q}, \mathcal{QV}, M]}$$

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, \mathbf{if}\ 0\ \mathbf{then}\ M\ \mathbf{else}\ N] \to^1_{\mathsf{if}_0} [\mathcal{Q}, \mathcal{QV}, N]}$$

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, U\langle r_{i_1}, \dots, r_{i_n}\rangle] \to^1_{Uq} \left[U_{\langle\langle r_{i_1}, \dots, r_{i_n}\rangle\rangle}\mathcal{Q}, \mathcal{QV}, \langle r_{i_1}, \dots, r_{i_n}\rangle\right]}$$

$$\frac{c \in \{0,1\} \quad p_c = \langle \mathcal{Q}|m^\dagger_{r,c}m_{r,c}|\mathcal{Q}\rangle}{[\mathcal{Q}, \mathcal{QV}, \mathbf{meas}(r)] \to^{p_c}_{\mathbf{meas}_r} [\mathcal{M}_{r,c}(\mathcal{Q}), \mathcal{QV} - \{r\}, !c]}$$

$$\frac{r \text{ variable fresca}}{[\mathcal{Q}, \mathcal{QV}, \mathbf{new}(c)] \to^1_{\mathbf{new}} [\mathcal{Q} \otimes |r \mapsto c\rangle, \mathcal{QV} \cup \{r\}, r]}$$

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, L((\lambda\pi.M)N)] \to^1_{\mathsf{l.cm}} [\mathcal{Q}, \mathcal{QV}, (\lambda\pi.LM)N]}$$

$$\frac{}{[\mathcal{Q}, \mathcal{QV}, ((\lambda\pi.M)N)L] \to^1_{\mathsf{r.cm}} [\mathcal{Q}, \mathcal{QV}, (\lambda\pi.ML)N]}$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, N]}{[\mathcal{Q}, \mathcal{QV}, \langle M_1, \dots, M, \dots M_k\rangle] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, \langle M_1, \dots, N, \dots M_k\rangle]}\ \mathsf{t}_i$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, N] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, P]}{[\mathcal{Q}, \mathcal{QV}, MN] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, MP]}\ \mathsf{r.a} \qquad \frac{[\mathcal{Q}, \mathcal{QV}, M] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, P]}{[\mathcal{Q}, \mathcal{QV}, MN] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, PN]}\ \mathsf{l.a}$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, N]}{[\mathcal{Q}, \mathcal{QV}, \mathbf{new}(M)] \to^p_\alpha [\mathcal{R}, \mathcal{RV}, \mathbf{new}(N)]}\ \mathsf{in.new}$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, N]}{[\mathcal{Q}, \mathcal{QV}, \mathbf{meas}(M)] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, \mathbf{meas}(N)]}\ \mathsf{in.meas}$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, N]}{[\mathcal{Q}, \mathcal{QV}, \mathbf{if}\ M\ \mathbf{then}\ L\ \mathbf{else}\ P] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, \mathbf{if}\ N\ \mathbf{then}\ L\ \mathbf{else}\ P]}\ \mathsf{in.if}$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, N]}{[\mathcal{Q}, \mathcal{QV}, (\lambda!x.M)] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, (\lambda!x.N)]}\ \mathsf{in}.\lambda_1$$

$$\frac{[\mathcal{Q}, \mathcal{QV}, M] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, N]}{[\mathcal{Q}, \mathcal{QV}, (\lambda\pi.M)] \to_\alpha^p [\mathcal{R}, \mathcal{RV}, (\lambda\pi.N)]}\ \mathsf{in}.\lambda_2$$

Donde notamos con $U_{\langle\langle r_{i_1},\ldots,r_{i_n}\rangle\rangle}$ al estado resultante luego de aplicar la matriz U al estado inicial $\mathcal{Q}$, sobre los qubits relevantes.

El conjunto de todas las etiquetas se denota con $\mathcal{L}$. Los conjuntos de etiquetas mencionados en el Lema 3.25 son

$$\begin{aligned} \mathcal{K} &= \{\mathsf{l.cm}, \mathsf{r.cm}\} \\ \mathcal{N} &= \mathcal{L} - (\mathcal{K} \cup \{\mathbf{meas}_r\}) \end{aligned}$$

Apéndice B

Demostraciones

B.1. Demostraciones del Capítulo 4

Demostración del Lema 4.16. Usando el criterio simplificado del Lema 1.37.

Supongamos que $D \twoheadrightarrow_{FJ} E \twoheadrightarrow_S F$. Debemos encontrar E' tal que $D \twoheadrightarrow_S E' \twoheadrightarrow^*_{FJ} F$. Procedemos por inducción en la forma de las reducciones.

- (F, S). Tenemos:

$$\begin{array}{lll} (p,a),(q,b),ds & \twoheadrightarrow_F & (q,b),(p,a),ds \\ & \twoheadrightarrow_S & (q_1,b),(q_2,b),(p,a),ds \end{array}$$

  Tomamos $N' = (p,a),(q_1,b),(q_2,b),ds$ (dos pasos Flip).

- (J, S). Tenemos:

$$\begin{array}{lll} (p,a),(q,a),ds & \twoheadrightarrow_J & (p+q,a),ds \\ & \twoheadrightarrow_S & (\alpha,a),(\beta,a),ds \end{array}$$

  con $\alpha + \beta = p + q$.

  Analizamos por casos.

  Si $p < \alpha$, tomamos $E' = (p,a),(\alpha - p,a),(q + p - \alpha,a),ds$ y hacemos Join de los primeros dos elementos.

  Si $p \geq \alpha$, tomamos $E' = (p + q - \beta,a),(\beta - q,a),(q,a),ds$ y hacemos Join del segundo y tercer elemento.

- (Tx, S). Tenemos :

$$(p,a),ds \twoheadrightarrow_{Tx} (p,a),ds' \twoheadrightarrow_S (p_1,a),(p_2,a),ds'$$

  Tomamos $E' = (p_1,a),(p_2,a),ds$, y aplicamos T^2x.

- (F, TS). Tenemos:

$$(p,a),(q,b),ds \twoheadrightarrow_F (q,b)(p,a),ds \twoheadrightarrow_S (q_1,b),(q_2,b),(p,a),ds$$

  Tomamos $E' = (p,a),(q_1,b),(q_2,b),ds$ (dos pasos de F).

- (J, TS). Tenemos:

$$\begin{array}{lll}(p,a),(q,a),(r,b),ds & \twoheadrightarrow_J & (p+q,a),(r,b),ds \\ & \twoheadrightarrow_S & (p+q,a),(r_1,b),(r_2,b),ds\end{array}$$

  Tomamos $E' = (p,a),(q,a),(r_1,b),(r_2,b),ds$.

- (Tx, Ty). Inmediato por HI. ■

Notar que si tenemos $(p,a),ds \twoheadrightarrow_P E$, sabemos que E tiene la forma $pA +\!\!+ ds'$ donde $ds \twoheadrightarrow_P ds'$ y tenemos o bien $a \longmapsto A$ o bien $A = \left[(1,a)\right]$. Es decir, $\left[(1,a)\right] \twoheadrightarrow_P A$. Usaremos esta descomposición a continuación.

Demostración del Lema 4.18. Usamos el criterio simplificado del Lema 1.37.

Supongamos que $D \twoheadrightarrow_{FJ} E \twoheadrightarrow_P F$. Buscamos E' tal que $D \twoheadrightarrow_P E' \twoheadrightarrow^*_{FJ} F$. Por inducción en la forma de la primer reducción.

- Flip. Tenemos

$$\begin{array}{lll}(p_1,a_1),(p_2,a_2),ds & \twoheadrightarrow_{FJ} & (p_2,a_2),(p_1,a_1),ds \\ & \twoheadrightarrow_P & p_2A_2 +\!\!+ p_1A_1 +\!\!+ ds'\end{array}$$

  para algunas A_1, A_2, ds' con $\left[(1,a_i)\right] \twoheadrightarrow_P A_i$ $(i = 1,2)$ y $ds \twoheadrightarrow_P ds'$.

  Tomamos $E' = p_1A_1 +\!\!+ p_2A_2 +\!\!+ ds'$. Por composicionalidad y multiplicidad tenemos que $D \twoheadrightarrow_P E'$. Además, $E' \twoheadrightarrow^*_F F$ haciendo suficientes Flips para intercambiar los conjuntos de lugar.

- Join. Tenemos

$$\begin{array}{lll}(p_1,a),(p_2,a),ds & \twoheadrightarrow_{FJ} & (p_1+p_2,a),ds \\ & \twoheadrightarrow_P & (p_1+p_2)A +\!\!+ ds'\end{array}$$

  para algunas A, ds' con $\left[(1,a)\right] \twoheadrightarrow_P A$ y $ds \twoheadrightarrow_P ds'$.

  Tomamos $E' = p_1A +\!\!+ p_2A +\!\!+ ds'$. Por composicionalidad y multiplicidad tenemos que $D \twoheadrightarrow_P E'$. Además, $E' \twoheadrightarrow^*_{FJ} F$ haciendo suficientes Flips para "intercalar" los conjuntos y luego hacer Join para cada par de elementos.

- Tail. Tenemos

$$\begin{array}{lll}(p,a),ds & \twoheadrightarrow_{FJ} & (p,a),ds' \\ & \twoheadrightarrow_P & pA +\!\!+ ds''\end{array}$$

  para algunas A, ds'' con $\left[(1,a)\right] \twoheadrightarrow_P A$ y $ds' \twoheadrightarrow_P ds''$.

Por la premisa de TAIL tenemos $ds \twoheadrightarrow_{FJ} ds'$. Por la HI, entonces, tenemos que existe un E'' tal que $ds \twoheadrightarrow_P E'' \twoheadrightarrow_{FJ} ds''$.

Tomamos $E' = pA +\!\!+ E''$. ■

Demostración del Lema 4.19. Analizaremos el caso cuando se parte desde un singleton. La prueba se extiende fácilmente a los otros casos, dado que el SPLIT ocurre dentro del alcance de un único elemento.

Tenemos $(1, a) \twoheadrightarrow_P \left[(p_i, a_i)\right]_i$, donde $a \longmapsto \left[(p_i, a_i)\right]_i$. El SPLIT ocurre en alguna posición, llamémosla k. Entonces la reducción es:

$$\left[(p_i, a_i)\right]_i \twoheadrightarrow_S (p_1, a_1) : \cdots : (\alpha, a_k) : (p_k - \alpha, a_k) : \cdots : (p_n, a_n)$$

Sea $E = (\frac{\alpha}{p_k}, a) : (1 - \frac{\alpha}{p_k}, a) : ds$. Podemos usar la evolución paralela, por igual en ambos pares, para evolucionar a:

$$\begin{array}{rcl} (\frac{\alpha}{p_k})\left[(p_i, a_i)\right]_i & +\!\!+ & (1 - \frac{\alpha}{p_k})\left[(p_i, a_i)\right]_i \\ = \quad \left[(\frac{\alpha}{p_k} p_i, a_i)\right]_i & +\!\!+ & \left[((1 - \frac{\alpha}{p_k}) p_i, a_i)\right]_i \end{array}$$

Haciendo FLIPS para intercalar las distribuciones, y uniendo con JOIN para todo $i \neq k$ llegamos a las mismas probabilidades de la otra distribución. Para k, tenemos los pesos $\frac{\alpha}{p_k}\ p_k = \alpha$ y $(1 - \frac{\alpha}{p_k})\ p_k = p_k - \alpha$, como se buscaba. ■

B.2. Demostraciones sobre λ_1

Demostramos algunos lemas auxiliares.

Lema B.1 (Debilitamiento)**.** *Si* $\Gamma \subseteq \Gamma'$ *y* $\Delta \subseteq \Delta'$*, entonces* $\Gamma \mid \Delta \vdash M$ *implica* $\Gamma' \mid \Delta' \vdash M$*.*

Demostración. Por inducción en la buena formación de M.

- WF-VAR, WF-VAR!. Trivial.
- WF-APP.

  Tenemos $M = PQ$ con $\Gamma = \Gamma_1 \cup \Gamma_2$ y

  $$\Gamma_1 \mid \Delta \vdash P \qquad \Gamma_2 \mid \Delta \vdash Q$$

  Sea $\Gamma_1' = (\Gamma' - \Gamma) \cup \Gamma_1$. Por la HI, tenemos

  $$\Gamma_1' \mid \Delta \vdash P$$

  Entonces, por WF-APP, tenemos

  $$\Gamma_1' \cup \Gamma_2 \mid \Delta \vdash PQ$$

  Dado que $(\Gamma' - \Gamma) \cup \Gamma_1 \cup \Gamma_2 = \Gamma'$, esta es nuestra meta.

- WF-$\oplus$. Análogo al caso para la aplicación.
- WF-Abs.

  Tenemos $M = \lambda x.P$ con la premisa

  $$\Gamma, x \mid \Delta \vdash P$$

  Por la HI, tenemos

  $$\Gamma', x \mid \Delta \vdash P$$

  (si $x \in \Gamma' - \Gamma$, debemos α-convertir)

  Concluimos por WF-Abs.
- WF-Abs!.

  Análogo al caso WF-Abs.
- WF-Bang.

  Tenemos $\Gamma \mid \Delta \vdash !M$ por la premisa

  $$\cdot \mid \Delta \vdash M$$

  Por la HI, tenemos

  $$\cdot \mid \Delta' \vdash M$$

  Concluimos por WF-Bang, usando Γ'. ■

Lema B.2. *Si* $\Gamma_1, x \mid \Delta \vdash M$ *y* $\Gamma_2 \mid \Delta \vdash N$ *entonces* $\Gamma_1 \cup \Gamma_2 \mid \Delta \vdash M[N/x]$.

Demostración. Por inducción en la buena formación de M.

- WF-Var.

  Tenemos $M = y$ con $y \in \Gamma, x$.

  Si $M = x$ tenemos $M[N/x] = N$ y concluimos por la buena formación de N, aplicando el Lema B.1.

  Si $M = y$, tenemos $M[N/x] = y$ y concluimos por WF-Var.
- WF-App.

  Tenemos $M = PQ$, con

  $$\Gamma_3 \mid \Delta \vdash P \qquad \Gamma_4 \mid \Delta \vdash Q \qquad \Gamma_3 \cup \Gamma_4 = \Gamma_1, x$$

  Sólo uno de Γ_3 y Γ_4 contiene a x. Supongamos que es $\Gamma_3 = \Gamma_3', x$. Entonces, aplica nuestra HI y tenemos que

  $$\Gamma_3', x \mid \Delta \vdash P \implies \Gamma_3' \cup \Gamma_2 \mid \Delta \vdash P[N/x]$$

Entonces, por WF-App

$$\Gamma'_3 \cup \Gamma_2 \cup \Gamma_4 \mid \Delta \vdash P[N/x]\ Q$$

Como tenemos que $\Gamma'_3 \cup \Gamma_4 = \Gamma_1$ y x no está libre en Q, esta es nuestra meta. Análogamente se demuestra el caso cuando $x \in \Gamma_4$.

- WF-$\oplus$. Análogo al caso de la aplicación.
- WF-Abs.

  Tenemos $M = \lambda y.P$ con $\Gamma_1, x, y \mid \Delta \vdash P$, con $y \neq x$. Por la HI, tenemos $\Gamma_1 \cup \Gamma_2, y \mid \Delta \vdash P[N/x]$. Aplicando WF-Abs, tenemos $\Gamma_1 \cup \Gamma_2 \mid \Delta \vdash \lambda y.P[N/x]$, nuestra meta.
- WF-Abs!.

  Tenemos $M = \lambda !y.P$ con $\Gamma_1, x \mid \Delta, y \vdash P$. Por la HI, $\Gamma_1 \cup \Gamma_2 \mid \Delta, y \vdash P[N/x]$. Aplicando WF-Abs!, tenemos nuestra meta.
- WF-Var!. No aplica.
- WF-Bang.

  Tenemos $M = !P$. Por la premisa, tenemos $\cdot \mid \Delta \vdash P$, y entonces x no está libre en P. Concluimos por WF-Bang. ■

Lema B.3. *Si* $\Gamma \mid \Delta, x \vdash M$ $y \cdot \mid \Delta \vdash N$ *entonces* $\Gamma \mid \Delta \vdash M[N/x]$.

Demostración. Por inducción en la buena formación de M.

- WF-Var.

  Tenemos $M = y$, con $y \in \Gamma$. La sustitución no tiene efecto ($y \neq x$), y debemos demostrar $\Gamma \mid \Delta \vdash y$. Concluimos con WF-Var.
- WF-Var!.

  Tenemos $M = y$, analizamos por casos. Si $y = x$, debemos demostrar $\Gamma \mid \Delta \vdash N$. Concluimos por nuestra hipótesis sobre N y el Lema B.1. Si $y \neq x$, debemos demostrar $\Gamma \mid \Delta \vdash y$. Por inversión, debemos tener $y \in \Delta$. Entonces, concluimos por WF-Var!.
- WF-App.

  Tenemos $M = PQ$. Por inversión, debemos tener

  $$\Gamma_1 \mid \Delta, x \vdash P \qquad \Gamma_2 \mid \Delta, x \vdash Q$$

  con $\Gamma = \Gamma_1 \cup \Gamma_2$. Aplicando la HI tenemos

  $$\Gamma_1 \mid \Delta \vdash P[N/x] \qquad \Gamma_2 \mid \Delta \vdash Q[N/x]$$

  Concluimos por WF-App.

- WF-$\oplus$. Análogo al caso de la aplicación.
- WF-Abs.

  Tenemos $M = \lambda y.P$ con $\Gamma, y \mid \Delta, x \vdash P$. Por la HI, tenemos $\Gamma, y \mid \Delta \vdash P[N/x]$. Concluimos por WF-Abs.
- WF-Abs!.

  Tenemos $M = \lambda!y.P$ con $\Gamma \mid \Delta, x, y \vdash P$ (debemos tener $y \neq x$). Por la HI tenemos $\Gamma \mid \Delta, y \vdash P[N/x]$. Concluimos por WF-Abs!.
- WF-Bang.

  Tenemos $M = !P$ con $\Gamma \mid \Delta, x \vdash !P$. Por inversión, tenemos $\cdot \mid \Delta, x \vdash P$. Aplicando la HI, tenemos $\cdot \mid \Delta \vdash P[N/x]$. Concluimos por WF-Bang. ■

Ahora, podemos demostrar la preservación de buena formación.

Demostración del Lema 5.2. Por inducción en $M \longmapsto D$.

- R-β.

  Tenemos $M = (\lambda x.P)Q$. Por inversión de la buena formación, tenemos

  $$\Gamma_1, x \mid \Delta \vdash P \qquad \Gamma_2 \mid \Delta \vdash Q \qquad \Gamma_1 \cup \Gamma_2 \mid \Delta \vdash M$$

  Por el Lema B.2 obtenemos el resultado.
- R-β!.

  Tenemos $M = (\lambda!x.P)!Q$. Por inversión de la buena formación, tenemos

  $$\Gamma \mid \Delta, x \vdash P \qquad \cdot \mid \Delta \vdash Q$$

  Concluimos aplicando el Lema B.3.
- R-AppL.

  Tenemos $M = PQ$ y $P \longmapsto D$. Por inversión de la buena formación, tenemos

  $$\Gamma_1 \mid \Delta \vdash P \qquad \Gamma_2 \mid \Delta \vdash Q \qquad \Gamma_1 \cup \Gamma_2 \mid \Delta \vdash M$$

  Por la HI, tenemos que $\Gamma_1 \mid \Delta \vdash P_i$ para cada P_i en D. Entonces, por WF-App, tenemos $\Gamma_1 \cup \Gamma_2 \mid \Delta \vdash P_i Q$ para cada i, como se necesita.
- R-AppR. Análogo al anterior.

- R-λ.

  Tenemos $M = (\lambda x.P)$ y $P \longmapsto D$. Por inversión de la buena formación, tenemos

  $$\Gamma, x \mid \Delta \vdash P$$

  Por la HI, tenemos que $\Gamma, x \mid \Delta \vdash P_i$ para cada P_i en D. Entonces, por WF-Abs, tenemos $\Gamma \mid \Delta \vdash \lambda x.P_i$ para cada i, como se necesita.

- R-λ!.

  Análogo al anterior.

- R-$\oplus$. Tenemos $M = P \oplus_q Q$. Por inversión de la buena formación, tenemos

  $$\Gamma_1 \mid \Delta \vdash P \qquad \Gamma_2 \mid \Delta \vdash Q$$

  Ya que $\Gamma_i \subseteq \Gamma$, podemos aplicar el Lema B.1 para conseguir nuestra meta de

  $$\Gamma \mid \Delta \vdash P \qquad \Gamma \mid \Delta \vdash Q$$

- R-$\oplus$-L y R-$\oplus$-R.

  Análogo a los casos R-AppL y R-AppR. ■

Apéndice C

Notación

Símbolo	Explicación/Definición	Introducción
$\mathbb{R}^+$	Números reales positivos	
$\mathscr{L}(A)$	Listas sobre el conjunto A	Página vii
$L +\!\!+ L'$	Concatenación de listas	Página vii
$\mathscr{D}(A)$	Distribuciones (de lista) sobre A	Página viii
$w(D)$	Peso total de una distribución	Página viii
$\mathscr{D}_1(A)$	Distribuciones normalizadas sobre A	Página viii
$\mathscr{D}_{0,1}(A)$	Distribuciones normalizadas o nulas sobre A	Página viii
αD	Escala de una distribución D por $\alpha \in \mathbb{R}^+$	Página viii
$\mathcal{A}$	ARS o PARS	Página 1 y 22
WN	Normalización débil	Página 5
SN	Normalización fuerte	Página 5
UN	Unicidad de formas normales	Página 6
CR	Confluencia, confluencia de distribuciones	Página 7 y 41
SCR	Semiconfluencia	Página 10
LC	Confluencia local	Página 11
$\Diamond$	Propiedad diamante	Página 13
$\mathcal{A} \models P$	El sistema $\mathcal{A}$ cumple la propiedad P	Página 6
$\alpha \dashv \beta$	α conmuta sobre β	Página 18
$\mathcal{T}(a)$	Árboles de computación	Página 24, 31 y 36
$\mathrm{supp}(T)$	Soporte de un árbol	Página 25
$\mathrm{Prob}(T, c)$	Probabilidad asignada a c en T	Página 25
$\mathcal{M}$	MPARS	Página 34
$\longmapsto$	Evolución puntual	Página 34

Símbolo	Explicación/Definición	Introducción
$\mathrm{Det}(\mathcal{M})$	Determinización de $\mathcal{M}$	Página 38
$\twoheadrightarrow$	Evolución de distribuciones	Página 38
$E, TF, T^2S, \ldots$	Notación explícita para evolución	Página 39
$\twoheadrightarrow_E$	Evolución propia	Página 39
$\sim$	Paso de equivalencia	Página 39
$\approx$	$= \sim^*$	
UTD	Unicidad de distribuciones terminales	Página 42
$\twoheadrightarrow_P$	Evolución paralela	Página 50
$\twoheadrightarrow_{SP}$	$= (\twoheadrightarrow_S \cup \twoheadrightarrow_P)$	Página 54
Q^*	Cálculo cuántico de Dal Lago, Masini y Zorzi 2011	
$\mathcal{Q}^*$	MPARS que representa a Q^*	Página 44
λ_1	λ-cálculo multiprobabilista	Página 68
$\mathbb{D}(A)$	Distribuciones (matemáticas) sobre A	Página 77
$\mathrm{FS}(D)$	Propiedad de soporte finito	Página 83

Bibliografía

Atzemoglou, Philip (2014). «The dagger lambda calculus». En: *Proceedings of ithe 11th workshop on Quantum Physics and Logic (QPL'14)*. Ed. por Bob Coecke, Ichiro Hasuo y Prakash Panangaden. Vol. 172. Electronic Proceedings in Theoretical Computer Science. Open Publishing Association, págs. 217-235.

Baader, Franz y Tobias Nipkow (1998). *Term Rewriting and All That*. Cambridge University Press.

Barendregt, Hendrik P. (1981). *The Lambda Calculus, its Syntax and Semantics*. Studies in Logic and the Foundations of Mathematics. North-Holland.

Bournez, Olivier y Florent Garnier (2005). «Proving Positive Almost-sure Termination». En: *Proceedings of the 16th International Conference on Term Rewriting and Applications (RTA'05)*. Ed. por Jürgen Giesl. Vol. 3467. Lecture Notes in Computer Science. Springer-Verlag, págs. 323-337.

Bournez, Olivier y Claude Kirchner (2002). «Probabilistic Rewrite Strategies. Applications to ELAN». En: *Proceedings of the 13th International Conference on Rewriting Techniques and Applications (RTA'02)*. Ed. por Sophie Tison. Vol. 2378. Lecture Notes in Computer Science. Springer-Verlag, págs. 252-266.

Church, Alonzo y James Barkley Rosser (1936). «Some Properties of Conversion». En: *Transactions of the American Mathematical Society* 39.3, págs. 472-482.

Dal Lago, Ugo y Charles Grellois (2017). *Probabilistic Termination by Monadic Affine Sized Typing*. Aceptado en el 26th European Symposium on Programming (ESOP'17). Versión extendida disponible en arXiv:1701.04089.

Dal Lago, Ugo, Andrea Masini y Margherita Zorzi (2011). «Confluence Results for a Quantum Lambda Calculus with Measurements». En: *Proceedings of the 6th International Workshop on Quantum Physics and Logic (QPL'09)*. Ed. por Bob Coecke, Prakash Panangaden y Peter Selinger. Vol. 270.2. Electronic Notes in Theoretical Computer Science. Elsevier, págs. 251-261.

Di Pierro, Alessandra, Chris Hankin y Herbert Wiklicky (2005). «Probabilistic λ-calculus and Quantitative Program Analysis». En: *Journal of Logic and Computation* 15.2, págs. 159-179.

Díaz-Caro, Alejandro y Gilles Dowek (2016). *Typing quantum superpositions and projective measurements*. En revisión. Disponible en arXiv:1601.04294.

Díaz-Caro, Alejandro, Pablo Arrighi, Manuel Gadella y Jonathan Grattage (2011). «Measurements and Confluence in Quantum Lambda Calculi With Explicit Qubits». En: *Proceedings of the Joint 5th International Workshop on Quantum Physics and Logic and 4th Workshop on Developments in Computational Models (QPL/DCM'08)*. Ed. por Bob Coecke, Ian Mackie, Prakash Panangaden y Peter Selinger. Vol. 270.1. Electronic Notes in Theoretical Computer Science. Elsevier, págs. 59-74.

Feynman, Richard (1987). «Negative Probability». En: *Quantum Implications: Essays in Honour of David Bohm*, págs. 235-248.

Huet, Gérard (1980). «Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems: Abstract Properties and Applications to Term Rewriting Systems». En: *Journal of the ACM* 27.4, págs. 797-821.

Newman, Maxwell H. A. (1942). «On Theories with a Combinatorial Definition of "Equivalence"». En: *Annals of Mathematics* 43.2, págs. 223-243.

Nielsen, Michael A. e Isaac L. Chuang (2011). *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 10th. Cambridge University Press.

Selinger, Peter (2004). «Towards a Quantum Programming Language». En: *Mathematical Structures in Computer Science* 14.4, págs. 527-586.

Selinger, Peter y Benoît Valiron (2005). «A Lambda Calculus for Quantum Computation with Classical Control». En: *Proceedings of the 7th International Conference on Typed Lambda Calculi and Applications (TLCA'05)*. Ed. por Paweł Urzyczyn. Vol. 3461. Lecture Notes in Computer Science. Springer-Verlag, págs. 354-368.

Sethi, Ravi (1974). «Testing for the Church-Rosser Property». En: *Journal of the ACM* 21.4, págs. 671-679.

Simpson, Alex (2005). «Reduction in a Linear Lambda-calculus with Applications to Operational Semantics». En: *Proceedings of the 16th International Conference on Term Rewriting and Applications (RTA'05)*. Ed. por Jürgen Giesl. Vol. 3467. Lecture Notes in Computer Science. Springer-Verlag, págs. 219-234.

TeReSe (2003). *Term Rewriting Systems*. Cambridge Tracts in Theoretica. Cambridge University Press.

van Tonder, André (2004). «A Lambda Calculus for Quantum Computation». En: *SIAM Journal on Computing* 33.5, págs. 1109-1135.