UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS


# EL PARADIGMA DE PROGRAMACIÓN CON PRUEBAS

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
LIC. EN CIENCIAS DE LA COMPUTACIÓN
PRESENTA:


GABRIELA BELÉN RAMÍREZ JIMÉNEZ


DIRECTOR DE TESIS:
FAVIO EZEQUIEL MIRANDA PEREA

CIUDAD UNIVERSITARIA, CD. MX., 2025

**Agradecimientos**

El desarrollo del presente trabajo de tesis tuvo lugar en el marco del proyecto

**"Algunos aspectos de la lógica en la topología, la computación y los fundamentos de la matemática"**
**(PAPIIT, IN108810)**

otorgado por la Dirección General de Asuntos del Personal Académico
de la Universidad Nacional Autónoma de México.

Gracias a mis padres María Enriqueta Jiménez Salazar y Héctor Ramírez Luna por su amor y apoyo incondicional. Por haber contribuido en mi formación, por haber creído en mí y alentarme en todo momento.

Gracias a mi hermano Rodrigo Ramírez Jiménez por ser amigo, motivación y alegría.

Gracias al Dr. Favio Ezequiel Miranda Perea por sus enseñanzas, su asesoría, por su comprensión y paciencia, por promover el estudio y la investigación. Sobre todo gracias por las tardes de trabajo compartidas que me permitieron conocerlo mejor como ser humano y así admirarlo más.

A mis abuelos Gloria Salazar Flores y Enrique Jiménez Piña, a mis tíos, primos y sobrinos por su cariño. A Carlos Enrique, Erick Daniel, Dana Michelle, Christopher, Leslie Alejandra, Paola Yamilet, Valentín, Jade Ivanna, Patricio y Elif Luciana por su afecto, por enseñarme que las habilidades se adquieren con procesos graduales y por ser una razón para perseverar.

A el Dr. Samuel Jurado Cárdenas, Dra. Margarita Becerra, Dra. Clauida Becerra, Psi. Lilia Correa Graves, Psi. Jovita González, Dra. Adoración, Dra. Griselda Iris Flores Flores, Dra. Silvia Viridiana Vázquez Jiménez, Dra. Arlette y Dra Brisa por guiarme y escucharme, por fortalecerme, por animarme a enfrentar las situaciones y no desistir.

A mis amigos por todo lo vivido, por su ayuda y por compartir sus conocimientos: Nelly Flores Sánchez, Mariana Zamora Carbajal, Xochiquétzal Hernández, Ulises Román, Adrián Pérez, Víctor Ortiz Othón, Iván David Almada Pérez, Manuel Alcántara Juárez y Jaime Aguilar Reyes.

# Índice general

# Introducción

Las matemáticas y las ciencias de la computación están fuertemente vinculadas sobre todo en el área de la lógica. De hecho, en cierto sentido la lógica ha sido más efectiva en las ciencias de la computación que en las matemáticas, para una discusión de este punto de vista véase el artículo [2]. Muestra de ello son los conceptos y métodos de la lógica que fundamentan la teoría de tipos, la cual es un pilar del área de lenguajes de programación.

La teoría de tipos es un paradigma para el diseño, análisis e implementación de lenguajes de programación. Es útil para entender conceptos como la abstracción de datos, el polimorfismo y la herencia (véase por ejemplo [17]). De ella se desprenden nuevas técnicas para la implementación de compiladores que mejoran la eficiencia y la integridad del código generado. La teoría de tipos es el estudio de los sistemas de tipos. Reynolds define un sistema de tipos como una disciplina sintáctica que refuerza niveles de abstracción. Un sistema de tipos es una forma de gramática sensible al contexto que impone restricciones en la formación de programas para asegurar que no ocurran errores debidos a una interpretación incorrecta de valores. Clasificar los lenguajes de programación de acuerdo a la estructura de sus tipos tiene muchos beneficios, por ejemplo los conceptos del lenguaje pueden presentarse modularmente, evitando así la confusión de distintos conceptos y la estructura de tipos puede ser utilizada para razonar acerca del comportamiento de un programa.

En 1934 Gerhard Gentzen propuso un formalismo lógico conocido como *Deducción Natural*, el cual pretende modelar matemáticamente el razonamiento usual. Alonzo Church ideó el cálculo lambda en 1936, un formalismo para programas con el que demostró que cualquier función computable podía ser representada por un término lambda. Más tarde en 1940 introdujo el cálculo lambda con tipos en el cual extiende la noción de juicios con suposiciones al incluir términos y tipos. En 1956 Drag Prawitz mostró cómo simplificar directamente pruebas de deducción natural. Ese mismo año Curry y Feys publicaron un trabajo sobre combinadores, una variante de cálculo lambda simplificado, en el cual Curry descubrió la correspondencia entre los tipos de los combinadores y las leyes de la lógica formulada por Hilbert. En 1969 W. A. Howard juntó los resultados de Curry y de Prawitz y estableció la correspondencia entre la deducción natural y el cálculo lambda, conocida hoy como correspondencia de Curry-Howard (véase [3]).

La correspondencia de Curry-Howard establece a grandes rasgos que es lo mismo hacer pruebas que hacer programas y que simplificar una prueba corresponde a ejecutar un programa. La importancia de tal hecho traspasa el ámbito teórico al ser el fundamento para la construcción de lenguajes de programación y diversas aplicaciones. Scheme, Ml, Standard ML, Miranda, Haskell y O'Caml son lenguajes basados en cálculo lambda. Lenguajes de bases de datos para la biomedicina, sistemas para el entendimiento de lenguaje natural, demostradores de teoremas, seguridad en sistemas, son algunas de las aplicaciones que tienen como base los trabajos de Curry y Howard. Es evidente el potencial de la correspondencia y por ello existe interés en el uso de ésta para la programación.

Una prueba formal no es exactamente un programa sino que, además de su contenido algorítmico, también tiene partes conceptuales que fundamentan sus resultados. La distinción entre lo algorítmico y lo conceptual puede hacerse a diferentes niveles:

- *Los separadores lógicos.* Los conectivos proposicionales tienen un contenido algortímico mientras que los cuantificadores son partes conceptuales que indican cierto grado de generalidad.
- *La definición de los objetos.* Las definiciones iterativas y recursivas de tipos de datos, si bien son conceptualmente equivalentes, no tienen el mismo contenido algortímico, puesto que corresponden a diferentes maneras de construir y acceder a los datos.
- *Las pruebas.* Las distintas maneras de hacer pruebas, por inducción por ejemplo, pueden contener partes algorítmicas que ayudan al cálculo de resultados y partes conceptuales independientes del resultado.

El propósito principal de este trabajo consiste en presentar al llamado paradigma de programación con pruebas, desarrollado originalmente por Krivine y Parigot en [5], pero definiendo y utilizando una lógica más amigable y comprensible desde el punto de vista de la programación. Al emplear el paradigma con pruebas para programar son tres las tareas a realizar:

- Especificar un problema con fórmulas lógicas, por ejemplo, el problema de obtener la concatenación de dos listas.
- Especificar un algoritmo que resuelva el problema, por ejemplo las ecuaciones que definen recursivamente la función de concatenación.
- Diseñar un programa que implemente al algoritmo anterior, por ejemplo, un programa en Haskell para la función concatenación.

La forma usual de obtener los programas es derivándolos directamente de la especificación lógica del problema. El inconveniente de este método es la extracción de programas (es decir, el procedimiento para obtener un programa formal, que en nuestro caso será un término del cálculo lambda, directa y automáticamente de la prueba lógica) a partir de pruebas cuyas fórmulas involucran cuantificadores existenciales, los cuales son difíciles de interpretar computacionalmente. En contraste, en este trabajo realizamos las tareas anteriores de la siguiente manera:

- Se definen los tipos de datos involucrados en el problema mediante una fórmula.
- Se especifica el algoritmo ecuacionalmente, definiendo una función $f$.
- Utilizando deducción natural y el razonamiento ecuacional, se construye una prueba que muestra que $f$ cumple la especificación del problema.
- A partir de dicha prueba se obtiene de manera automática un término del cálculo lambda, el cual corresponde al programa deseado.

Este trabajo tiene por objetivo definir una lógica que sirva como base para un prototipo de un lenguaje de programación cuyos programas sean pruebas formales y por lo tanto sean correctos por construcción. Dicha lógica, denotada $\mathsf{AF2}^{M\mu\nu}$, es una extensión de la lógica de segundo orden $\mathsf{AF2}$ con operadores primitivos de recursión y correcursión, y cuya semantica subyacente se refiere a la construcción de puntos fijos de operadores monótonos. Así mismo se incluye un mecanismo de definición de ecuaciones que permite especificar nuestros algoritmos directamente.

En el primer capítulo se explica la teoría que fundamenta la lógica propuesta. En el segundo capítulo se discuten la lógica de segundo orden $\mathsf{AF2}$, los sistemas de deducción natural y el cálculo lambda, que son los formalismos sobre los cuales definiremos en el tercer capítulo la lógica propuesta, a la que llamamos $\mathsf{AF2}^{M\mu\nu}$. La utilidad de esta lógica se evidencia en el cuarto capítulo donde mostramos diversos ejemplos de programación con pruebas. Por último se exponen las conclusiones producto del desarrollo del trabajo, asímismo se proponen posibles líneas de trabajo a futuro.

# Capítulo 1

# Preliminares

En este capítulo se expone la teoría que sustenta el presente trabajo. Se explican la teoría de puntos fijos, el teorema de Knaster-Tarski, los principios de inducción y coinducción, los principios de construcción y destrucción de puntos fijos y los principios de Mendler. En el tercer capítulo se hará explícita la relevancia de todos esos conceptos.

## 1.1. Teoría de puntos fijos

Un punto fijo $X$ de una función $\Phi$ es una solución a la ecuación $X = \Phi(X)$, es decir, es un punto que es mapeado a sí mismo bajo la función. Muchas estructuras empleadas en ciencias de la computación se definen por punto fijo, por ejemplo:

- Gramáticas libres de contexto. El lenguaje $b^n a b^n$ se genera mediante

$$\mathbf{G} ::= a \mid b\,\mathbf{G}\,b$$

- Semántica denotativa del ciclo while

$$[\mathbf{while}\ \ \boldsymbol{B}\ \ \mathbf{do}\ \ \boldsymbol{C}]\,\sigma = \begin{cases} [\mathbf{while}\ \ \boldsymbol{B}\ \ \mathbf{do}\ \ \boldsymbol{C}]\,([C]\,\sigma) & si\ \ [B]\,\sigma\ es\ falso \\ \\ \sigma & en\ otro\ caso \end{cases}$$

- Tipos de datos recursivos

$$data\ \mathbf{List\ a} = Nil \mid Cons\ a\ \ (\mathbf{List\ a})$$

Es fácil ver que estos ejemplos son casos particulares de una ecuación de la forma $X = \Phi(X)$. Si bien una ecuación de esta forma siempre puede plantearse, es importante estudiar en qué casos tiene solución, dichos casos corresponden a definiciones recursivas válidas. A continuación desarrollamos la teoría que garantiza las soluciones a estas ecuaciones de punto fijo.

Utilizaremos la teoría de órdenes parciales y latices para justificar nuestras definiciones. Empezamos recordando qué es un orden parcial.

**Definición 1.1** (Orden parcial)**.** Un orden parcial sobre un conjunto A es una relación binaria $\sqsubseteq$ sobre A que cumple

- Reflexividad: $\forall x.(x \sqsubseteq x)$
- Transitividad: $\forall x.\forall y.\forall z.(x \sqsubseteq y \ \wedge \ y \sqsubseteq z \ \rightarrow \ x \sqsubseteq z)$
- Antisimetría: $\forall x.\forall y.(x \sqsubseteq y \ \wedge \ y \sqsubseteq x \ \rightarrow \ x = y)$

Una vez definido un orden parcial existen ciertos elementos de interés llamados elementos extremos. En particular nos interesan los siguientes.

**Definición 1.2.** [Cotas inferiores y superiores] Sean $A$ y $B$ dos conjuntos tales que $B \subseteq A$ y $c \in A$.

- Decimos que c es cota superior de B si y sólo si $\forall x \in B \ (x \sqsubseteq c)$
- Decimos que c es cota inferior de B si y sólo si $\forall x \in B \ (c \sqsubseteq x)$

Dentro de las cotas superiores e inferiores de un conjunto $B$ son de importancia aquellas llamadas supremo e ínfimo, cuya definición es la siguiente.

**Definición 1.3.** [Supremos e ínfimos] Sean $A$ y $B$ dos conjuntos tales que $B \subseteq A$ y $s, i, c \in A$.

- Decimos que $s$ es el supremo de B si y sólo si se cumplen las siguientes dos condiciones:
  - $\forall x \in B \ (x \sqsubseteq s)$
  - $\forall c \in A \ (\forall x \in B \ (x \sqsubseteq c) \rightarrow s \sqsubseteq c)$

  Estas dos condiciones definen a $s$ como la mínima cota superior de $B$. En caso de que dicha cota exista es única y escribimos $s = \bigsqcup B$.

- Decimos que $i$ es el ínfimo de B si y sólo si se cumplen las siguientes dos condiciones:
  - $\forall x \in B \ (i \sqsubseteq x)$
  - $\forall c \in A \ (\forall x \in B \ (c \sqsubseteq x) \rightarrow c \sqsubseteq i)$

  Tales condiciones definen a $i$ como la máxima cota inferior de $B$. En caso de que dicha cota exista es única y escribimos $i = \bigsqcap B$.

Si $B$ es un conjunto de dos elementos $B = \{x, y\}$, escribimos $x \sqcup y$ en vez de $\sqcup\{x, y\}$ y similarmente $x \sqcap y$ significa $\sqcap\{x, y\}$.

Los elementos supremo e ínfimo de un conjunto no tienen porque existir en cualquier orden parcial.Los órdenes que nos interesan son aquellos para los que sí existen, los cuales se conocen como retículas o latices.

**Definición 1.4.** [Latiz] Sea $\langle \mathcal{L}, \sqsubseteq \rangle$ un conjunto parcialmente ordenado. $\mathcal{L}$ es una latiz si y sólo si cualesquiera dos elementos en $\mathcal{L}$ tienen supremo e ínfimo en $\mathcal{L}$, es decir si:

- $\forall x, \forall y \in \mathcal{L}\ (x \sqcup y \in \mathcal{L})$
- $\forall x, \forall y \in \mathcal{L}\ (x \sqcap y \in \mathcal{L})$

**Definición 1.5.** [Latiz completa] Una latiz $\langle \mathcal{L}, \sqsubseteq \rangle$ es completa si y sólo si cualquier subconjunto de $\mathcal{L}$ tiene supremo e ínfimo en $\mathcal{L}$. Esto es, para cualquier subconjunto $M \subseteq \mathcal{L}$ se cumple:

- $\bigsqcup M \in \mathcal{L}$
- $\bigsqcap M \in \mathcal{L}$

El ejemplo clásico de latiz completa es el orden parcial dado por la contención de conjuntos sobre la potencia de un conjunto dado, es decir el orden $\langle \mathcal{P}(A), \subseteq \rangle$. En este caso el supremo es la unión y el ínfimo es la intersección de conjuntos.

Las ecuaciones de punto fijo $X = \Phi(X)$ que nos interesa resolver serán aquellas en donde $\Phi : \mathcal{L} \to \mathcal{L}$ es un operador en una latiz. Esta clase de ecuaciones no siempre tienen solución. Sin embargo, en el caso en que $\Phi$ sea un operador monótono las soluciones existen. El concepto de monotonía al que nos referimos es el siguente

**Definición 1.6.** Sean $\langle \mathcal{L}, \sqsubseteq \rangle$ una latiz y $\Phi : \mathcal{L} \to \mathcal{L}$ un operador. Decimos que $\Phi$ es monótono si y solo sí para cualesquiera $X, Y \in \mathcal{L}$, si $X \sqsubseteq Y$ entonces $\Phi(X) \sqsubseteq \Phi(Y)$.

A continuación presentaremos los conceptos necesarios para definir y garantizar la existencia de soluciones para ecuaciones de punto fijo sobre operadores monótonos.

**Definición 1.7.** Sean $\langle \mathcal{L}, \sqsubseteq \rangle$ una latiz y $\Phi : \mathcal{L} \to \mathcal{L}$ un operador monótono. Se dice que $X \in \mathcal{L}$ es:

- $\Phi$-cerrado o punto prefijo de $\Phi$ si $\Phi(X) \sqsubseteq X$
- $\Phi$-denso o punto postfijo de $\Phi$ si $X \sqsubseteq \Phi(X)$
- $\Phi$-inductivo si es menor que cada punto prefijo de $\Phi$, es decir, si $\Phi(Y) \sqsubseteq Y$ implica $X \sqsubseteq Y$
- $\Phi$-coinductivo si es mayor que cada punto postfijo de $\Phi$, es decir, si $Y \sqsubseteq \Phi(Y)$ implica $Y \sqsubseteq X$

Veamos algunas propiedades que involucran a estos conceptos.

**Proposición 1.1.** Sea $\Phi : \mathcal{L} \to \mathcal{L}$ un operador monótono. Entonces:

i. $\Phi$ tiene a lo más un punto prefijo inductivo.
ii. $\Phi$ tiene a lo más un punto postfijo coinductivo.
iii. Los puntos prefijos inductivos de $\Phi$ son puntos fijos.

IV. Los puntos postfijos coinductivos de $\Phi$ son puntos fijos

*Demostración.* I. Supóngase que existen dos puntos prefijos inductivos $X_1$ y $X_2$, habrá que demostrar que son el mismo. Dado que $X_2$ es prefijo, $\Phi(X_2) \sqsubseteq X_2$ y puesto que $X_1$ es inductivo, se puede concluir que $X_1 \sqsubseteq X_2$. Análogamente, como $X_1$ es prefijo, $\Phi(X_1) \sqsubseteq X_1$ y ya que $X_2$ es inductivo, $X_2 \sqsubseteq X_1$. Por lo tanto $X_1 = X_2$, es decir a lo más hay un punto prefijo inductivo.

II. La demostración para este caso es análoga a la anterior.

III. Sea $X$ un punto prefijo e inductivo. Como $X$ es prefijo, se tiene $\Phi(X) \sqsubseteq X$. Por otro lado como $X$ es prefijo y $\Phi$ es monótono se sigue que $\Phi(\Phi(X)) \sqsubseteq \Phi(X)$, es decir $\Phi(X)$ es punto prefijo y dado que $X$ es inductivo se tiene $X \sqsubseteq \Phi(X)$. Así se concluye que $X = \Phi(X)$.

IV. La demostración para tal caso es semejante a la anterior.

□

**Proposición 1.2.** Sea $\Phi : \mathcal{L} \to \mathcal{L}$ un operador monótono. Entonces:

I. Si $X_i$ es una colección de puntos prefijos de $\Phi$ entonces el ínfimo $\bigsqcap_i X_i$ también lo es.

II. Si $X_i$ es una colección de puntos postfijos de $\Phi$ entonces el supremo $\bigsqcup_i X_i$ también lo es.

*Demostración.* I. Supóngase que $X_i$ es prefijo para toda i, es decir $\Phi(X_i) \sqsubseteq X_i$. Por definición de ínfimo $\bigsqcap_i \Phi(X_i) \sqsubseteq \bigsqcap_i X_i$. Puesto que $\Phi$ es monótono y dado que $\bigsqcap_i X_i \sqsubseteq X_i$ para toda i, se sigue que $\Phi(\bigsqcap_i X_i) \sqsubseteq \Phi(X_i)$. Notese que $\Phi(\bigsqcap_i X_i)$ es cota inferior de $\Phi(X_i)$ y ya que $\bigsqcap_i \Phi(X_i)$ es la máxima cota inferior se tiene que $\Phi(\bigsqcap_i X_i) \sqsubseteq \bigsqcap_i \Phi(X_i)$. Por transitividad se concluye $\Phi(\bigsqcap_i X_i) \sqsubseteq \bigsqcap_i X_i$ lo cual significa que $\bigsqcap_i X_i$ es prefijo.

II. La demostración es análoga a la anterior.

□

A continuación presentamos el famoso teorema de Knaster-Tarski, el cual garantiza la existencia de puntos fijos para operadores monótonos.

**Teorema 1.1** (Knaster-Tarski)**.** Sean $\langle \mathcal{L}, \sqsubseteq \rangle$ una latiz completa y $\Phi : \mathcal{L} \to \mathcal{L}$ un operador monótono. Entonces:

- $\Phi$ tiene un mínimo punto fijo, denotado $\mu(\Phi)$, dado por $\mu(\Phi) = \bigsqcap \{X \in \mathcal{L} | \Phi(X) \sqsubseteq X\}$
- $\Phi$ tiene un máximo punto fijo, denotado $\nu(\Phi)$, dado por $\nu(\Phi) = \bigsqcup \{X \in \mathcal{L} | X \sqsubseteq \Phi(X)\}$

*Demostración.* Para el primer caso se demuestra primero que $\mu(\Phi)$ es punto fijo y después que es el mínimo.

a. Haciendo uso de la proposición 1.1, basta demostrar que $\mu(\Phi)$ es prefijo e inductivo. Que $\mu(\Phi)$ es prefijo es una consecuencia de la proposición 1.2. Por otra parte $\mu(\Phi)$ está por debajo de cada punto prefijo de $\Phi$, pero esto es precisamente la definición de que $\mu(\Phi)$ sea inductivo.

b. Puesto que $\mu(\Phi)$ es inductivo, se tiene que para cualquier $X$, si $\Phi(X) \sqsubseteq X$ entonces $\mu(\Phi) \sqsubseteq X$. En particular, si $Y$ es un punto fijo también es prefijo y por lo tanto $\mu(\Phi) \sqsubseteq Y$.

La demostración del caso $\nu(\Phi)$ tiene un argumento dual al anterior.

$\square$

Nos interesa definir tipos de datos mediante una ecuación de punto fijo, de manera que dichos tipos de datos existirán cuando la solución correspondiente a tal ecuación exista. Veamos un par de ejemplos.

En teoría de conjuntos, los números naturales se definen como la intersección de todos aquellos conjuntos que tienen al cero y son cerrados bajo la función sucesor, denotada *suc*. Esta idea corresponde a la definición de punto fijo: $\Phi(X) = \{0\} \cup X^{suc}$, donde $X^{suc} = \{suc(x) | x \in X\}$.

$\Phi$ así definido es monótono, para comprobarlo suponemos $X \subseteq Y$. Sea $p \in \Phi(X)$, esto es, $p \in \{0\} \cup X^{suc}$. Si $p \in \{0\}$, entonces $p \in \Phi(Y)$. En caso de que $p \in X^{suc}$, $p$ es de la forma $p = suc(q)$ tal que $q \in X$. Por hipótesis $q \in Y$ y de acuerdo a la definición de $Y^{suc}$, $suc(q) \in Y$, o lo que es lo mismo $p \in Y^{suc}$. De los casos anteriores podemos concluir que $p \in \{0\} \cup Y^{suc}$, lo cual se puede reescribir como $p \in \Phi(Y)$, por lo tanto $\Phi$ es monótono.

Dado que $\Phi$ es monótono tiene un mínimo punto fijo, cuya definición, de acuerdo al teorema de Knaster-Tarski es:

$$\mu(\Phi) = \bigcap \{X \in \mathcal{L} | \Phi(X) \subseteq X\}$$

Por otra parte, utilizando la definición particular de $\Phi$ se tiene que:

$$\mu(\Phi) = \bigcap \{X \in \mathcal{L} | \, \{0\} \cup X^{suc} \subseteq X\}$$

Pero esta última corresponde a la definición conjuntista de los números naturales, por lo tanto $\mathbb{N} = \mu(\Phi)$.

Del mismo modo se definen las listas sobre un conjunto $A$ como la intersección de todas las listas que tienen a la lista vacía *nil* y son cerradas bajo la función constructora de listas denotada *cons*. La construcción mediante una ecuación de punto fijo utiliza el operador monótono $\Phi(X) = \{nil\} \cup X^{cons}$, donde $X^{cons} = \{cons(x, y) | x \in A \wedge y \in X\}$. El conjunto de las listas sobre $A$ es entonces el mínimo punto fijo de $\Phi$:

$$\mu(\Phi) = \bigcap \{X \in \mathcal{L} | \Phi(X) \subseteq X\}$$

Conforme a la definición anterior de $\Phi$ se tiene:

$$\mu(\Phi) = \bigcap \{X \in \mathcal{L} | \, \{nil\} \cup X^{cons} \subseteq X\}$$

Y dado que esta última corresponde a la definición conjuntista de listas, $\mathbb{L}_{\mathcal{A}} = \mu(\Phi)$.

Otro tipo de datos que se puede definir mediante una ecuación de punto fijo son las listas infinitas o streams sobre $A$, que se representan como $\mathbb{S}_{\mathcal{A}} = \{[a_1, a_2, \ldots, a_n, \ldots] \, | a_i \in A\}$. Dado que cada elemento del tipo $\mathbb{S}_{\mathcal{A}}$ es infinito no es posible dar una definición constructiva de $\mathbb{S}_{\mathcal{A}}$. En su lugar se utilizan funciones destructoras para analizar cada uno de sus elementos, separándolos en dos componentes. Esto es, dado un stream $s = [a_1, a_2, \ldots, a_n, \ldots]$, las funciones destructoras *head* y *tail* nos permiten separar su cabeza y su cola, a saber $head(s) = a_1$ y $tail(s) = [a_2, \ldots]$. De esta manera podemos manipular el primer elemento del stream $s$, así como su cola la cual es nuevamente una lista infinita.

Para justificar la existencia de $\mathbb{S}_{\mathcal{A}}$ es necesario emplear una idea dual a la definición inductiva, utilizando las funciones destructoras recién definidas el conjunto $\mathbb{S}_{\mathcal{A}}$ será el más grande posible $X$ cuyos elementos puedan ser destruidos en una cabeza que pertenece a A y una cola que sigue perteneciendo a $X$. El planteamiento anterior se expresa en términos de una ecuación de punto fijo. Sea $\Phi(X)$ el operador definido como $\Phi(X) = A^{head} \cap X^{tail}$, donde $A^{head} = \{x \in X | head(x) \in A\}$ y $X^{tail} = \{x \in X | tail(x) \in X\}$. Es fácil ver que $\Phi$ es monótono y por lo tanto el conjunto de los streams sobre A se define como el máximo punto fijo de $\Phi$:

$$\nu(\Phi) = \bigcup \{X \in \mathcal{L} | X \subseteq A^{head} \cap X^{tail}\}$$

Esta definición captura la idea anterior, por lo cual $\mathbb{S}_{\mathcal{A}} = \nu(\Phi)$.

El teorema de Knaster-Tarski no solo es útil para saber si un operador tiene un punto fijo, sino que también permite obtener los principios de inducción y coinducción como corolario. Estos principios son indispensables para verificar propiedades de las definiciones de punto fijo.

**Corolario 1.** *Sea $\Phi : \mathcal{L} \to \mathcal{L}$ un operador monótono en una latiz completa $\mathcal{L}$. Entonces para cualquier $M \in \mathcal{L}$ se cumplen los siguientes principios:*

- *Inducción: Si $\Phi(M) \sqsubseteq M$ entonces $\mu(\Phi) \sqsubseteq M$*
- *Coinducción: Si $M \sqsubseteq \Phi(M)$ entonces $M \sqsubseteq \nu(\Phi)$*

*Demostración.* La demostración se divide en dos casos.

- Inducción: de la definición de $\mu(\Phi)$ es inmediato que si $\Phi(M) \sqsubseteq M$ entonces $\mu(\Phi) \sqsubseteq M$.
- Coinducción: análogo al caso anterior.

□

Los principios enunciados en este corolario se conocen como los principios de inducción y coinducción convencionales.

A continuación vemos cómo obtener los principios usuales de inducción para números naturales y listas a partir de los principios dados en el corolario anterior. El principio de inducción usual para números naturales es el siguiente:

$$\forall X(0 \in X \ \rightarrow \ \forall x(x \in X \rightarrow suc(x) \in X) \ \rightarrow \ \mathbb{N} \subseteq X)$$

Éste se obtiene del principio general como sigue, recordando que el orden es la contención y el ínfimo es la intersección:

$$\begin{array}{l}
\forall X(\Phi(X) \subseteq X \rightarrow \mu(\Phi) \subseteq X) \\
\forall X(\{0\} \cup X^{suc} \subseteq X \rightarrow \mathbb{N} \subseteq X) \\
\forall X(0 \in X \ \rightarrow \ \forall x(x \in X^{suc} \rightarrow x \in X) \ \rightarrow \mathbb{N} \subseteq X) \\
\forall X(0 \in X \ \rightarrow \ \forall x(x \in X \rightarrow suc(x) \in X) \ \rightarrow \mathbb{N} \subseteq X)
\end{array}$$

El principio de inducción para listas sobre un conjunto $A$ es:

$$\forall X(nil \in X \ \rightarrow \ \forall a \forall l(a \in A \wedge l \in X \rightarrow cons(a,l) \in X) \ \rightarrow \ \mathbb{L}_{\mathcal{A}} \subseteq X)$$

Éste también se obtiene del principio general de inducción como sigue:

$$\begin{array}{l}
\forall X(\Phi(X) \subseteq X \rightarrow \mu(\Phi) \subseteq X) \\
\forall X(\{nil\} \cup X^{cons} \subseteq X \rightarrow \mathbb{L}_{\mathcal{A}} \subseteq X) \\
\forall X(nil \in X \ \rightarrow \ \forall x(x \in X^{cons} \rightarrow x \in X) \ \rightarrow \mathbb{L}_{\mathcal{A}} \subseteq X) \\
\forall X(0 \in X \ \rightarrow \ \forall a \forall l(a \in A \wedge l \in X \ \rightarrow cons(a,l) \in X) \ \rightarrow \mathbb{L}_{\mathcal{A}} \subseteq X)
\end{array}$$

Por otra parte, también es posible obtener el principio de coinducción para streams de $A$ a partir del colorario antes mencionado. El principio de coinducción para streams sobre un conjunto $A$ es

$$\forall X(X \subseteq A^{head} \cap X^{tail} \rightarrow X \subseteq \mathbb{S}_{\mathcal{A}})$$

Mismo que se obtiene a partir del principio general de coinducción, utilizando la definición particular de $\Phi$ y siendo el orden la contención:

$$\begin{array}{l}
\forall X(X \subseteq \Phi(X) \rightarrow X \subseteq \nu(\Phi)) \\
\forall X(X \subseteq A^{head} \cap X^{tail} \rightarrow X \subseteq \mathbb{S}_{\mathcal{A}})
\end{array}$$

El uso del principio de coinducción se muestra en el siguiente ejemplo de construcción [1] de listas infinitas, en el cual la notación $\ell = (x : s)$ corresponde a lista infinita $\ell$ tal que $head\ \ell = x$ y $tail\ \ell = s$. Es importante observar que una lista infinita no se puede construir por recursión, siendo el principio de coinducción hasta el momento la única herramienta para construirlas.

Sean $\mathcal{L}_1$ y $\mathcal{L}_2$ listas definidas mediante las siguientes ecuaciones:

[1] De este ejemplo se observa que si bien no es posible generar mediante un proceso de construcción al tipo de listas infinitas, razón por la cual se utilizan las funciones destructoras, sí es factible construir ciertas listas infinitas.

$$\begin{aligned}\mathcal{L}_1 &= \{0 \ : \ \mathcal{L}_2\} \\ \mathcal{L}_2 &= \{1 \ : \ \mathcal{L}_1\}\end{aligned}$$

Intuitivamente estas ecuaciones definen a las siguientes listas infinitas:

$$\mathcal{L}_1 = \{0, 1, 0, 1, \ldots\}$$
$$\mathcal{L}_2 = \{1, 0, 1, 0, \ldots\}$$

De entrada no es claro que esta sea una construcción válida de listas infinitas. Para justificar su pertinencia apelamos al principio de coinducción de la siguiente manera.

Sea $R = \{\mathcal{L}_1 \ , \ \mathcal{L}_2\}$ y sea $\mathcal{S}_{\{0,1\}}$ el conjunto de listas estrictamente infinitas de ceros y unos. El caso particular del principio de coinducción para $R$ se expresa como:

$$\text{Si} \quad R \subseteq \{0,1\}^{head} \cap R^{tail} \quad \text{entonces} \quad R \subseteq \mathcal{S}_{\{0,1\}}$$

De manera que para ver que $R$ es un conjunto bien definido de listas infinitas, es suficiente con mostrar que $R \subseteq \{0,1\}^{head} \cap R^{tail}$.

- $R \subseteq \{0,1\}^{head}$ puesto que $\{0,1\}^{head} \ = \ \{x \in R \mid \mathit{head}\ x \in \{0,1\}\} \ = \ R.$

- $R \subseteq R^{tail}$ puesto que $R^{tail} \ = \ \{x \in R \mid \mathit{tail}\ x \in R\} = \ R.$

De modo que, por el principio de coinducción $R \subseteq \mathcal{S}_{\{0,1\}}$.

Los principios de inducción y coinducción recién presentados serán de gran utilidad posteriormente.

## 1.2. Principios de inducción y coinducción extendidos

Los principios de inducción y coinducción convencionales resultan demasiados débiles para probar ciertas propiedades, veamos un ejemplo.

Sea $\varphi : \mathbb{R} \to \mathbb{R}$ tal que $\varphi(x) = x^2$. Queremos demostrar que $\forall x \in \mathbb{N}(\varphi(x) \in \mathbb{N})$, es decir que el cuadrado de un natural también es un número natural. De acuerdo al axioma de inducción para naturales con $X := \left\{x \in \mathbb{R} | x^2 \in \mathbb{N}\right\}$, hay que demostrar que $0 \in X$ y que $\forall x(x \in X \to suc(x) \in X)$:

- $0 \in X$ pues $0^2 = 0 \in \mathbb{N}$
- Hipótesis de inducción: supóngase que $n \in X$, es decir $n^2 \in \mathbb{N}$.
- Se debe demostrar que $(n+1)^2 \in \mathbb{N}$, es decir que $n^2 + 2n + 1 \in \mathbb{N}$.

- Por hipótesis de inducción $n^2 \in \mathbb{N}$, por lo que $n^2 + 1 \in \mathbb{N}$.
- Falta demostrar que $2n \in \mathbb{N}$, para lo cual hay que ver que $n \in \mathbb{N}$. Pero ésto no es posible demostrar pues de $n$ solo se sabe que $n \in X$.

De la demostración anterior se puede ver que el principio de inducción no es suficiente para demostrar la propiedad, ya que es necesaria la suposición $n \in X \cap \mathbb{N}$. Es por eso que existe un principio de inducción convencional extendido que incluye en la hipótesis la pertenencia al conjunto inductivo correspondiente, en este caso los números naturales. Lo mismo ocurre para el principio de coinducción convencional. Estos principios también son consecuencia del teorema de Knaster-Tarski.

**Corolario 2.** *Sea $\Phi : \mathcal{L} \to \mathcal{L}$ un operador monótono en una latiz completa $\mathcal{L}$. Entonces para cualquier $M \in \mathcal{L}$ se cumplen los siguientes principios:*

- *Inducción extendida: Si $\Phi(\mu(\Phi) \bigsqcap M) \sqsubseteq M$ entonces $\mu(\Phi) \sqsubseteq M$*
- *Coinducción extendida: Si $M \sqsubseteq \Phi(\nu(\Phi) \bigsqcup M)$ entonces $M \sqsubseteq \nu(\Phi)$*

*Demostración.* Se demuestra el caso de la coinducción extendida, para el de inducción se sigue un argumento dual al presentado a continuación. Se define $M'$ como $M' = \nu(\Phi) \bigsqcup M$, de manera que la hipótesis es $M \sqsubseteq \Phi(M')$. Hay que demostrar que $M \sqsubseteq \nu(\Phi)$. Puesto que claramente $M \sqsubseteq M'$, basta ver que $M' \sqsubseteq \nu(\Phi)$. Esto se hará mediante el principio de coinducción convencional, según el cual basta mostrar que $M' \sqsubseteq \Phi(M')$. Por la definición de $M'$ se cumple que $\nu(\Phi) \sqsubseteq M'$ y como $\Phi$ es un operador monótono entonces $\Phi(\nu(\Phi)) \sqsubseteq \Phi(M')$, lo cual es lo mismo que $\nu(\Phi) \sqsubseteq \Phi(M')$. De lo anterior y de la hipótesis se sigue finalmente que $\nu(\Phi) \bigsqcup M \sqsubseteq M'$, es decir $M' \sqsubseteq \Phi(M')$.

□

En el caso particular del operador $\Phi$ que define a los números naturales se obtiene el siguiente principio de inducción extendida:

$$\forall X(0 \in X \ \to \ \forall x(x \in \mathbb{N} \cap X \to suc(x) \in X) \ \to \ \mathbb{N} \subseteq X)$$

el cual se obtiene del principio de inducción general como sigue:

$$\begin{array}{l}
\forall X(\Phi(\mu(\Phi) \cap X) \subseteq X \to \mu(\Phi) \subseteq X) \\
\forall X(\Phi(\mathbb{N} \cap X) \subseteq X \to \mathbb{N} \subseteq X) \\
\forall X(\{0\} \cup (\mathbb{N} \cap X)^{suc} \subseteq X \to \mathbb{N} \subseteq X) \\
\forall X(0 \in X \land (\mathbb{N} \cap X)^{suc} \subseteq X \to \mathbb{N} \subseteq X) \\
\forall X(0 \in X \land \forall x(x \in (\mathbb{N} \cap X)^{suc} \to x \in X) \ \to \mathbb{N} \subseteq X) \\
\forall X(0 \in X \land \forall x(x \in (\mathbb{N} \cap X) \to suc(x) \in X) \ \to \mathbb{N} \subseteq X) \\
\forall X(0 \in X \ \to \ \forall x(x \in (\mathbb{N} \cap X) \to suc(x) \in X) \ \to \mathbb{N} \subseteq X)
\end{array}$$

Con el axioma de inducción extendido ya es posible demostrar la propiedad anterior como sigue. Sea $\varphi(x) = x^2$ con $\varphi : \mathbb{R} \to \mathbb{R}$. Sea $X := \{x \in \mathbb{R} | x^2 \in \mathbb{N}\}$, hay que demostrar que $0 \in X$ y que $\forall x(x \in \mathbb{N} \cap X \to suc(x) \in X)$ para concluir que $\forall x \in \mathbb{N}(\varphi(x) \in \mathbb{N})$. La prueba es como sigue:

- $0 \in X$ pues $0^2 = 0 \in \mathbb{N}$
- Hipótesis de inducción: supóngase que $n \in \mathbb{N} \cap X$, es decir que $n \in \mathbb{N}$ y $n^2 \in \mathbb{N}$
- Se debe demostrar que $(n+1)^2 \in \mathbb{N}$, es decir que $n^2 + 2n + 1 \in \mathbb{N}$.
    - Dado que por hipótesis de inducción $n^2$ y $n \in \mathbb{N}$, se tiene que $n^2 + 2n + 1 \in \mathbb{N}$

Los principios de (co)inducción nos serán de utilidad más adelante.

## 1.3. Monotonización de un operador

De la discusión anterior se observa que una vez definido un tipo de datos como los número naturales o las listas, mediante un operador monótono $\Phi$, el teorema de Knaster-Tarski no solo garantiza la coherencia de dicha definición sino que tambien proporciona principios de inducción para el tipo correspondiente, los cuales son indispensables para demostrar propiedades del tipo en cuestión.

Con los principios de inducción y coinducción es posible definir tipos de datos $\mu(\Phi)$ y $\nu(\Phi)$ siempre que $\Phi$ sea monótono. Sin embargo, existen tipos de datos cuyo operador de definición $\Phi$ no es monótono, por ejemplo *data Expr* $= \mu(\Phi)$ donde

$$\Phi(X) = (X \to X) \cup (X \ \times \ X) \cup String$$

Esta ecuación de punto fijo corresponde a la definición de un tipo de datos para expresiones del cálculo $\lambda$ (sección 2.5), el cuál se define en programación funcional como:

$$data\ Expr \ = \ Lam\ (Expr- > Expr) \mid App\ Expr\ Expr \mid Var\ String$$

En casos como éste no es posible utilizar la teoría de puntos fijos recien desarrollada para garantizar la coherencia de la definición. Sin embargo sí es posible fundamentar su existencia, definiendo un operador a partir de $\Phi$ el cual sí será monótono.

**Definición 1.8.** [Monotonización de un operador] Sea $\Phi : \mathcal{L} \to \mathcal{L}$ un operador, donde $\langle \mathcal{L}, \bigsqcap \rangle$ es una latiz completa. Hay dos formas de definir un operador monótono a partir de $\Phi$.

- La monotonización superior $\Phi^{\sqsupseteq} : \mathcal{L} \to \mathcal{L}$ se define como:

  $$\Phi^{\sqsupseteq}(M) = \bigsqcup \{\Phi(X) | X \sqsubseteq M\}$$

  Es decir, $\Phi^{\sqsupseteq}(M)$ es el supremo de todas aquellas $\Phi(X) \in \mathcal{L}$ tales que $X \sqsubseteq M$
- La monotonización inferior $\Phi^{\sqsubseteq} : \mathcal{L} \to \mathcal{L}$, definida como:

  $$\Phi^{\sqsubseteq}(M) = \bigsqcap \{\Phi(X) | M \sqsubseteq X\}$$

Es decir, $\Phi^{\sqsubseteq}(M)$ es el ínfimo de todas aquellas $\Phi(X) \in \mathcal{L}$ tales que $M \sqsubseteq X$

Veamos ahora algunas propiedades relevantes de la monotonización.

**Proposición 1.3.** Sea $\Phi : \mathcal{L} \to \mathcal{L}$ un operador, con $\langle \mathcal{L}, \bigsqcap \rangle$ una latiz completa. Se cumplen las siguientes propiedades:

1. $\Phi^{\sqsubseteq}$ y $\Phi^{\sqsupseteq}$ son monótonos.
2. Para cualquier $M \in \mathcal{L}$, $\quad \Phi^{\sqsubseteq}(M) \sqsubseteq \Phi(M) \sqsubseteq \Phi^{\sqsupseteq}(M)$.
3. Si $\Phi$ es monótono entonces $\Phi^{\sqsubseteq} = \Phi = \Phi^{\sqsupseteq}$.
4. Si $\Phi^{\sqsubseteq} = \Phi$ o $\Phi = \Phi^{\sqsupseteq}$ entonces $\Phi$ es monótono.
5. $\Phi(\mu(\Phi^{\sqsupseteq})) \sqsubseteq \mu(\Phi^{\sqsupseteq})$. Es decir, el mínimo punto fijo de la monotonización superior es punto prefijo.
6. $\nu(\Phi^{\sqsubseteq}) \sqsubseteq \Phi(\nu(\Phi^{\sqsubseteq}))$. Es decir, el máximo punto fijo de la monotonización inferior es punto postfijo.

*Demostración.*

1. Para demostrar que $\Phi^{\sqsubseteq}$ es monótono, supongamos $M \sqsubseteq N$. Necesitamos ver que $\Phi^{\sqsubseteq}(M) \sqsubseteq \Phi^{\sqsubseteq}(N)$. Es claro que $\{\Phi(X) | N \sqsubseteq X\} \subseteq \{\Phi(X) | M \sqsubseteq X\}$, de donde por propiedades de monotonía del ínfimo se sigue que $\Phi^{\sqsubseteq}(M) \sqsubseteq \Phi^{\sqsubseteq}(N)$.

   La demostración para el caso $\Phi^{\sqsupseteq}$ es análoga.
2. Dado que $\Phi^{\sqsubseteq}(M)$ es una cota inferior de $\{\Phi(X) | M \sqsubseteq X\}$ y $\Phi(M) \in \{\Phi(X) | x \sqsubseteq M\}$ se tiene entonces que $\Phi^{\sqsubseteq}(M) \sqsubseteq \Phi(M)$. Análogamente se demuestra que $\Phi(M) \sqsubseteq \Phi^{\sqsupseteq}(M)$.
3. Sea $M \in \mathcal{L}$. Si $\Phi$ es monótono entonces $\Phi(M) \sqsubseteq \Phi(X)$ para cualquier $X$ tal que $M \sqsubseteq X$ lo cual implica que $\Phi(M) \sqsubseteq \Phi^{\sqsubseteq}(M)$. Pero por la propiedad anterior $\Phi^{\sqsubseteq}(M) \sqsubseteq \Phi(M)$. Por lo tanto $\Phi^{\sqsubseteq}(M) = \Phi(M)$ para cualquier $M \in \mathcal{L}$, lo que implica que $\Phi^{\sqsubseteq} = \Phi$. La demostración para la igualdad $\Phi = \Phi^{\sqsupseteq}$ es análoga.
4. Se sigue de la parte 1.
5. Se puede ver que $\Phi^{\sqsupseteq}(\mu(\Phi^{\sqsupseteq})) \sqsubseteq \mu(\Phi^{\sqsupseteq})$, es decir, $\bigsqcup \left\{\Phi(X) | X \sqsubseteq \mu(\Phi^{\sqsupseteq})\right\} \sqsubseteq \mu(\Phi^{\sqsupseteq})$. En particular $\Phi(\mu(\Phi^{\sqsupseteq})) \sqsubseteq \mu(\Phi^{\sqsupseteq})$, ya que $\mu(\Phi^{\sqsupseteq}) \sqsubseteq \mu(\Phi^{\sqsupseteq})$.
6. Es análogo a la parte 5.

$\square$

La monotonización de un operador y sus propiedades son el fundamento para justificar definiciones inductivas cuyo operador de definición no sea monótono, como en el ejemplo del tipo *Expr* dado arriba. Mas aún, a pesar de que el operador no sea monótono podemos definir principios de inducción y coinducción, los cuales se conocen como principios de Mendler ([9, 10]).

**Proposición 1.4.** [Principios de inducción y coinducción de Mendler] Sean $\Phi : \mathcal{L} \to \mathcal{L}$ y $M \in \mathcal{L}$.

- Inducción: si $\forall X (X \sqsubseteq M \to \Phi(X) \sqsubseteq M)$ entonces $\mu(\Phi^{\sqsupseteq}) \sqsubseteq M$
- Inducción extendida: si $\forall X (X \sqsubseteq \mu(\Phi^{\sqsupseteq}) \to X \sqsubseteq M \to \Phi(X) \sqsubseteq M)$ entonces $\mu(\Phi^{\sqsupseteq}) \sqsubseteq M$

- Coinducción: si $\forall X(M \sqsubseteq X \to M \sqsubseteq \Phi(X))$ entonces $M \sqsubseteq \nu(\Phi^{\sqsubseteq})$
- Coinducción extendida: si $\forall X(\nu(\Phi^{\sqsubseteq}) \sqsubseteq X \to M \sqsubseteq X \to M \sqsubseteq \Phi(X))$ entonces $M \sqsubseteq \nu(\Phi^{\sqsubseteq})$

*Demostración.* Las demostraciones utilizan los principios de (co)inducción convencionales para $\Phi^{\sqsubseteq}$ y $\Phi^{\sqsupseteq}$. A continuación se demuestra el caso del principio de coinducción extendido de Mendler.

Supóngase $\forall X(\nu(\Phi^{\sqsubseteq}) \sqsubseteq X \to M \sqsubseteq X \to M \sqsubseteq \Phi(X))$, entonces $M \sqsubseteq \Phi(X)$ para cualquier X tal que $\nu(\Phi^{\sqsubseteq}) \bigsqcup M \sqsubseteq X$. A partir de ello se tiene $M \sqsubseteq \Phi^{\sqsubseteq}(\nu(\Phi^{\sqsubseteq}) \bigsqcup M)$ y aplicando el principio de co-inducción extendido para $\Phi^{\sqsubseteq}$ se obtiene $M \sqsubseteq \nu(\Phi^{\sqsubseteq})$.

□

Las definiciones y propiedades recién presentadas son útiles aún cuando el operador $\Phi$ sea monótono. Si se define $\Phi(X) = \{0\} \cup X^{suc}$, el conjunto de los números naturales es $\mathbb{N} = \mu(\Phi)$, pero como en este caso $\Phi$ es monótono, entonces $\Phi = \Phi^{\sqsupseteq}$, por lo que es posible obtener el axioma de inducción de Mendler para números naturales:

$$0 \in M \to \forall X(X \subseteq M \to X^{suc} \subseteq M) \to \mathbb{N} \subseteq M$$

a partir del principio de inducción de Mendler, como sigue:

$$\begin{array}{l}
Si\ \forall X(X \subseteq M \to \Phi(X) \subseteq M)\ entonces\ \mu(\Phi^{\sqsupseteq}) \subseteq M \\
Si\ \forall X(X \subseteq M \to \{0\} \cup X^{suc} \subseteq M)\ entonces\ \mathbb{N} \subseteq M \\
\forall X(X \subseteq M \to 0 \in M \wedge X^{suc} \subseteq M) \to \mathbb{N} \subseteq M \\
\forall X((X \subseteq M \to 0 \in M) \wedge (X \subseteq M \to X^{suc} \subseteq M)) \to \mathbb{N} \subseteq M \\
\forall X((X \subseteq M \to 0 \in M) \wedge \forall X(X \subseteq M \to X^{suc} \subseteq M)) \to \mathbb{N} \subseteq M \\
0 \in M \wedge \forall X(X \subseteq M \to X^{suc} \subseteq M) \to \mathbb{N} \subseteq M \\
0 \in M \to \forall X(X \subseteq M \to X^{suc} \subseteq M) \to \mathbb{N} \subseteq M
\end{array}$$

El principio de inducción de Mendler se puede reescribir de manera más familiar como sigue.

Sea $\varphi$ una propiedad de números naturales. Si

- $\varphi(0)$
- $\forall X(\forall x \in X(\varphi(x)) \to \forall x \in X(\varphi(suc\ x)))$

entonces

$$\forall x \in \mathbb{N}(\varphi(x))$$

Este principio es poco conocido hasta donde sabemos y nos dice intuitivamente que para demostrar una propiedad para todos los naturales, basta mostrar que $\Phi(0)$ y, suponiendo que todos los elementos de un conjunto arbitrario $X$ cumplen $\Phi$, demostrar que todos los sucesores de elementos de $X$ cumplen $\Phi$.

Con esto terminamos nuestra discusión acerca de la teoría de puntos fijos. El siguiente apartado se dedica a definir la lógica base de nuestro trabajo la cual será extendida más adelante utilizando los conceptos recién estudiados.

# Capítulo 2

# La lógica de segundo orden

En la descripción de sistemas lógicos el concepto de orden se refiere a una clasificación de variables que establece sobre cuáles se puede cuantificar. Por ejemplo, en la lógica de predicados de primer orden se permite cuantificar sobre variables que denoten individuos, las cuales reciben el nombre de variables de primer orden.

Así, la lógica de predicados de segundo orden es una extensión de la lógica de predicados de primer orden que además permite la cuantificación sobre propiedades de individuos. La lógica de segundo orden a la que nos referimos en este trabajo es el sistema llamado aritmética funcional de segundo orden, denotada AF2, el cual es un sistema de deducción natural para la lógica de predicados de segundo orden desarrollado por Leivant y Krivine [6, 4].

Dicha lógica posee un sistema de codificación de pruebas y un mecanismo de razonamiento ecuacional que utiliza la igualdad de Leibniz, características que permiten la extracción de programas a partir de especificaciones lógicas mediante el paradigma de programación con pruebas de Krivine-Parigot [5, 16]. Este método de síntesis de programas asegura que los programas (términos $\lambda$) extraidos a partir de pruebas de totalidad [1] para funciones son correctos.

A continuación se presentan la sintáxis de la lógica de segundo orden, la interpretación BHK, la deducción natural y el cálculo lambda, fundamentos que permitirán definir después la lógica AF2.

## 2.1. Sintáxis

La sintáxis de la lógica de segundo orden es la siguiente:

- *Variables*: se cuenta con dos conjuntos infinitos ajenos de variables, llamadas variables de primer orden, denotadas con letras minúsculas $x, y, z, \ldots$ y variables de segundo orden, denotadas con letras mayúsculas $X, Y, Z, \ldots$.

[1] Es decir, dada $f : A \to B$, una prueba de totalidad para $f$ es una prueba de que $\forall a \in A\ \exists b \in B\ (f(a) = b)$.

- *Términos*: se definen de la manera usual a partir de variables de primer orden y de una signatura fija $\Sigma$ que incluye símbolos de función $f$ de un índice o número de argumentos fijo.

$$r, s, t ::= x \mid f(t_1, \ldots, t_n)$$

En adelante será conveniente escribir $f(\vec{t})$ en vez de $f(t_1, \ldots, t_n)$. Obsérvese que esta definición incluye el caso en que el índice de un símbolo de función $f$ es cero, el cual corresponde a un símbolo de constante.

- *Fórmulas*: se definen de la manera usual como sigue:

$$A, B, C ::= \mathcal{P}(\vec{t}) \mid A \to B \mid A \wedge B \mid A \vee B \mid \forall x A \mid \forall X A$$

donde las variables $x$ y $X$ se consideran ligadas en las fórmulas $\forall x A$ y $\forall X A$ respectivamente. En adelante algunas veces se adopta la notación de punto en un cuantificador para declarar que su alcance se considera lo más lejos posible sintácticamente, por ejemplo la proposición $\forall x.A \to B$ significa $\forall x(A \to B)$ y no $(\forall x A) \to B$.

- *Predicados*: los predicados pueden ser variables de segundo orden $X$, símbolos de predicado $P$ de una signatura fija $\Sigma$ o bien predicados por comprensión $\mathcal{F}$.

$$\mathcal{P} ::= X \mid P \mid \mathcal{F}$$

Un predicado por comprensión es una expresión de la forma $\mathcal{F} =_{def} \{\vec{x} \,|\, A\}$ donde $A$ es una fórmula y $\vec{x}$ es un vector de variables de primer orden, usualmente libres en $A$. El índice o número de argumentos de $\mathcal{F}$ se define como la longitud del vector $\vec{x}$. Esta clase de predicados representa al conjunto de tuplas $\vec{t}$ tales que $A[\vec{x} := \vec{t}]$ es válida. En particular, para cualquier tupla de términos $\vec{s}$, la fórmula atómica $\mathcal{F}\vec{s}$ se define como la fórmula $A[\vec{x} := \vec{s}]$. La operación de sustitución de variables por términos se define más adelante.

Obsérvese que los predicados y las fórmulas se definen simultaneamente de manera inductiva. Más aún, la lógica de segundo orden es lo suficientemente expresiva para definir los conectivos $\wedge, \vee$, así como el cuantificador existencial $\exists$, mediante $\forall$ e $\to$. En este trabajo son considerados como primitivos por mera conveniencia. Asímismo, nótese que la negación está ausente, lo cual significa que se está usando una lógica minimal, ya que se desea trabajar con los aspectos constructivos de la lógica, los cuales desaparecen con la negación.

A continuación mostramos algunos ejemplos de la expresividad de la lógica de segundo orden, cuya justificación puede consultarse en el artículo *La lógica proposicional de segundo orden* [14].

**Ejemplo 1.** *Los conectivos lógicos $\wedge, \vee$, los cuantificadores existenciales así como los números naturales y su principio de inducción se definen como sigue:*

- *La conjunción: sean $A, B$ dos fórmulas y $X$ una variable de segundo orden de índice cero que no figura en $A$ ni en $B$.*

$$A \wedge B =_{def} \forall X.(A \to B \to X) \to X$$

- *La disyunción: sean $A, B$ dos fórmulas y $X$ una variable de segundo orden de índice cero que no figura en $A$ ni en $B$.*

$$A \vee B =_{def} \forall X.(A \to X) \to (B \to X) \to X$$

- *La cuantificación existencial de primer orden: sean $A$ una fórmula y $X$ una variable de segundo orden de índice cero tal que $X \notin FV(A)$.*

$$\exists x A =_{def} \forall X \big(\forall x(A \to X) \to X\big)$$

- *La cuantificación existencial de segundo orden: sean $A$ una fórmula y $Y$ una variable de segundo orden de índice cero tal que $Y \notin FV(A)$.*

$$\exists X A =_{def} \forall Y \big(\forall X(A \to Y) \to Y\big)$$

- *Los números naturales: se definen como el predicado más pequeño $\mathbb{N}$ cerrado bajo el cero $0$ y la función sucesor $s$. Esta idea se captura con el siguiente predicado por comprensión:*

$$\mathbb{N}(y) =_{def} \{y|\forall X.\forall x(X(x) \to X(s(x))) \to X(0) \to X(y)\}$$

- *Postulado de inducción de Peano:*

$$\forall X.\forall x(X(x) \to X(s(x))) \to X(0) \to \forall x.\mathbb{N}(x) \to X(x)$$

Los dos últimos ejemplos se pueden generalizar, utilizando la teoría de puntos fijos desarrollada anteriormente, para definir cualquier estructura de datos usual, como listas o árboles. Sin embargo la definición utiliza directamente el principio de inducción correspondiente y por lo tanto se aleja de las definiciones usuales en lenguajes de programación. Nuestro objetivo final es definir una lógica que cuente con un mecanismo de definición de tipos de datos similares a los hallados en lenguajes funcionales.

### 2.1.1. La igualdad de Leibniz

En lógica constructiva el símbolo de igualdad puede ser considerado primitivo, aunque lo usual para los sistemas de primer orden y para aplicaciones, es tratarlo como un símbolo de predicado binario distinguido, el cual satisface ciertos axiomas no lógicos, además de las propiedades de reflexividad, simetría y transitividad. En el caso de la lógica de segundo orden la igualdad es definible y permite el razonamiento ecuacional directo como veremos en esta sección.

**Definición 2.1.** Dados dos términos $r$ y $s$ definimos la relación de igualdad de Leibniz, denotada $r = s$, como la cerradura universal de primer orden de la fórmula

$$\forall X.X(r) \to X(s).$$

Cualquier fórmula de la forma $r = s$ se llamará ecuación. Se observa que una ecuación es una fórmula con una única variable ligada de segundo orden y en donde convenimos que las variables de primer orden están ligadas universalmente, aunque no escribimos dichos cuantificadores para simplificar la notación.

Más adelante se mostrará que esta definición de igualdad cumple las propiedades esperadas, es decir las propiedades de la igualdad usual. En particular la lógica AF2 se sirve ampliamente del razonamiento ecuacional provisto por la igualdad de Leibniz.

## 2.2. Sustitución

Dado que se tienen dos clases de variables y en ambas está presente un mecanismo de ligado, es necesario definir con cuidado las operaciones de sustitución. Los conjuntos $Var(t)$ de variables de primer orden de un término $t$, $Var_2(\mathcal{P})$ de variables de segundo orden de un predicado $\mathcal{P}$ y $FV(A)$ de variables libres de primer y segundo orden de una fórmula $A$ se definen de la manera natural.

**Definición 2.2.** La *sustitución* de variables de primer orden $\vec{x} = x_1, \dots, x_n$ por términos $\vec{t} = t_1, \dots, t_n$ en un término $r$, denotada $r[\vec{x} := \vec{t}]$ se define como la sustitución textual de cada presencia de $x_i$ por $t_i$ en $r$.

**Definición 2.3.** La *sustitución* de variables de primer orden $\vec{x}$ por términos $\vec{t}$ en la fórmula $A$, denotada $A[\vec{x} := \vec{t}]$ se define recursivamente como sigue:

$$
\begin{array}{rcl}
\mathcal{Q}(r_1, \dots, r_m)[\vec{x} := \vec{t}\,] & = & \mathcal{Q}\big(r_1[\vec{x} := \vec{t}\,], \dots, r_m[\vec{x} := \vec{t}\,]\big) \\
(A \wedge B)[\vec{x} := \vec{t}\,] & = & \big(A[\vec{x} := \vec{t}\,] \wedge B[\vec{x} := \vec{t}\,]\big) \\
(A \vee B)[\vec{x} := \vec{t}\,] & = & \big(A[\vec{x} := \vec{t}\,] \vee B[\vec{x} := \vec{t}\,]\big) \\
(A \to B)[\vec{x} := \vec{t}\,] & = & \big(A[\vec{x} := \vec{t}\,] \to B[\vec{x} := \vec{t}\,]\big) \\
(\forall x A)[\vec{x} := \vec{t}\,] & = & \forall x\big(A[\vec{x} := \vec{t}\,]\big) \text{ si } x \notin \vec{x} \cup Var(\vec{t}) \\
(\forall X A)[\vec{x} := \vec{t}\,] & = & \forall X\big(A[\vec{x} := \vec{t}\,]\big)
\end{array}
$$

La condición en el caso de las cuantificaciones de primer orden siempre puede cumplirse al renombrar la variable ligada $x$ en la fórmula original. Esto no causa problema alguno pues dos fórmulas que difieren únicamente en los nombres de sus variables ligadas se consideran iguales. Esta convención se conoce como $\alpha$-equivalencia, por ejemplo $\forall x \mathcal{P}(x)$ y $\forall y \mathcal{P}(y)$ son fórmulas $\alpha$-equivalentes. Obsérvese que si se desea que la operación de sustitución defina a una función total, en lugar de fórmulas, es necesario manipular clases de $\alpha$-equivalencia de fórmulas, situación que no se hace explícita. Lo mismo sucede para el caso de la sustitución en cuantificaciones de segundo orden definida más adelante.

**Definición 2.4.** La *sustitución* de una variable de segundo orden $X$ por el predicado $\mathcal{P}$ en el predicado $\mathcal{Q}$, denotada $\mathcal{Q}[X := \mathcal{P}]$ se define como sigue:

$$
\begin{array}{rcl}
X[X := \mathcal{P}\,] & = & \mathcal{P} \\
Y[X := \mathcal{P}\,] & = & Y \\
P[X := \mathcal{P}\,] & = & P \\
\{\vec{x} \mid A\}[X := \mathcal{P}\,] & = & \{\vec{x} \mid A[X := \mathcal{P}]\}
\end{array}
$$

**Definición 2.5.** La *sustitución* de una variable de segundo orden $X$ por el predicado $\mathcal{P}$ en la fórmula $A$, denotada $A[X := \mathcal{P}]$ se define recursivamente como sigue:

$$
\begin{array}{rcl}
\mathcal{Q}(r_1, \dots, r_m)[X := \mathcal{P}\,] & = & \mathcal{Q}[X := \mathcal{P}](r_1, \dots, r_m) \\
(A \to B)[X := \mathcal{P}\,] & = & \big(A[X := \mathcal{P}\,] \to B[X := \mathcal{P}\,]\big) \\
(A \wedge B)[X := \mathcal{P}\,] & = & \big(A[X := \mathcal{P}\,] \wedge B[X := \mathcal{P}\,]\big) \\
(A \vee B)[X := \mathcal{P}\,] & = & \big(A[X := \mathcal{P}\,] \vee B[X := \mathcal{P}\,]\big) \\
(\forall x A)[X := \mathcal{P}\,] & = & \forall x\big(A[X := \mathcal{P}\,]\big) \\
(\forall Y A)[X := \mathcal{P}\,] & = & \forall Y\big(A[X := \mathcal{P}\,]\big) \textbf{ si } Y \notin \{X\} \cup Var_2(\mathcal{P})
\end{array}
$$

## 2.3. La interpretación BHK

La lógica clásica se basa en la noción primitiva de *verdad*. La verdad de una fórmula es una propiedad absoluta de la misma, en el sentido de que no depende de ningún razonamiento, entendimiento o acción. Un enunciado declarativo libre de ambigüedades es verdadero o no verdadero (es decir, falso), independientemente de si conocemos su valor de verdad, o de si lo probamos o verificamos en cualquier forma posible. Por otra parte, en muchas aplicaciones no basta con saber cual es la solución a un problema, sino que es necesario construirla, de modo que es deseable separar los métodos de prueba que proporcionan soluciones, de aquellos que no lo hacen. Por lo tanto, pragmáticamente tiene sentido considerar un enfoque constructivo de la lógica. Esta lógica, llamada constructiva o intuicionista se basa en la idea de que la verdad significa demostrabilidad (Brouwer 1907,1918). Para entender la lógica intuicionista, debemos olvidar la noción clásica (tarskiana) de verdad. Los juicios acerca de una fórmula ya no se basan en un valor de verdad asignado a ella, sino en la habilidad para construirla mediante una prueba explícita. Como consecuencia los conectivos no se deben definir mediante tablas de verdad. En su lugar, se explica el significado de fórmulas compuestas en términos de sus construcciones. Tal explicación se da de manera intuitiva mediante la llamada interpretación de Brouwer-Heyting-Kolmogorov (BHK) que es reconocida ampliamente como la semántica intensional de la lógica intuicionista.

**Definición 2.6.** La *interpretación BHK* se define recursivamente como sigue:

- Una prueba de una variable proposicional $p$ se supone conocida y dada por un contexto previamente definido.
- Una prueba de $A \wedge B$ es un par $\langle p, q\rangle$ donde $p$ es una prueba de $A$ y $q$ es una prueba de $B$.
- Una prueba de $A \vee B$ es una prueba $p$ de $A$ o una prueba $q$ de $B$, junto con una etiqueta que indique cual de las dos fórmulas se está probando.
- Una prueba de $A \to B$ es un método funcional $f$ que transforma cualquier prueba $p$ de $A$ en una prueba $f(p)$ de $B$.

**Ejemplo 2.** *Algunos ejemplos son:*

- *La función identidad id es una prueba de $A \to A$, pues dada cualquier prueba $p$ de $A$, $id(p) = p$ es una prueba de $A$.*
- *Si $f$ es una prueba de $A \to B$ y $g$ es una prueba de $B \to C$ entonces $g \circ f$ es una prueba de $A \to C$. Por lo tanto la función $F$, dada por $F(f)(g) = g \circ f$ es una prueba de $(A \to B) \to (B \to C) \to A \to C$*
- *La función swap definida como $swap(\langle p, q\rangle) = \langle q, p\rangle$ es una prueba de $A \wedge B \to B \wedge A$.*
- *Una prueba de $A \to A \vee B$ está dada por la función $f(p) = \mathsf{inl}\, p$. Obsérvese que* inl *es la etiqueta que indica que la prueba de $A \vee B$ es en realidad una prueba de $A$.*

Observamos que la discusión anterior sólo se refiere a la lógica proposicional, la versión más común de la interpretación BHK para los cuantificadores es la siguiente:

- Una prueba de $\forall x A$ es un método funcional que transforma a cada objeto $a$ en una prueba de $A[x := a]$.
- Una prueba de $\forall X A$ es un método funcional que transforma a cada relación $R$ en una prueba de $A[X := R]$.

Sin embargo nosotros usaremos una versión donde los cuantificadores no tienen contenido computacional (ver [1]), lo cual significa, para nuestros propósitos, lo siguiente:

- Una prueba de $\forall x A$ es una prueba de $A(a)$ que no depende de ninguna propiedad del objeto $a$, es decir, una prueba de $A$ paramétrica en $a$.
- Una prueba de $\forall X A$ es una prueba de $A(R)$ que no depende de ninguna propiedad de la relación $R$, es decir, una prueba de $A$ paramétrica en $R$.

La semántica intuitiva anterior consta de afirmaciones metalógicas pero puede formalizarse de diversas maneras, siendo las más usadas la semántica de álgebras de Heyting y la semántica de mundos posibles de Kripke, ambas equivalentes (véase [20]). Nosostros utilizaremos la interpretación BHK, para darle sentido como un sistema de codificación de prueba, pero para lograr este objetivo primero se necesita una noción formal de prueba, lo cual es abordado en la siguiente sección.

## 2.4. Deducción natural

Se elige como sistema deductivo para la lógica de segundo orden, al sistema de deducción natural con hipótesis localizadas, es decir, con contextos de fórmulas.

**Definición 2.7.** Un *contexto* es un conjunto de fórmulas $\Gamma = \{A_1, \ldots, A_n\}$.

Por convención se omiten las llaves de conjunto así como la operación de unión, escribiendo $\Gamma, \Delta$ en vez de $\Gamma \cup \Delta$ y $\Gamma, A$ en vez de $\Gamma \cup \{A\}$. Mas aún, en un contexto de la forma $\Gamma, A$ se supone que que $A$ no figura en $\Gamma$. El conjunto de variables libres del contexto $\Gamma = \{A_1, \ldots, A_n\}$ se define como $FV(\Gamma) =_{def} FV(A_1) \cup \ldots \cup FV(A_n)$.

**Definición 2.8.** Un *secuente* es una expresión de la forma $\Gamma \vdash A$ donde $\Gamma$ es un contexto y $A$ es una fórmula. En particular el secuente $\vdash A$ significa $\varnothing \vdash A$.

El secuente $\Gamma \vdash A$ expresa la relación de derivabilidad de la fórmula $A$ a partir de las hipótesis dadas en $\Gamma$. Esta relación se define formalmente mediante las siguientes reglas de inferencia entre secuentes.

**Definición 2.9.** La relación de *derivabilidad* $\Gamma \vdash A$ se define recursivamente como sigue:

- Regla de inicio:

$$\frac{}{\Gamma, A \vdash A}\ (Hip)$$

- Implicación:

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\ (\to I) \qquad \frac{\Gamma \vdash A \to B \quad \Gamma \vdash A}{\Gamma \vdash B}\ (\to E)$$

- Conjunción:

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ (\wedge I) \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\ (\wedge E) \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\ (\wedge E)$$

- Disyunción

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\ (\vee_1 I) \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\ (\vee_2 I)$$

$$\frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C}\ (\vee E)$$

- Cuantificación universal de primer orden:

$$\frac{\Gamma \vdash A \quad x \notin FV(\Gamma)}{\Gamma \vdash \forall x A}\ (\forall I) \qquad \frac{\Gamma \vdash \forall x A}{\Gamma \vdash A[x := t]}\ (\forall E)$$

- Cuantificación universal de segundo orden:

$$\frac{\Gamma \vdash A \quad X \notin FV(\Gamma)}{\Gamma \vdash \forall X A}\ (\forall^2 I) \qquad \frac{\Gamma \vdash \forall X A}{\Gamma \vdash A[X := \mathcal{P}]}\ (\forall^2 E)$$

Obsérvese que a cada conectivo o cuantificador le corresponden reglas que lo introducen, denotadas con $I$, así como reglas que lo eliminan, denotadas con $E$. Esta simetría o dualidad en las reglas es de gran importancia y proporciona un determinismo en la relación de derivabilidad. Cada fórmula compuesta puede ser derivada usando una única regla de introducción. Así mismo, la información de cada fórmula compuesta puede ser utilizada para derivar otras fórmulas mediante una única regla de eliminación.

Una derivación de un secuente particular se define como:

**Definición 2.10.** Una *derivación* del secuente $\Gamma \vdash A$ es una sucesión finita de secuentes $\Gamma_1 \vdash A_1, \ldots, \Gamma_n \vdash A_n$ tal que:

- $\Gamma_i \vdash A_i$ es instancia de la regla $(Hip)$ ó
- $\Gamma_i \vdash A_i$ es conclusión de alguna regla de inferencia tal que las premisas necesarias figuran antes en la sucesión.
- $\Gamma \vdash A$ es el último elemento de la sucesión.

La siguiente proposición asegura que la lógica es monótona.

**Proposición 2.1.** La siguiente regla de inferencia es derivable:

$$\frac{\Gamma \vdash A}{\Gamma, B \vdash A}\ (Mon)$$

*Demostración.* Inducción sobre $\vdash$. □

### 2.4.1. Reglas para la igualdad

A continuación se muestra que la igualdad de Leibniz cumple las propiedades usuales de la igualdad por lo que permite el razonamiento ecuacional común en matemáticas.

**Proposición 2.2.** La igualdad de Leibniz es una relación de equivalencia compatible con funciones, es decir, se cumple lo siguiente para cualquier contexto $\Gamma$:

- Reflexividad (Ref): $\Gamma \vdash r = r$
- Transitividad (Trn): Si $\Gamma \vdash r = s$ y $\Gamma \vdash s = t$ entonces $\Gamma \vdash r = t$
- Simetría (Sim): Si $\Gamma \vdash r = s$ entonces $\Gamma \vdash s = r$
- Compatibilidad (Com): Si $\Gamma \vdash r_i = s_i$, para toda $1 \leq i \leq n$ entonces

$$\Gamma \vdash f(r_1, \ldots, r_n) = f(s_1, \ldots, s_n)$$

*Demostración.* Es claro que se cumplen la reflexividad y la transitividad. Para mostrar la simetría definimos el predicado por comprensión $\mathcal{F} =_{def} \{x|x = r\}$, donde $x \notin Var(r)$, y mostramos que $\Gamma \vdash \mathcal{F}(s)$.

| | | |
|---|---|---|
| 1. | $\Gamma \vdash \forall X.X(r) \to X(s)$ | $(Hip)$ y def. de $r = s$. |
| 2. | $\Gamma \vdash \mathcal{F}(r) \to \mathcal{F}(s)$ | $(\forall^2 E)$, 1, $[X := \mathcal{F}]$ |
| 3. | $\Gamma \vdash \mathcal{F}(r)$ | ((Ref)) |
| 4. | $\Gamma \vdash \mathcal{F}(s)$ | $(\to E), 2, 3$ |

Para la compatibilidad: mostramos el caso para $n = 2$, el caso general resulta de una sencilla inducción. Sean $\mathbb{E} = \{r_1 = s_1, r_2 = s_2\}, \mathcal{F}_1 =_{def} \{x|f(r_1, r_2) = f(x, r_2)\}$, donde $x \notin Var(r_2)$, $\mathcal{F}_2 =_{def} \{y|f(s1, r2) = f(s1, y)\}$ donde $y \notin Var(s_1)$.

| | | |
|---|---|---|
| 1. | $\Gamma \vdash \mathcal{F}_1(r_1)$ | (Ref) |
| 2. | $\Gamma \vdash \forall X.X(r_1) \to X(s_1)$ | $(Hip)$ y def. de $r_1 = s_1$. |
| 3. | $\Gamma \vdash \mathcal{F}_1(r_1) \to \mathcal{F}_1(s_1)$ | $(\forall^2 E)$, 2, $[X := \mathcal{F}_1]$ |
| 4. | $\Gamma \vdash \mathcal{F}_1(s_1)$ | $(\to E)$, 1, 3 |
| 5. | $\Gamma \vdash \mathcal{F}_2(r_2)$ | (Ref) |
| 6. | $\Gamma \vdash \forall X.X(r_2) \to X(s_2)$ | $(Hip)$ y def. de $r_2 = s_2$. |
| 7. | $\Gamma \vdash \mathcal{F}_2(r_2) \to \mathcal{F}_2(s_2)$ | $(\forall^2 E)$, 6, $[X := \mathcal{F}_2]$ |
| 8. | $\Gamma \vdash \mathcal{F}_2(s_2)$ | $(\to E)$, 5, 7 |
| 9. | $\Gamma \vdash f(r_1, r_2) = f(s_1, s_2)$ | (Trn)), 4, 8 |

□

El razonamiento ecuacional común se formaliza mediante la siguiente relación de inferencia.

**Definición 2.11.** Dado un conjunto de ecuaciones $\mathbb{E}$, decimos que la ecuación $r = s$ se deriva a partir de $\mathbb{E}$, lo cual denotamos con $\mathbb{E} \rhd r = s$, si y sólo si existe una derivación mediante las siguientes reglas de inferencia:

- (Eq-Ax). $\mathbb{E} \rhd r = s$, si $r = s$ es un caso particular de una ecuación en $\mathbb{E}$, es decir, $r = s$ se obtiene de una ecuación de $\mathbb{E}$ instanciando sus variables.
- Reflexividad, simetria, transitividad y compatibilidad con funciones:

$$\frac{}{\mathbb{E} \rhd r = r}\ (\textsc{Eq-Ref}) \qquad \frac{\mathbb{E} \rhd r = s}{\mathbb{E} \rhd s = r}\ (\textsc{Eq-Sim})$$

$$\frac{\mathbb{E} \rhd r = s \quad \mathbb{E} \rhd s = t}{\mathbb{E} \rhd r = t}(\textsc{Eq-Trn})$$

$$\frac{\mathbb{E} \rhd r_1 = s_1 \ \ \ldots \ \ \mathbb{E} \rhd r_n = s_n}{\mathbb{E} \rhd f(r_1, \ldots, r_n) = f(s_1, \ldots, s_n)}(\textsc{Eq-Fun})$$

La siguiente proposición muestra que el razonamiento ecuacional usual es capturado por la igualdad de Leibniz, puesto que las reglas de la definición 2.11 son derivables en la lógica de segundo orden.

**Proposición 2.3.** Si $\mathbb{E} \rhd r = s$ entonces $\mathbb{E} \vdash r = s$.

*Demostración.* Inducción sobre $\rhd$, utilizando la proposición 2.2 □

El sistema de deducción natural recién presentado nos permitirá construir pruebas acerca de cierta noción de correctud de la especificación de un problema, a partir de las cuales se extraerá un programa funcional que cumpla dicha especificación. En este trabajo entenderemos un programa funcional como un término del cálculo lambda, por ello es necesario presentar brevemente este formalismo.

## 2.5. El cálculo $\lambda$

El cálculo $\lambda$ fue inventado en la década de 1930 por Alonso Church. En un inicio era parte de un sistema más complejo que pretendia ser un fundamento matemático para la lógica, pero en 1935 Kleene y Rosser, estudiantes de Church, descubrieron que dicho sistema era inconsistente. No obstante el subsistema de términos conocido hoy como cálculo $\lambda$ ha sido desde entonces estudiado independientemente como un modelo de cómputo. Este cálculo se ocupa únicamente de funciones, en particular modela funciones que toman otras funciones como argumento o que devuelven funciones como resultado, es decir funciones de orden superior. En términos de lenguajes de programación, el cálculo $\lambda$ es un prototipo extremadamente simple de un lenguaje funcional puro de orden superior, por lo que resulta conveniente presentar una breve introducción a este formalismo. Para un estudio profundo véase [20].

La sintaxis del cálculo $\lambda$ puro es la siguiente:

$$e ::= x \mid \lambda x.e \mid ee$$

Es decir, una expresión del cálculo $\lambda$ (un $\lambda$-término) es una expresión de alguna de las siguientes formas:

- *Variables*: la variable $x$ representa a una función cualquiera.

- *Abstracción lambda*: una expresión de la forma $\lambda x.e$ se llama *abstracción lambda* o simplemente *abstracción*. El término $\lambda x.e$ indica una abstracción de los valores particulares de la variable $x$ en la expresión $e$ lo cual causa que $e$ se considere una función de $x$, es decir, la expresión $\lambda x.e$ define anónimamente a la función $x \mapsto e$ que asocia a cada valor $x$ la expresión $e$ . El punto en una abstracción es, como en el caso de los cuantificadores de la lógica, un uso particular de la llamada sintaxis de orden superior en lenguajes de programación y denota el ligado de $x$ en $e$, además indica que el alcance del ligado para $x$ se extiende a la derecha tanto como sea posible. Por ejemplo, $\lambda x.xy$ significa $\lambda x.(xy)$ y no $(\lambda x.x)y$. Para facilitar la escritura de abstracciones usaremos la siguiente convención

$$\lambda x_1 x_2 \dots x_n.e =_{def} \lambda x_1.\lambda x_2.\dots.\lambda x_n.e$$

- *Aplicación*: la expresión $e_1 e_2$ representa a la aplicación de la función $e_1$ al argumento $e_2$. La aplicación se asocia a la izquierda de manera que $e_1\, e_2\, e_3$ significa $(e_1\, e_2)\, e_3$ y nunca $e_1\, (e_2\, e_3)$. Obsérvese que no hay un operador explícito para la aplicación, ésta se denota simplemente mediante la escritura secuencial de dos expresiones.

Es importante observar que los $\lambda$-términos son las únicas expresiones válidas en el cálculo $\lambda$ y que el operador $\lambda$ se comporta de manera similar a un cuantificador en la lógica usual, es decir, en la abstracción $\lambda x.e$ el operador $\lambda$ liga a la variable $x$ en la expresión $e$, por lo que se utiliza la noción de variables libres y ligadas usual en lógica. En particular se tiene que el conjunto de variables libres de una abstracción se define como $FV(\lambda x.e) =_{def} FV(e) \setminus \{x\}$. Por ejemplo en $(\lambda xy.xzy)(\lambda z.zy)u$ las dos presencias de $x$ son ligadas así como las dos primeras presencias de $y$ y las dos últimas presencias de $z$ mientras que el resto de las presencias de cada variable son libres.

El sistema de $\lambda$-términos es dinámico, a diferencia del sistema de términos de una lógica, en el sentido de que los términos operan entre si de acuerdo a una relación binaria llamada semántica operacional. Esta semántica denotada $\to_\beta$ se define como la cerradura de la siguiente regla, conocida universalmente como $\beta$-reducción, bajo todos los constructores de término.

$$(\lambda x.e)t \mapsto_\beta e[x := t]$$

Lo que esta regla nos dice es que la acción de aplicar la función $\lambda x.e$ al argumento $t$ consiste en sustituir el parámetro $x$ por el término $t$ en el término $e$. Se observa entonces que todo paso de evaluación es simplemente una substitución definida de la manera usual mediante el uso de la $\alpha$-equivalencia (renombre de variables ligadas) para evitar la captura de variables libres:

- $x[x := r] = r$.
- $y[x := r] = y$ si $x \neq y$.
- $(ts)[x := r] = t[x := r]s[x := r]$.
- $(\lambda yt)[x := r] = \lambda y.t[x := r]$ donde s.p.g. $y \neq x$ y $y \notin FV(r)$.

Dos relaciones de importancia, derivadas de $\to_\beta$, son su cerradura reflexiva-transitiva, denotada $\to_\beta^\star$ y su cerradura reflexiva-simetrica-transitiva, denotada $=_\beta$ y llamada $\beta$-equivalencia.

**Ejemplo 3.** *Los siguientes son ejemplos de $\beta$-reducciones y $\beta$-equivalencia.*

- $(\lambda xx)y \to y$
- $(\lambda xy)t \to y$
- $(\lambda x.x(xy))t \to t(ty)$
- $(\lambda x.x(\lambda xx))(ur) \to (ur)(\lambda xx)$
- $(\lambda x.(\lambda y.yx)z)v \to (\lambda y.yv)z$
- *Si* $I =_{def} \lambda xx$ *entonces* $III \to^{\star} I$ *e* $III =_{\beta} II$
- *Si* $S =_{def} \lambda xyz.xz(yz)$ *y* $K =_{def} \lambda xy.x$ *entonces* $SKK \to^{\star} I$ *y* $SKK =_{\beta} II$.

Los términos $ISK$ se conocen como combinadores y forman la base de la llamada lógica combinatoria de Curry.

El cálculo lambda es lo suficientemente poderoso para representar tipos de datos como booleanos y números naturales. Veamos de que manera.

**Ejemplo 4** (Booleanos)**.** *Considérense los siguientes $\lambda$-términos.*

- $\mathsf{true} := \lambda x \lambda y.x$
- $\mathsf{false} := \lambda x \lambda y.y$
- $\mathsf{test} := \lambda b.\lambda t \lambda e.bte$
- $\mathsf{not} := \lambda z.z\,\mathsf{false}\,\mathsf{true}$
- $\mathsf{and} := \lambda x \lambda y \lambda z \lambda w.x(yzw)w$

*Es fácil cerciorarse de lo siguiente:*

- $\mathsf{test}\,\mathsf{true}\,e_1\,e_2 \to^{\star} e_1$
- $\mathsf{test}\,\mathsf{false}\,e_1\,e_2 \to^{\star} e_2$

*por lo que el término* $\mathsf{test}$ *se comporta como el condicional booleano*

- $\mathsf{not}\,\mathsf{true} \to^{\star} \mathsf{false}$
- $\mathsf{not}\,\mathsf{false} \to^{\star} \mathsf{true}$
- $\mathsf{and}\,\mathsf{false}\,b \to^{\star} \mathsf{false}$
- $\mathsf{and}\,\mathsf{true}\,b \to^{\star} b$

**Ejemplo 5** (Números naturales (numerales de Church))**.** *Los $\lambda$-términos conocidos como numerales de Church se definen como sigue*

- $\bar{0} := \lambda s.\lambda z.z$

- $\bar{1} := \lambda s.\lambda z.sz$
- $\bar{2} := \lambda s.\lambda z.s(sz)$
- $\bar{3} := \lambda s.\lambda z.s(s(sz))$
- $\bar{n} := \lambda s.\lambda z.\underbrace{s(\ldots(s}_{n\ veces}\,z)\ldots)$

*Las operaciones aritméticas básicas se definen como sigue, y la semántica operacional del cálculo lambda permite verificar que se comportan de la forma esperada.*

- $\mathsf{suc} := \lambda n.\lambda s.\lambda z.s(nsz)$ *tal que*
$$\forall n \in \mathbb{N}\big(\,\mathsf{suc}\,\bar{n} \to^{\star} \overline{n+1}\big)$$
- $\mathsf{suma} := \lambda n\lambda m.m(\lambda u\lambda f\lambda z.f(ufz))n$ *tal que*
$$\forall n, m \in \mathbb{N}\big(\mathsf{suma}\,\bar{n}\,\bar{m} \to^{\star} \overline{n+m}\big)$$
- $\mathsf{prod} := \lambda m.\lambda n.m(\mathsf{suma}\ n)\bar{0}$ *tal que*
$$\forall n, m \in \mathbb{N}\big(\mathsf{prod}\,\bar{n}\,\bar{m} \to^{\star} \overline{n*m}\big)$$
- $\mathsf{iszero} := \lambda m.m(\lambda x.\mathsf{false})\mathsf{true}$ *tal que*
$$\mathsf{iszero}\ \bar{0} \to^{\star} \mathsf{true} \qquad \forall n \in \mathbb{N}\big(\,\mathsf{iszero}\,(\overline{n+1}) \to^{\star} \mathsf{false}\big)$$

Las definiciones anteriores son conocidas desde la introducción del cálculo lambda en la década de 1930. Se observa que estas definiciones no son fáciles de leer como programas funcionales, es decir, el proceso de síntesis de la definición no se ha discutido. Esto puede hacerse de diversas maneras, sin embargo no es de nuestro interés hacerlo aquí pues dichas definiciones pueden justificarse mediante el mecanismo de extracción de programas de AF2 discutido en la sección 2.6.1.

### 2.5.1. Extensión con pares e inyecciones

Se pretende utilizar el cálculo lambda como un sistema de codificación o representación de pruebas en la lógica, para así poder extraer programas ($\lambda$ términos) automáticamente a partir de las pruebas en deducción natural. Para ello es necesario extender el cálculo lambda puro con términos que permitan representar las pruebas que involucran conjunciones y disyunciones. Esta extensión se define a continuación.

- $\lambda$-*términos*: agregamos términos para las operaciones de par ordenado, proyecciones, inyecciones y análisis de casos.
$$e ::= \ldots \mid \langle e, e\rangle \mid \mathsf{fst}\, e \mid \mathsf{snd}\, e \mid \mathsf{inl}\, e \mid \mathsf{inr}\, e \mid \mathsf{case}(r, x.s, y.t)$$
donde en el último caso la notación $x.s\ y.t$ denota un ligado de la respectiva variable en el término que está después del punto.

- *Semántica operacional*: la relación $\mapsto_\beta$ se extiende como sigue:

$$\begin{array}{rcl}
\mathsf{fst}\langle r, s\rangle & \mapsto_\beta & r \\
\mathsf{snd}\langle r, s\rangle & \mapsto_\beta & s \\
\mathsf{case}(\mathsf{inl}\, r, x.s, y.t) & \mapsto_\beta & s[x := r] \\
\mathsf{case}(\mathsf{inr}\, r, x.s, y.t) & \mapsto_\beta & t[y := r]
\end{array}$$

Las reglas para el constructor de casos reflejan el análisis de una inyección de $r$, dependiendo de si es izquierda ($\mathsf{inl}$) o derecha ($\mathsf{inr}$), eligiendo el caso correcto $s$ o $t$ mediante la sustitución del parámetro $x$ o $y$ por el valor particular analizado $r$.

Una vez presentadas las bases del cálculo $\lambda$, el siguiente paso es relacionar éste con el sistema de deducción natural para así lograr un formalismo que permita hablar de pruebas y programas al mismo tiempo. Esto se logrará en la siguiente sección.

## 2.6. La lógica AF2

A grandes rasgos, la lógica AF2 es un sistema deductivo para la lógica de segundo orden que cuenta con codificación de pruebas y un razonamiento ecuacional. Antes de definirla hay que establecer el sistema de codificación o representación de pruebas, para lo cual se utiliza el cálculo lambda.

La teoría de la demostración es una rama de la lógica que se encarga de estudiar la estructura de las pruebas, comparar e identificar pruebas, así como de distinguir una prueba de otra. Para llevar a cabo esas acciones se desea elegir una notación conveniente para las pruebas, por lo cual se usará el cálculo $\lambda$. A cada prueba de una fórmula $A$, se le asociará un código único mediante un término del cálculo lambda, esta asociación se denota con $t : A$ y se define de acuerdo a la interpretación BHK definida en la subsección 2.3.Puesto que las pruebas dependen de hipótesis particulares, en realidad a partir de la relación binaria $\Gamma \vdash A$ obtenemos una nueva relación ternaria $\Gamma \vdash t : A$ cuyo significado es que la fórmula $A$ es derivable a partir de las hipótesis $\Gamma$, siendo el $\lambda$-término $t$ un código de prueba para la derivación original $\Gamma \vdash A$. En esta nueva relación los contextos de hipótesis son de la forma $\Gamma = \{x_1 : A_1, \ldots, x_n : A_n\}$, es decir, en este caso no sólo se supone una fórmula $A_i$, sino también se supone la existencia de una prueba $x_i$ de $A_i$, la cual por ser desconocida se codifica mediante una variable del cálculo $\lambda$. Por ejemplo para la implicación las reglas son:

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B}\ (\to I) \qquad \frac{\Gamma \vdash f : A \to B \quad \Gamma \vdash t : A}{\Gamma \vdash f\, t : B}\ (\to E)$$

Obsérvese que en el nuevo sistema deductivo figuran tanto los términos de la lógica como los $\lambda$-términos. Sin embargo se usa la misma notación para los términos de la lógica y para los $\lambda$-términos. Por ejemplo en el secuente $\vdash \lambda x.x : P(x) \to P(x)$ las presencias de $x$ antes de los dos puntos no tienen relación alguna con las de la implicación que está después de los dos puntos.

Al utilizar al cálculo $\lambda$ como un sistema notacional para codificar las pruebas, su semántica operacional juega un papel importante, puesto que permite simplificar pruebas redundantes. Considérese el problema de construir una prueba de $\Gamma \vdash B$, dadas las derivaciones $\Gamma \vdash \lambda x.e : A \to B$ y $\Gamma \vdash r : A$. La solución que salta a la vista es aplicar el modus ponens (regla $(\to E)$) con lo que se obtiene la derivación $\Gamma \vdash (\lambda x.e)r : B$. Sin embargo dado el determinismo de las reglas de inferencia y la prueba dada $\Gamma \vdash \lambda x.e : A \to B$, se sabe que previamente debió obtenerse la prueba $\Gamma, x : A \vdash e : B$ puesto que la regla usada como último paso de la derivación fue $(\to I)$ necesariamente. Ahora bien, nótese que como ya se tiene una prueba $\Gamma \vdash r : A$, no es necesario suponer $x : A$ puesto que en la prueba codificada por $e$ se puede sustituir cada prueba $x$ de $A$ por la prueba dada $r$, obteniendo así una prueba de $B$ sin necesidad de usar el modus ponens, a saber la prueba $\Gamma \vdash e[x := r] : B$. Este proceso es general por lo que siempre podemos reducir la prueba $(\lambda x.e)r$ a la prueba $e[x := r]$, así, ambas pruebas pueden considerarse equivalentes, pero esta equivalencia es parte de la $\beta$-equivalencia, es decir $(\lambda x.e)r =_\beta e[x := r]$. Las mismas consideraciones son válidas para la extensión con pares e inyecciones. Con esta idea ya es posible definir el sistema deductivo AF2.

La sintaxis de la lógica AF2 es la presentada al inicio del capítulo, mientras que su sistema deductivo se obtiene del sistema de deducción natural de la sección 2.4 mediante la anotación de pruebas con $\lambda$-términos.

- Regla de inicio:

$$\frac{}{\Gamma, x : A \vdash x : A}\ (Var)$$

- Reglas para los conectivos:

$$\frac{\Gamma, x : A \vdash r : B}{\Gamma \vdash \lambda x.r : A \to B}\ (\to I) \qquad \frac{\Gamma \vdash r : A \to B \quad \Gamma \vdash s : A}{\Gamma \vdash rs : B}\ (\to E)$$

$$\frac{\Gamma \vdash r : A \quad \Gamma \vdash s : B}{\Gamma \vdash \langle r, s\rangle : A \wedge B}\ (\wedge I) \quad \frac{\Gamma \vdash s : A \wedge B}{\Gamma \vdash \mathsf{fst}\, s : A}\ (\wedge E) \quad \frac{\Gamma \vdash s : A \wedge B}{\Gamma \vdash \mathsf{snd}\, s : B}\ (\wedge E)$$

$$\frac{\Gamma \vdash r : A}{\Gamma \vdash \mathsf{inl}\, r : A \vee B}\ (\vee I) \quad \frac{\Gamma \vdash r : B}{\Gamma \vdash \mathsf{inr}\, r : A \vee B}\ (\vee I)$$

$$\frac{\Gamma \vdash r : A \vee B \quad \Gamma, x : A \vdash s : C \quad \Gamma, y : B \vdash t : C}{\Gamma \vdash \mathsf{case}(r, x.s, y.t) : C}\ (\vee E)$$

- Reglas para los cuantificadores:

$$\frac{\Gamma \vdash t : A \quad x \notin FV(\Gamma)}{\Gamma \vdash t : \forall x A}\ (\forall I) \qquad \frac{\Gamma \vdash t : \forall x A}{\Gamma \vdash t : A[x := r]}\ (\forall E)$$

$$\frac{\Gamma \vdash t : A \quad X \notin FV(\Gamma)}{\Gamma \vdash t : \forall X A}\ (\forall^2 I) \qquad \frac{\Gamma \vdash t : \forall X A}{\Gamma \vdash t : A[X := \mathcal{P}]}\ (\forall^2 E)$$

  Las reglas anteriores son las versiones con pruebas anotadas del sistema de deducción natural para la lógica de segundo orden presentados en 2.4. Nótese que en el caso de los cuantificadores el término $t$ asociado a $A$ es el mismo que para la cuantificación $\forall x A$ o $\forall X A$, lo cual refleja que las pruebas de cuantificaciones no tienen un contenido computacional.

  La última regla es característica de AF2 e incorpora el razonamiento ecuacional directamente en la lógica.

- Regla ecuacional:

$$\frac{\Gamma \vdash t : A[x := r] \quad \mathbb{E} \rhd r = s}{\Gamma \vdash t : A[x := s]}\ (Eq)$$

En algunas ocasiones, al utilizar la regla ecuacional en una derivación $\Gamma \vdash t : A$ es importante observar la dependencia de la misma con respecto a un conjunto particular de ecuaciones $\mathbb{E}$, esto se hará mediante la notación $\Gamma \vdash_{\mathbb{E}} t : A$.

En la siguente sección se muestran ejemplos de derivaciones en la lógica AF2 que conllevan la justificación de la definición de algunos de los $\lambda$-términos dados en los ejemplos 4 y 5.

### 2.6.1. Extracción de programas en AF2

En esta sección se ejemplifica el método de extracción de programas ($\lambda$-términos) a partir de pruebas en AF2 desarrollado en los trabajos de Krivine, Parigot y Miranda Perea [4, 5, 16, 12]. La idea a grandes rasgos es la siguiente: sea $f : A \to B$ una función especificada mediante un conjunto de ecuaciones $\mathbb{E}$ y tal que existan tipos de datos formales (predicados de índice 1 con ciertas propiedades semánticas) $\mathcal{A}$ y $\mathcal{B}$ que representen a los conjuntos $A$ y $B$. Para hallar un programa ($\lambda$-término) $t$ que implemente a $f$ basta con mostrar formalmente que $\vdash_{\mathbb{E}} t : \forall x.\mathcal{A}(x) \to \mathcal{B}(f(x))$.

Enseguida se muestran algunos ejemplos.

**Ejemplo 6** (Valores Booleanos)**.** *Sean* true, false *dos constantes representando a los valores booleanos. Definimos el predicado por comprensión correspondiente al tipo booleano como*

$$\mathbb{B}(x) = \{x \mid \forall X.X(\mathsf{true}) \to X(\mathsf{false}) \to X(x)\}$$

*Esta definición es similar a la de los números naturales (ejemplo 1) y proviene del principio de inducción para booleanos. Ahora ya se puede saber de dónde surgen los $\lambda$-términos para los valores de verdad del ejemplo 4.*

$$\begin{array}{lll}
1. & x : X(\mathsf{true}), y : X(\mathsf{false}) \vdash X(\mathsf{t}) & (Var) \\
2. & x : X(\mathsf{true}) \vdash \lambda y.x : X(\mathsf{false}) \to X(\mathsf{t}) & (\to I), 1 \\
3. & \vdash \lambda x.\lambda y.x : X(\mathsf{true}) \to X(\mathsf{false}) \to X(\mathsf{t}) & (\to I), 2 \\
4. & \vdash \lambda x.\lambda y.x : \forall X.X(\mathsf{true}) \to X(\mathsf{false}) \to X(\mathsf{t}) & (\forall^2 I), 3
\end{array}$$

*Por lo tanto se prueba que* $\vdash \mathsf{true} : \mathbb{B}(\mathsf{true})$*. Similarmente se obtiene que* $\vdash \mathsf{false} : \mathbb{B}(\mathsf{false})$*.*

El siguiente ejemplo hace uso del razonamiento ecuacional.

**Ejemplo 7** (Conjunción)**.** *Sean* and *un símbolo de función binario y* $\mathbb{E} = \{\mathsf{and}(\mathsf{true}, x) = x,\ \mathsf{and}(\mathsf{false}, x) = \mathsf{false}\}$ *y* $\mathcal{F} =_{def} \{z \mid X(\mathsf{and}(z, y))\}$*. Hay que mostrar que*

$$\vdash \mathsf{and} : \forall x \forall y.\mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{and}(x, y))$$

*donde* and *es el $\lambda$-término definido en el ejemplo 4. Sea* $\Gamma = \{x : \mathbb{B}(x), y : \mathbb{B}(y),\ z : X(\mathsf{true}), w :$

$X(\mathsf{false})\}$.

| | | |
|---|---|---|
| 1. | $\Gamma \vdash y : \mathbb{B}(y)$ | $(Var)$ |
| 2. | $\Gamma \vdash y : X(\mathsf{true}) \to X(\mathsf{false}) \to X(y)$ | $(\forall^2 E), 1, [X := X]$ |
| 3. | $\Gamma \vdash z : X(\mathsf{true})$ | $(Var)$ |
| 4. | $\Gamma \vdash yz : X(\mathsf{false}) \to X(y)$ | $(\to E), 2, 3$ |
| 5. | $\Gamma \vdash w : X(\mathsf{false})$ | $(Var)$ |
| 6. | $\Gamma \vdash yzw : X(y)$ | $(\to E), 4, 5$ |
| 7. | $\Gamma \vdash_{\mathbb{E}} yzw : X(\mathsf{and}(\mathsf{true}, y))$ | $(Eq), 6$ |
| 8. | $\Gamma \vdash x : \mathbb{B}(x)$ | $(Var)$ |
| 9. | $\Gamma \vdash x : \mathcal{F}(\mathsf{true}) \to \mathcal{F}(\mathsf{false}) \to \mathcal{F}(x)$ | $(\forall^2 E), 8, [X := \mathcal{F}]$ |
| 10. | $\Gamma \vdash x(yzw) : \mathcal{F}(\mathsf{false}) \to \mathcal{F}(x)$ | $(\to E), 9, 7$ |
| 11. | $\Gamma \vdash w : X(\mathsf{and}(\mathsf{false}, y))$ | $(Eq), 5$ |
| 12. | $\Gamma \vdash x(yzw)w : \mathcal{F}(x)$ | $(\to E), 10, 11$ |
| 13. | $x : \mathbb{B}(x), y : \mathbb{B}(y),\ z : X(\mathsf{true}) \vdash \lambda w.x(yzw)w : X(\mathsf{false}) \to \mathcal{F}(x)$ | $(\to I), 12$ |
| 14. | $x : \mathbb{B}(x), y : \mathbb{B}(y) \vdash \lambda z.\lambda w.x(yzw)w : X(\mathsf{true}) \to X(\mathsf{false}) \to \mathcal{F}(x)$ | $(\to I), 12$ |
| 15. | $x : \mathbb{B}(x), y : \mathbb{B}(y) \vdash \lambda z.\lambda w.x(yzw)w : \forall X.X(\mathsf{true}) \to X(\mathsf{false}) \to X(\mathsf{and}(x, y))$ | $(\forall^2 I), 14$ *y def. de* $\mathcal{F}$ |
| 16. | $x : \mathbb{B}(x) \vdash \lambda y.\lambda z.\lambda w.x(yzw)w : \mathbb{B}(y) \to \mathbb{B}(\mathsf{and}(x, y))$ | $(\to I), 15$ *y def. de* $\mathbb{B}$ |
| 17. | $\vdash \lambda x.\lambda y.\lambda z.\lambda w.x(yzw)w : \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{and}(x, y))$ | $(\to I), 16$ |
| 18. | $\vdash \lambda x.\lambda y.\lambda z.\lambda w.x(yzw)w : \forall y.\mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{and}(x, y))$ | $(\forall I), 17$ |
| 19. | $\vdash \mathsf{and} : \forall x.\forall y.\mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{and}(x, y))$ | $(\forall I), 18$ |

Pasamos ahora a mostrar las derivaciones de los númerales de Church y de la suma de números naturales dadas en el ejemplo 5.

**Ejemplo 8** (Numerales de Church)**.** *La representación de los números naturales en el cálculo lambda dada en el ejemplo 5 se obtiene similarmente al caso de los booleanos. Como ejemplo se tiene la extracción del numeral* $\bar{2}$*, es decir, se muestra que* $\vdash \bar{2} : \mathbb{N}(s(s(0)))$*, donde la definición de* $\mathbb{N}$ *es la dada en el ejemplo 1*

| | | |
|---|---|---|
| 1. | $x : \forall x(X(x) \to X(s(x))),\ y : X(0) \vdash x : \forall x(X(x) \to Xs(x))$ | $(Var)$ |
| 2. | $x : \forall x(X(x) \to X(s(x))),\ y : X(0) \vdash x : X(0) \to X(s(0))$ | $(\forall E)\ 1,\ [x := 0]$ |
| 3. | $x : \forall x(X(x) \to X(s(x))),\ y : X(0) \vdash x : X(s(0)) \to X(s(s(0)))$ | $(\forall E)\ 1,\ [x := s(0)]$ |
| 4. | $x : \forall x(X(x) \to X(s(x))),\ y : X(0) \vdash y : X(0)$ | $(Var)$ |
| 5. | $x : \forall x(X(x) \to X(s(x))),\ y : X(0) \vdash xy : X(s(0))$ | $(\to E)\ 2, 4$ |
| 6. | $x : \forall x(X(x) \to X(s(x))),\ y : X(0) \vdash x(xy) : X(s(s(0)))$ | $(\to E)\ 3, 5$ |
| 7. | $x : \forall x(X(x) \to X(s(x))) \vdash \lambda y.x(xy) : X(0) \to X(s(s(0)))$ | $(\to I)\ 6$ |
| 8. | $\vdash \lambda x.\lambda y.x(xy) : \forall x(X(x) \to X(s(x))) \to X(0) \to X(s(s(0)))$ | $(\to I)\ 7$ |
| 9. | $\vdash \bar{2} : \mathbb{N}(s(s(0)))$ | $(\forall^2 I)\ 8$ |

En el siguiente ejemplo el uso de razonamiento ecuacional es indispensable.

**Ejemplo 9** (Suma de números naturales)**.** *Sean* $\mathsf{sum}$ *un símbolo de función binario,* $\mathbb{E} = \{\mathsf{sum}(x, 0) = x,\ \mathsf{sum}(x, s(y)) = s(\mathsf{sum}(x, y))\}$*. Vamos a mostrar que*

$$\vdash \mathsf{suma} : \forall x.\forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathbb{N}(\mathsf{sum}(x, y))$$

*donde* $\mathsf{suma}$ *es el* $\lambda$*-término definido en el ejemplo 5. La prueba requiere del predicado por comprensión* $\mathcal{F} =_{def} \{w \mid \mathbb{N}(\mathsf{sum}(x, w))\}$*.*

*Primero demostremos que*

$$\vdash \lambda u \lambda f \lambda z.f(ufz) : \forall z.\mathcal{F}(z) \to \mathcal{F}(s(z)) \qquad (**)$$

*para lo cual se considera* $\Gamma = \{u : \mathcal{F}(z),\ f : \forall x.X(x) \to X(s(x)),\ z : X(0)\}$.

| | | |
|---|---|---|
| 1. | $\Gamma \vdash f : \forall x(X(x) \to X(s(x)))$ | $(Var)$ |
| 2. | $\Gamma \vdash u : \mathbb{N}(\mathsf{sum}(x,z))$ | $(Var)$ *y def. de* $\mathcal{F}$ |
| 3. | $\Gamma \vdash z : X(0)$ | $(Var)$ |
| 4. | $\Gamma \vdash u : \forall x(X(x) \to X(s(x))) \to X(0) \to X(\mathsf{sum}(x,z))$ | $(\forall^2 E), 2, [X := X]$ |
| 5. | $\Gamma \vdash uf : X(0) \to X(\mathsf{sum}(x,z))$ | $(\to E), 4, 1$ |
| 6. | $\Gamma \vdash ufz : X(\mathsf{sum}(x,z))$ | $(\to E), 5, 3$ |
| 7. | $\Gamma \vdash f : X(\mathsf{sum}(x,z)) \to X(s(\mathsf{sum}(x,z)))$ | $(\forall E), 1, [x := \mathsf{sum}(x,z)]$ |
| 8. | $\Gamma \vdash f(ufz) : X(s(\mathsf{sum}(x,z)))$ | $(\to E), 7, 6$ |
| 9. | $\Gamma \vdash_{\mathbb{E}} f(ufz) : X(\mathsf{sum}(x,s(z)))$ | $(Eq), 8$ |
| 10. | $u : \mathcal{F}(z),\ f : \forall x(X(x) \to X(s(x))) \vdash \lambda z.f(ufz) : X(0) \to X(\mathsf{sum}(x,s(z)))$ | $(\to I), 9$ |
| 11. | $u : \mathcal{F}(z) \vdash \lambda f.\lambda z.f(ufz) : \forall x(X(x) \to X(s(x))) \to X(0) \to X(\mathsf{sum}(x,s(z)))$ | $(\to I), 10$ |
| 12. | $u : \mathcal{F}(z) \vdash \lambda f.\lambda z.f(ufz) : \mathbb{N}(\mathsf{sum}(x,s(z)))$ | $(\forall^2 I), 11$ *y def. de* $\mathbb{N}$ |
| 13. | $\vdash \lambda u.\lambda f.\lambda z.f(ufz) : \mathcal{F}(z) \to \mathcal{F}(s(z))$ | $(\to I), 12$ *y def. de* $\mathcal{F}$ |
| 14. | $\vdash \lambda u.\lambda f.\lambda z.f(ufz) : \forall z.\mathcal{F}(z) \to \mathcal{F}(s(z))$ | $(\forall I), 13$ |

*Para obtener la derivación principal se tiene* $\Gamma = \{x : \mathbb{N}(x), y : \mathbb{N}(y)\}$

| | | |
|---|---|---|
| 1. | $\Gamma \vdash_{\mathbb{E}} \lambda u.\lambda f.\lambda z.f(ufz) : \forall z.\mathcal{F}(z) \to \mathcal{F}(s(z))$ | $(Mon)$ *en* $(**)$ |
| 2. | $\Gamma \vdash y : \forall X.\forall z(X(z) \to X(s(z))) \to X(0) \to X(y)$ | $(Var)$ |
| 3. | $\Gamma \vdash y : \forall z(\mathcal{F}(z) \to \mathcal{F}(s(z))) \to \mathcal{F}(0) \to \mathcal{F}(y)$ | $(\forall^2 E), 2, [X := \mathcal{F}]$ |
| 4. | $\Gamma \vdash y(\lambda u.\lambda f.\lambda z.f(ufz)) : \mathcal{F}(0) \to \mathcal{F}(y)$ | $(\to E), 3, 1$ |
| 5. | $\Gamma \vdash x : \mathbb{N}(x)$ | $(Var)$ |
| 6. | $\Gamma \vdash_{\mathbb{E}} x : \mathbb{N}(\mathsf{sum}(x,0))$ | $(Eq), 5$ |
| 7. | $\Gamma \vdash y(\lambda u.\lambda f.\lambda z.f(ufz))x : \mathcal{F}(y)$ | $(\to E), 4, 6$ *y def. de* $\mathcal{F}$ |
| 8. | $x : \mathbb{N}(x) \vdash \lambda y.y(\lambda u.\lambda f.\lambda z.f(ufz))x : \mathbb{N}(y) \to \mathcal{F}(y)$ | $(\to I), 7$ |
| 9. | $\vdash \lambda x \lambda y.y(\lambda u.\lambda f.\lambda z.f(ufz))x : \mathbb{N}(x) \to \mathbb{N}(y) \to \mathcal{F}(y)$ | $(\to I), 8$ |
| 10. | $\vdash \lambda x \lambda y.y(\lambda u.\lambda f.\lambda z.f(ufz))x : \forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathcal{F}(y)$ | $(\forall I), 9$ |
| 11. | $\vdash \mathsf{suma} : \forall x.\forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathbb{N}(\mathsf{sum}(x,y))$ | $(\forall I), 10$ *y def. de* $\mathcal{F}$ |

Para finalizar, se mencionan dos propiedades importantes que muestran las repercusiones de la semántica operacional del cálculo lambda en las derivaciones de la lógica.

**Proposición 2.4** (Preservación de pruebas)**.** Si $\Gamma \vdash t : A$ y $t \to_{\beta}^{\star} t'$ entonces $\Gamma \vdash t' : A$.

*Demostración.* La prueba de esta importante propiedad no es trivial puesto que las reglas para los cuantificadores y el razonamiento ecuacional no son rastreables mediante los códigos de prueba. Una prueba puede consultarse en [4]. □

Esta propiedad implica que los reductos de un $\lambda$-término que codifica a una prueba de una fórmula $A$ siguen siendo códigos para pruebas de $A$.

**Proposición 2.5** (Normalización fuerte)**.** Si $\Gamma \vdash t : A$ entonces $t$ es un $\lambda$-término fuertemente normalizable, es decir, existe un $\lambda$-término $e$ que no es $\beta$-reducible y tal que $t \to_\beta^\star e$.

*Demostración.* La prueba consiste en mostrar que $\mathsf{AF2}$ se encaja en la lógica proposicional de segundo orden (ver [14]) la cual se sabe que es fuertemente normalizable (ver [4]). $\square$

Esta propiedad implica que si un $\lambda$-término resulta ser un código de prueba entonces puede simplificarse de manera exhaustiva, lo cual junto con la propiedad de preservación implica que una prueba de una fórmula $A$ siempre puede simplificarse hasta conseguir una prueba llamada normal o canónica que resulta ser la prueba más simple de $A$.

# Capítulo 3

# La lógica $\mathsf{AF2}^{M\mu\nu}$

En este capítulo se presenta la lógica $\mathsf{AF2}^{M\mu\nu}$, la cual es una extensión de la lógica AF2 que además incluye definiciones (co)inductivas basadas en los principios de Mendler presentadas en [11]. Adicionalmente hemos desarrollado un método de definición por cláusulas, el cual es cercano a los mecanismos usuales en programación funcional.

Aunque la lógica AF2 permite obtener programas para cualquier función que involucre tipos de datos usuales en computación, en muchos casos éstos no tienen el comportamiento deseado. Por ejemplo, la única manera de definir la función predecesor en naturales produce un programa con tiempo de ejecución lineal, cuando lo que se espera es tiempo de ejecución constante (ver [15]). Para resolver este problema se han definido algunas extensiones de AF2 con definiciones inductivas como TTR (ver [16]) y también con definiciones coinductivas como $\mathsf{AF2}^{\mu\nu}$ (ver [18]).

Sin embargo, estas lógicas tienen como desventaja la pérdida de la normalización fuerte debido al uso de un combinador de punto fijo en el sistema de términos-prueba, con el cual se codifican derivaciones con lambda-términos. Lo que sucede es que una función iterativa $\mathsf{f}$ puede ser definida dentro de AF2 y por lo tanto la propiedad de normalización fuerte de AF2 garantiza que el programa $\overline{\mathsf{f}}$ correspondiente a la función $\mathsf{f}$ termina; pero el programa extraído para una función recursiva primitiva, por ejemplo el predecesor que se ejecuta en tiempo constante, emplea un combinador de punto fijo en las extensiones de AF2 y, por tanto, su terminación no es del todo obvia. Ello ha dado pie al empleo de sofisticados métodos para verificar que los programas de ese tipo efectivamente terminan (ver [8]).

En contraste, en el presente trabajo se propone la lógica $\mathsf{AF2}^{M\mu\nu}$. Ésta es una nueva extensión de AF2 con principios nativos de (co)recursión primitiva que además posee la propiedad de normalización fuerte y cuya característica principal es el uso de los principios de Mendler en vez de los principios convencionales. Esto resuelve en particular el problema de extracción de programas que involucran definiciones coinductivas como streams o árboles infinitos, el cual resulta difícil de resolver con principios convencionales.

Una de las principales motivaciones para proponer la lógica $\mathsf{AF2}^{M\mu\nu}$ consiste en establecer una sintaxis similar a la que se usa para declarar tipos de datos en los lenguajes de programación funcionales y así, al tener un referente más familiar, poder programar con pruebas más fácilmente.

En general, un tipo de datos recursivo se declara en los lenguajes funcionales del siguiente modo:

**Definición 3.1.** [Declaración de un tipo en un lenguaje funcional] Un tipo recursivo $D$ con parámetros $a_1 \ldots a_i$ se define como sigue:

$$data\ D\ a_1 \ldots a_i\ =\ C_1\ a_{11}\ \ldots\ a_{1k_1}\ |\ \ldots\ |\ C_n\ a_{n1}\ \ldots\ a_{nk_n}$$

donde $D$ es el nombre del tipo de dato a definir, $a_i$ es una variable de tipo y $C_i$ es un constructor de elementos de $D$ que recibe como entrada elementos de tipo $a_{i1} \ldots a_{ik_i}$, es decir, $C_i$ es una función cuya signatura es

$$C_i\ ::\ a_{i1}\ \to\ a_{i2}\ \ldots \to\ a_{ik_i}\ \to\ \ D\ a_1\ \ldots\ a_n$$

Si algún constructor $C_i$ no recibe parámetros de entrada, convenimos en que su signatura sea $C_i\ ::\ D\ a_1\ \ldots\ a_n$.

Algunos ejemplos comunes son:

- El tipo de valores opcionales de A, definido como

  $$data\ Maybe\ a\ =\ Nothing\ |\ Just\ a$$

  donde $Nothing :: Maybe\ a$ y $Just :: a\ \to\ \ Maybe\ a$.

- El tipo de números naturales definido como

  $$data\ Nat\ =\ Zero\ |\ Suc\ Nat$$

  donde $0 :: Nat$ y $suc :: Nat\ \to\ Nat$.

- El tipo de listas finitas sobre A definido como

  $$data\ List\ a\ =\ Nil\ |\ Cons\ a\ (List\ a)$$

  donde $Nil :: List\ a$ y $Cons :: a\ \to\ List\ a\ \to\ List\ a$.

- El tipo de árboles binarios no vacíos con nodos etiquetados por A definido como

  $$data\ BTree\ a\ =\ Leaf\ a\ |\ Node\ a\ (BTree\ a)\ (BTree\ a)$$

  donde $Leaf :: BTree\ a$ y $Node :: a\ \to\ BTree\ a\ \to\ (BTree\ a)\ \to\ (BTree\ a)$.

Como ya hemos observado, esta clase de definiciones corresponden a ecuaciones de punto fijo cuya solución es el mínimo punto fijo de un operador. También resulta natural preguntarse a qué corresponden, en programación funcional, los máximos puntos fijos de un operador. La respuesta se obtiene en los llamados tipos correcursivos, cuyo mecanismo de definición no esta contemplado directamente en los lenguajes funcionales actuales, pero puede modelarse haciendo uso de la evaluación perezosa, por ejemplo en HASKELL.

Esta clase de tipos de datos permite definir estructuras infinitas o potencialmente infinitas razonando con destructores, de manera dual al caso de tipos recursivos cuya definición se sirve de constructores. A continuación definimos una declaración para tipos correcursivos, inspirada en la declaración para tipos recursivos recién definida.

**Definición 3.2.** [Declaración de un tipo correcursivo] Un tipo correcursivo $S$ de $a_1 \dots a_k$ se define como sigue

$$codata\ S\ a_1 \dots a_k \ = \ D_1\ b_1 \mid \ \dots \ \mid D_n\ b_n$$

donde $S$ es el nombre del tipo de datos a definir, $a_i$ es una variable de tipo y $D_i$ es un destructor de elementos, el cual toma un elemento de $S$ y devuelve un elemento de $b_i$, es decir, $D_i$ es una función cuya signatura es $D_i :: S\ a_1\ \dots\ a_k \rightarrow b_i$.

Algunos ejemplos comunes de tipos correcursivos son:

- El tipo de listas estrictamente infinitas de A, definido como

  $$codata\ Stream\ a \ = \ Head\ a \mid Tail\ (Stream\ a)$$

  donde $Head :: Stream\ a \ \rightarrow \ a$ y $Tail :: Stream\ a \ \rightarrow \ Stream\ a$.

- El tipo de colistas [1] no vacías de A definido como

  $$codata\ CoList\ a = \ Head\ a \mid Tail\ (Maybe\ (Colist\ a)\ )$$

  donde $Head :: Colist\ a \ \rightarrow \ a$ y $Tail :: Colist\ a \ \rightarrow \ Maybe\ (Colist\ a)$.

- El tipo producto de A y B definido como

  $$codata\ Prod\ a\ b = \ Fst\ a \mid Snd\ b$$

  donde $Fst :: Prod\ a\ b \ \rightarrow \ a$ y $Snd :: Prod\ a\ b \ \rightarrow \ b$.

- El tipo de árboles binarios completos estrictamente infinitos con nodos etiquetados por A, definido como

  $$codata\ InfBTree\ a = \ Label\ a \mid Leftsbt\ (InfbTree\ a) \mid Rightsbt\ (InfbTree\ a)$$

  donde $Label :: InfBTree\ a \ \rightarrow \ a$, $Leftsbt :: InfBTree\ a \ \rightarrow \ InfBTree\ a$ y
  $Rightbt :: InfBTree\ a \ \rightarrow \ InfBTree\ a$.

[1] Una colista es una lista potencialmente finita, es decir, una lista que puede ser finita o infinita

## 3.1. Cláusulas y operadores sintácticos

Como se observa de los ejemplos anteriores, una definición data o codata consta de varias partes separadas por el símbolo |, siendo cada una de ellas la definición de un constructor o un destructor. Nuestro propósito es capturar este mecanismo de definición en una lógica que permita definir predicados inductivos y coinductivos. Para esto nos serviremos de los conceptos de cláusulas y operadores sintácticos discutidos a continuación.

**Definición 3.3.** Una cláusula $\mathcal{C}$ es una tupla de la forma

$$\langle \mathcal{F}_1^{(1)}, \ldots, \mathcal{F}_k^{(1)}, \mathsf{c}^{(k)} \rangle$$

donde $\mathcal{F}_i$ es un predicado de índice 1, es decir de un argumento y $\mathsf{c}$ es un símbolo de función de índice $k$ llamado etiqueta. Si $k = 0$ la cláusula correspondiente se denota como:

$$\langle -, \mathsf{c}^{(0)} \rangle$$

donde $-$ es un símbolo que indica que no hay predicados y $\mathsf{c}$ es un símbolo de función de índice 0, es decir, un símbolo de constante.

Utilizando el concepto de cláusula podemos definir una expresión formal llamada operador sintáctico la cual juega el papel de un operador que servirá para definir los predicados inductivos y coinductivos.

**Definición 3.4.** Un operador sintáctico $\Phi$ es una expresión de la forma

$$\Phi =_{def} \lambda X.(\mathcal{C}_1, \ldots, \mathcal{C}_k)$$

donde $X$ es una variable de segundo orden cuyo índice es 1 y $\mathcal{C}_i$ son cláusulas.

Intuitivamente, un operador sintáctico es una función que transforma un predicado en tuplas de predicados, por lo que es importante definir con detalle la aplicación de un operador sintáctico a un predicado $\mathcal{P}$.

**Definición 3.5.** Dado un operador $\Phi$ y un predicado $\mathcal{P}$, la aplicación de $\Phi$ a $\mathcal{P}$, denotada $\Phi(\mathcal{P})$ o simplemente $\Phi\,\mathcal{P}$, se define como la tupla

$$\Phi(\mathcal{P}) =_{def} (\mathcal{C}_1[X := \mathcal{P}], \ldots, \mathcal{C}_n[X := \mathcal{P}])$$

donde si $\mathcal{C} =_{def} \langle \mathcal{F}_1, \ldots, \mathcal{F}_k, \mathsf{c} \rangle$ entonces

$$\mathcal{C}[X := \mathcal{P}] =_{def} \langle \mathcal{F}_1[X := \mathcal{P}], \ldots, \mathcal{F}_k[X := \mathcal{P}], \mathsf{c} \rangle.$$

Más aún, dada una aplicación $\Phi(\mathcal{P})$, definimos la aplicación de la cláusula $\mathcal{C}_i$ al predicado $\mathcal{P}$, como $\mathcal{C}_i\mathcal{P} =_{def} \Pi_{k,i}(\Phi(\mathcal{P}))$, donde $\Pi_{k,i}$ es la i-ésima proyección en $k$-tuplas.

A continuación definimos el concepto de predicado (co)inductivo usando cláusulas y operadores sintácticos.

**Definición 3.6.** Sean $X^{(1)}$ una variable de predicado de índice 1 y $\Phi$ un operador sintáctico. Una expresión de la forma $\mu(\Phi)$ se llama predicado inductivo. En este caso las etiquetas de cada cláusula del operador $\Phi$ son llamadas constructores. Análogamente un predicado coinductivo es una expresión de la forma $\nu(\Phi)$ y en este caso hablamos de destructores en lugar de etiquetas.

Como ejemplo mostramos la definición como predicados (co)inductivos de los tipos de datos (co)recursivos definidos anteriormente.

- $Maybe\ a\ =\ \mu\ (\lambda X.(\langle -, \mathsf{nothing}\rangle, \langle \mathcal{P}, \mathsf{just}\rangle)$
- $Nat\ =\ \mu\ (\lambda X.(\langle -, \mathsf{zero}\rangle, \langle X, \mathsf{suc}\rangle)$
- $List\ a\ =\ \mu\ (\lambda X.(\langle -, \mathsf{nil}\rangle, \langle \mathcal{A}, X, \mathsf{cons}\rangle)$
- $BTree\ a\ =\ \mu\ (\lambda X.(\langle -, \mathsf{node}\rangle, \langle X, X, \mathsf{mBT}\rangle)$
- $Stream\ A\ =\ \nu\ (\lambda X.(\langle A,\ Head\rangle\ ,\ \langle X,\ Tail\rangle\ )\ )$
- $Colist\ C\ =\ \nu\ (\lambda X.(\langle C,\ Head\rangle\ ,\ \langle Maybe\ X,\ Tail\rangle\ )\ )$
- $Producto\ A\ B\ =\ \nu\ (\lambda X.(\langle A,\ Fst\rangle\ ,\ \langle B,\ Snd\rangle\ )\ )$
- $InfBTree\ A\ =\ \nu\ (\lambda X.(\langle A,\ Label\rangle\ ,\ \langle X,\ LeftsBT\rangle,\ \langle X,\ RightsBT\rangle\ )\ )$

Ahora que ya contamos con un mecanismo sintáctico para definir predicados inductivos y coinductivos, necesitamos definir la lógica que se encargará de darle sentido a esta clase de predicados, de tal forma que se conserve el mecanismo de definición de las declaraciones data y codata. A continuación definimos esta lógica, llamada $\mathsf{AF2}^{M\mu\nu}$.

- *Términos*: los términos se definen a partir de una signatura $\Sigma$ incluyendo símbolos de función $f$ de una aridad dada.
$$t ::= x \mid f(t_1, \ldots, t_n)$$
- *Predicados*: además de los predicados de la lógica de segundo orden (variables de segundo orden, símbolos de predicado de una signatura $\Sigma$ o predicados por comprensión) agregamos predicados inductivos $\mu(\Phi)$ y predicados coinductivos $\nu(\Phi)$.
$$\mathcal{P} ::= X \mid P \mid \mathcal{F} \mid \mu(\Phi) \mid \nu(\Phi)$$
- *Fórmulas*: son las mismas que en la lógica de segundo orden
$$A, B ::= \mathcal{P}(t_1, \ldots, t_n) \mid A \to B \mid \forall x A \mid \forall X A$$

La lógica $\mathsf{AF2}^{M\mu\nu}$ hereda las reglas de inferencia para los operadores lógicos y el mecanismo de razonamiento ecuacional de la lógica $\mathsf{AF2}$. Adicionalmente requerimos las reglas de inferencia para los predicados inductivos y coinductivos, las cuales discutimos a continuación.

## 3.2. Reglas de inferencia para predicados (co)inductivos

Para que nuestros predicados (co)inductivos se comporten de la manera deseada, necesitamos definir reglas de inferencia que garanticen las propiedades semánticas de puntos fijos discutidas en el capítulo 1. Mas aún, dado que nuestro propósito principal es razonar con definiciones de funciones dadas por un conjunto de ecuaciones, así como obtener pruebas que involucran dichas definiciones, nuestras reglas de inferencia serán diseñadas especialmente para este propósito, lo cual es una aportación nuestra.

**Definición 3.7.** Sean $\mathcal{K}^{(1)}$ un predicado y $\mathcal{C} = \langle \mathcal{F}_1^{(1)}, \ldots, \mathcal{F}_k^{(1)}, \mathsf{c}^{(k)} \rangle$ una cláusula. Definimos la relación de inclusión $\mathcal{C} \subseteq \mathcal{K}$ como:

$$\mathcal{C} \subseteq \mathcal{K} =_{def} \forall x_1 \ldots \forall x_k . \mathcal{F}_1 x_1 \to \ldots \to \mathcal{F}_k x_k \to \mathcal{K}(\mathsf{c}\, x_1 \ldots x_k)$$

Es decir una clásula $\mathcal{C}$ esta contenida en un predicado $\mathcal{K}$ si y sólo si la imagen de cualesquiera elementos de $\mathcal{F}_i$ bajo el constructor $c$ es un elemento de $\mathcal{K}$.

Mas aún, si $f$ es un símbolo de función de índice 1, definimos la relación de inclusión de $\mathcal{C}$ en $\mathcal{K}$ con respecto a $f$, denotada $\mathcal{C} \subseteq_f \mathcal{K}$, como:

$$\mathcal{C} \subseteq_f \mathcal{K} =_{def} \forall x_1 \ldots \forall x_k . \mathcal{F}_1 x_1 \to \ldots \to \mathcal{F}_k x_k \to \mathcal{K}(f(\mathsf{c}\, x_1 \ldots x_k))$$

Las definiciones anteriores serán de utilidad cuando lidiemos con predicados inductivos. Para el caso de los predicados coinductivos se utilizarán las siguientes definiciones.

**Definición 3.8.** Sean $\mathcal{K}^{(1)}$ un predicado y $\mathcal{C} = \langle \mathcal{F}^{(1)}, \mathsf{c}^{(1)} \rangle$ una cláusula. La relación de inclusión $\mathcal{K} \subseteq \mathcal{C}$ se define como:

$$\mathcal{K} \subseteq \mathcal{C} =_{def} \forall x . \mathcal{K}x \to \mathcal{F}(\mathsf{c}\ x)$$

Es decir $\mathcal{K} \subseteq \mathcal{C}$ si y solo si la imagen bajo $\mathsf{c}$ de cualquier elemento de $\mathcal{K}$ es un elemento de $\mathcal{F}$.

Mas aún, si $f$ es un símbolo de función de índice 1, la relación de inclusión de $\mathcal{K}$ en $\mathcal{C}$ con respecto a $f$, denotada $\mathcal{K} \subseteq_f \mathcal{C}$, se define como:

$$\mathcal{K} \subseteq_f \mathcal{C} =_{def} \forall x . \mathcal{K}\ x \to \mathcal{F}(\mathsf{c}(f\ x))$$

Con ayuda de estos conceptos de inclusión podemos definir las reglas de inferencia necesarias, cuya definición se basa en las propiedades de la teoría de puntos fijos desarrolladas en la sección 1.3.

- *Construcción inductiva*:

$$\frac{\Gamma \vdash r_1 : \mathcal{F}_1(\mu(\Phi))(t_1)\ \ldots\ \Gamma \vdash r_k : \mathcal{F}_k(\mu(\Phi))(t_k) \quad \mathcal{C}_i = \langle \mathcal{F}_1, \ldots, \mathcal{F}_k, \mathsf{c} \rangle}{\Gamma \vdash \mathsf{in}_{k,i}\, r_1 \ldots r_k : \mu(\Phi)(\mathsf{c}\ t_1 \ldots t_k)}\ (\mu I)$$

- *Destrucción coinductiva*:

$$\frac{\Gamma \vdash r : \nu(\Phi)t \quad \mathcal{C}_i = \langle \mathcal{F}, \mathsf{c} \rangle}{\Gamma \vdash \mathsf{out}_{k,i}\, r : \mathcal{F}(\nu(\Phi))(\mathsf{c}\ t)}\ (\nu E)$$

- *Recursión primitiva*:

$$\frac{\Gamma \vdash s_i : \forall X \big( X \subseteq \mu(\Phi) \to X \subseteq_f \mathcal{K} \to \mathcal{C}_i X \subseteq_f \mathcal{K} \big) \quad \Gamma \vdash r : \mu(\Phi)(t)}{\Gamma \vdash \mathsf{MRec}\, \vec{s}\, r : \mathcal{K}(ft)}\ (\mu E)$$

- *Iteración*:

$$\frac{\Gamma \vdash s_i : \forall X\big(X \subseteq_f \mathcal{K} \to \mathcal{C}_i X \subseteq_f \mathcal{K}\big) \quad \Gamma \vdash r : \mu(\Phi)(t)}{\Gamma \vdash \mathsf{MIt}\, \vec{s}\, r : \mathcal{K}(ft)} \; (\mu E^-)$$

- *Correcursión primitiva*:

$$\frac{\Gamma \vdash s_i : \forall X\big(\nu(\Phi) \subseteq X \to \mathcal{K} \subseteq_f X \to \mathcal{K} \subseteq_f \mathcal{C}_i X\big) \quad \Gamma \vdash r : \mathcal{K}(t)}{\Gamma \vdash \mathsf{MCoRec}\, \vec{s}\, r : \nu(\Phi)(ft)} \; (\nu I)$$

- *Coiteración*:

$$\frac{\Gamma \vdash s_i : \forall X\big(\mathcal{K} \subseteq_f X \to \mathcal{K} \subseteq_f \mathcal{C}_i X\big) \quad \Gamma \vdash r : \mathcal{K}(t)}{\Gamma \vdash \mathsf{MCoIt}\, \vec{s}\, r : \nu(\Phi)(ft)} \; (\nu I^-)$$

- *Semántica operacional*:

$$\begin{array}{rcl}
(\lambda x.r)s & \mapsto_\beta & r[x := s] \\
\mathsf{MIt}\ \vec{s}\,(\mathsf{in}_{n,i}\ t) & \mapsto_\beta & s_i\ (\mathsf{MIt}\ \vec{s})\ t \\
\mathsf{out}_{n,i}\ (\mathsf{MCoIt}\ \vec{s}\, t) & \mapsto_\beta & s_i\ (\mathsf{MCoIt}\ \vec{s})\ t \\
\mathsf{MRec}\ \vec{s}\,(\mathsf{in}_{n,i}\ t) & \mapsto_\beta & s_i\ (\lambda xx)(\mathsf{MRec}\ \vec{s})\ t \\
\mathsf{out}_{n,i}\ (\mathsf{MCoRec}\ \vec{s}\, t) & \mapsto_\beta & s_i\ (\lambda xx)(\mathsf{MCoRec}\ \vec{s})\ t
\end{array}$$

Donde $\vec{s}$ denota a un vector de longitud $n$ de la forma $\vec{s} = s_1\ s_2\ \ldots\ s_n$. Las nuevas reglas de la semántica operacional se justifican de la manera usual, pues corresponden a la simplificación de pruebas que involucran la aplicación de una regla de introducción seguida inmediatamente por una regla de eliminación del mismo operador lógico, que en este caso es $\mu$ o $\nu$.

Nuestro propósito principal, como se verá en el siguiente capítulo, es utilizar esta lógica como base del paradigma de programación con pruebas. No obstante, consideramos de importancia mencionar algunas propiedades generales de la lógica.

**Proposición 3.1.** (Preservación de tipos para $\mathsf{AF2}^{M\mu\nu}$). Si $\Gamma \vdash_{\mathbb{E}} t : A$ y $t \to^* t'$ entonces $\Gamma \vdash_{\mathbb{E}} t' : A$.

*Demostración.* La prueba no es trivial y queda fuera de los alcances de este trabajo. Una prueba similar puede encontrarse en [13]. □

Esta propiedad garantiza que si el programa o código de prueba $t$ cuyo tipo es $A$ se simplifica por medio de la semántica operacional, el programa $t'$ resultado de esta simplificación seguirá siendo del mismo tipo $A$. Sin esta propiedad sería imposible garantizar la correctud del método de programación con pruebas.

**Proposición 3.2.** (Normalización fuerte para $\mathsf{AF2}^{M\mu\nu}$). Si $\Gamma \vdash_{\mathbb{E}} t : A$ entonces t es fuertemente normalizable.

Recordemos que un término $t$ es fuertemente normalizable si y solo sí cualquier proceso de simplificación de $t$ mediante la semántica operacional termina. Es decir no existe una sucesión infinita de pasos de reducción de la forma $t \to t_1 \to t_2 \to \ldots$ . En otras palabras, la propiedad de normalización fuerte garantiza que si $t$ es un programa obtenido a partir de la lógica, como lo serán todos los programas mostrados en el siguiente capítulo, entonces $t$ es un programa que no se cicla. Es decir, $t$ es un programa totalmente correcto.

# Capítulo 4

# Programación con pruebas

En este capítulo se muestra el método de programación con pruebas en el sistema $\mathsf{AF2}^{M\mu\nu}$, el cual permite la extracción automática de un programa (es decir un $\lambda$-término) a partir de una prueba de la especificación correspondiente.

El método consiste en los siguientes pasos:

- Se definen los tipos de datos involucrados en el problema mediante un predicado inductivo o coinductivo.
- Se especifica la solución del problema ecuacionalmente, definiendo una función $f$ mediante un conjunto de ecuaciones $\mathbb{E}$.
- Utilizando deducción natural y el razonamiento ecuacional, se construye una prueba que muestra que $f$ cumple la especificación del problema con respecto a la entrada y salida. Es decir, si la especificación define una función $f : A \to B$, debemos mostrar que

  $$\vdash_{\mathbb{E}} \forall x.\mathcal{A}(x) \to \mathcal{B}(f(x))$$

  Donde $\mathcal{A}$ y $\mathcal{B}$ son las implementaciones de los tipos $A$ y $B$.
- A partir de dicha prueba se obtiene, de manera automática, un término $t$, el cual es el programa deseado.

Por ejemplo, si el problema consiste en obtener un programa que calcule la longitud de una lista de números, es decir una función $long : \mathbb{L}_{\mathbb{N}} \to \mathbb{N}$, el método se desarrolla como sigue:

- Definimos los predicados para números naturales $\mathbb{N}$ y listas de números naturales $\mathbb{L}_{\mathbb{N}}$.
- Especificamos la función longitud:

  $$\mathbb{E} = \{long\ (nil) = 0,\ long\ (cons\ x\ l) = 1\ +\ long\ l\}.$$

- Obtenemos una prueba de

$$\vdash_{\mathbb{E}} t : \forall x.\mathbb{L}_{\mathbb{N}}(x) \to \mathbb{N}(long\ x).$$

- El $\lambda$-término $t$ obtenido automáticamente en la prueba anterior resulta ser un programa que calcula la longitud de una lista.

La justificación matemática de este método se sirve del siguiente lema:

**Lema 4.1.** [Corrección] Sean $\mathcal{A}$ y $\mathcal{B}$ predicados, $f$ un símbolo de función de índice 1 con dominio $\mathcal{A}$ y codominio $\mathcal{B}$, y $\mathbb{E}_f = \{s_1 = t_1, \ldots, s_n = t_n\}$ una especificación ecuacional para $f$. Sea $\mathcal{M} = \langle \Lambda, \mathcal{I} \rangle$ un modelo para la lógica $\mathsf{AF2}^{M\mu\nu}$, donde $\Lambda$ es el conjunto de $\lambda - términos$ de prueba módulo $\beta - equivalencia$. Si $\mathcal{M} \models \mathbb{E}_f$, es decir, si las ecuaciones de $\mathbb{E}_f$ son válidas en $\mathcal{M}$, y

$$\vdash_{\mathbb{E}_f} t : \forall x.\mathcal{A}(x) \to \mathcal{B}(f(x))$$

entonces, para cualquier término $s$ tal que $\mathcal{M} \models \mathcal{A}(s)$, se cumple que

$$\mathcal{M} \models \quad t\ s = f\ (s)$$

La demostración de este lema requiere de conceptos semánticos del cálculo lambda y de la lógica, como la noción de modelo para predicados (co)inductivos, que caen fuera del alcance de este trabajo y que pueden consultarse en [11]. La idea es que, siendo $\lambda - términos$ los individuos de $\mathcal{M}$, la igualdad corresponde a la $\beta - equivalencia$ por lo que la ecuación que relaciona a $t$ y a $f$ dada por el lema, en realidad atestigua que $t$ es un programa que calcula a $f$. Si bien no mostramos aquí una demostración, nuestro propósito principal en este capítulo es mostrar diversos ejemplos que muestran la utilidad del método propuesto y cuya correctud es consecuencia directa de este lema.

El resto del capítulo se encarga de dar diversos ejemplos del método de programación con pruebas que extraen programas que involucran predicados inductivos y coinductivos. En cada caso damos la definición del tipo de datos con predicados (co)inductivos, la especificación funcional (una declaración *data* o *codata*) y la implementación en la lógica $\mathsf{AF2}^{M\mu\nu}$, la cual se obtiene de manera automática, de acuerdo al lema 4.1, mediante la demostración formal de que la función es total. Adicionalmente se muestran los casos particulares de las reglas de inferencia para constructores o destructores, así como la implementación de los mismos. Finalmente se dan ejemplos de extracción de programas y en algunos casos se verifica el buen comportamiento de la semántica operacional.

## 4.1. Extracción de programas con tipos inductivos

En esta sección desarrollamos ejemplos que involucran a los siguientes tipos:

- Booleanos
- Números naturales

- Listas finitas sobre un tipo A
- El tipo opcional Maybe
- Árboles binarios sin etiquetas

### 4.1.1. Booleanos

- **Definición**. Los valores booleanos corresponden al conjunto

$$\mathbb{B} =_{def} \{T, F\}$$

- **Especificación funcional**

$$data\ Bool\ =\ True \mid False$$

- **Implementación**. Considérense las siguientes cláusulas tales que $\mathcal{C}_1 = \langle -, \mathsf{true}\rangle$, $\mathcal{C}_2 = \langle -, \mathsf{false}\rangle$. Se define el predicado inductivo de valores booleanos como:

$$\mathbb{B} =_{def} \mu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2)$$

- **Reglas de construcción**

$$\frac{}{\Gamma \vdash \mathsf{in}_{2,1} : \mathbb{B}(\mathsf{true})}\ (\mu I)$$

$$\frac{}{\Gamma \vdash \mathsf{in}_{2,2} : \mathbb{B}(\mathsf{false})}\ (\mu I)$$

- **Constructores**
  - $\overline{\mathsf{true}} =_{def} \mathsf{in}_{2,1}$

    *Demostración.* [1]

$$1.\quad \Gamma \vdash \mathsf{in}_{2,1} : \mathbb{B}(\mathsf{true}) \quad (\mu I)$$

$$\therefore\ \overline{\mathsf{true}} = \mathsf{in}_{2,1}$$

□

  - $\overline{\mathsf{false}} =_{def} \mathsf{in}_{2,2}$

    *Demostración.*

$$1.\quad \Gamma \vdash \mathsf{in}_{2,2} : \mathbb{B}(\mathsf{false}) \quad (\mu I)$$

$$\therefore\ \overline{\mathsf{false}} = \mathsf{in}_{2,2}$$

[1] En todos los ejemplos de este capítulo entendemos por demostración a una derivación formal en la lógica $\mathsf{AF2}^{M\mu\nu}$ de que el término propuesto cumple con (es una prueba de) la especificación deseada.

□

- **Semántica operacional**.

$$\begin{array}{rcl} \mathsf{MIt}\ \vec{s}\,(\mathsf{true}) & \mapsto_\beta & s_1 \\ \mathsf{MIt}\ \vec{s}\,(\mathsf{false}) & \mapsto_\beta & s_2 \end{array}$$

Un ejemplo de extracción de programas es:

- **Condicional**. Sea $\mathsf{ite}$ un símbolo de función y $\mathbb{E}_{\mathsf{ite}} = \{\mathsf{ite}\ \ \mathsf{true}\ \ i\ \ j = i,\ \ \mathsf{ite}\ \ \mathsf{false}\ \ i\ \ j = j\}$. Entonces:

$$\vdash \overline{\mathsf{ite}} : \forall b.\forall i.\forall j.\mathbb{B}(b) \to \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{ite}\ \ b\ i\ j)$$

donde $\overline{\mathsf{ite}} = \lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b,\ s_1 = \lambda p.i$ y $s_2 = \lambda p.j$.

*Demostración.* Sean $\mathsf{f}\ \ b =_{def} \mathsf{ite}\ \ b\ i\ j$ y $\Gamma =_{def} \{b : B(b),\ i : A(i)\ j : A(j)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathcal{A} \to \langle -, \mathsf{true}\rangle \subseteq_f \mathcal{A}\big)$

$$\begin{array}{lll} 1. & \Gamma,\ p : X \subseteq_f \mathcal{A} \vdash i : \mathcal{A}(i) & (Var) \\ 2. & \Gamma,\ p : X \subseteq_f \mathcal{A} \vdash_{\mathbb{E}_{\mathsf{ite}}} i : \mathcal{A}(\mathsf{f}(\mathsf{true})) & (Eq), 1 \\ 3. & \Gamma,\ p : X \subseteq_f \mathcal{A} \vdash i : \langle -, \mathsf{true}\rangle \subseteq_f \mathcal{A} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2 \\ 4. & \Gamma \vdash \lambda p.i : X \subseteq_f \mathcal{A} \to \langle X, \mathsf{true}\rangle \subseteq_f \mathcal{A} & (\to I), 3 \\ 5. & \Gamma \vdash \lambda p.i : \forall X\big(X \subseteq_f \mathcal{A} \to \langle X, \mathsf{true}\rangle \subseteq_f \mathcal{A}\big) & (\forall^2 I)), 4 \end{array}$$

$$\therefore\ s_1 = \lambda p.i$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathcal{A} \to \langle -, \mathsf{false}\rangle \subseteq_f \mathcal{A}\big)$

$$\begin{array}{lll} 1. & \Gamma,\ p : X \subseteq_f \mathcal{A} \vdash j : \mathcal{A}(j) & (Var) \\ 2. & \Gamma,\ p : X \subseteq_f \mathcal{A} \vdash_{\mathbb{E}_{\mathsf{ite}}} j : \mathcal{A}(\mathsf{f}(\mathsf{false})) & (Eq), 1 \\ 3. & \Gamma,\ p : X \subseteq_f \mathcal{A} \vdash j : \langle -, \mathsf{false}\rangle \subseteq_f \mathcal{A} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2 \\ 4. & \Gamma \vdash \lambda p.j : X \subseteq_f \mathcal{A} \to \langle X, \mathsf{false}\rangle \subseteq_f \mathcal{A} & (\to I), 3 \\ 5. & \Gamma \vdash \lambda p.j : \forall X\big(X \subseteq_f \mathcal{A} \to \langle X, \mathsf{false}\rangle \subseteq_f \mathcal{A}\big) & (\forall^2 I)), 4 \end{array}$$

$$\therefore\ s_2 = \lambda p.j$$

- $\vdash \mathsf{ite} : \forall b.\forall i.\forall j.B(b) \to \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{f}\ \ b)$

$$\begin{array}{lll} 1. & \Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathcal{A} \to \langle -, \mathsf{true}\rangle \subseteq_f \mathcal{A}\big) & (Hip) \\ 2. & \Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathcal{A} \to \langle -, \mathsf{false}\rangle \subseteq_f \mathcal{A}\big) & (Hip) \\ 3. & \Gamma \vdash b : B(b) & (Var) \\ 4. & \Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ b : \mathcal{A}(\mathsf{f}\ \ b) & (\mu E^-), 1, 2, 3 \\ 5. & b : B(b),\ i : \mathcal{A}(i) \vdash \lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b : \mathcal{A}(j) \to \mathcal{A}(\mathsf{f}\ \ b) & (\to I)), 4 \\ 6. & b : B(b) \vdash \lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b : \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{f}\ \ b) & (\to I)), 5 \\ 7. & \vdash \lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b : B(b) \to \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{f}\ \ b) & (\to I)), 6 \\ 8. & \vdash \lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b : \forall j.B(b) \to \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{f}\ \ b) & (\forall I)), 7 \\ 9. & \vdash \lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b : \forall i.\forall j.B(b) \to \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{f}\ \ b) & (\forall I)), 8 \\ 10. & \vdash \lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b : \forall b.\forall i.\forall j.B(b) \to \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{f}\ \ b) & (\forall I)), 9 \end{array}$$

$\therefore \vdash \overline{\mathsf{ite}} : \forall b.\forall i.\forall j.\mathbb{B}(b) \to \mathcal{A}(i) \to \mathcal{A}(j) \to \mathcal{A}(\mathsf{ite}\ \ b\ i\ j)$, donde $\overline{\mathsf{ite}} = \lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\, s_1\ s_2\ b$

$\square$

- **Semántica operacional de la función** $\overline{\mathsf{ite}}$
  - $\overline{\mathsf{ite}}\,\overline{\mathsf{true}}\ i\ j \quad = \quad i.$

$$
\begin{array}{lcll}
\overline{\mathsf{ite}}\,\overline{\mathsf{true}}\ i\ j & \to & (\lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b)\ \overline{\mathsf{true}}\ i\ j & \to \\
(\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b)\ [b := \overline{\mathsf{true}}]\ i\ j & \to & (\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{true}})\ i\ j & \to \\
(\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{true}})\ [i := i]\ j & \to & (\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{true}})\ j & \to \\
\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{true}}\ [j := j] & \to & \mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{true}} & \to \\
\mathsf{MIt}\ \ s_1\ s_2\ (\mathsf{in}_{2,1}) & \mapsto_\beta & s_1\ (\mathsf{MIt}\ \ s_1\ s_2\ ) & \to \\
(\lambda p.i)(\mathsf{MIt}\ \ s_1\ s_2\ ) & \to & i[p := (\mathsf{MIt}\ \ s_1\ s_2\ )] & \to \\
i & & &
\end{array}
$$

$$\therefore\ \overline{\mathsf{ite}}\,\overline{\mathsf{true}}\ i\ j \quad = \quad i$$

  - $\overline{\mathsf{ite}}\,\overline{\mathsf{false}}\ i\ j \quad = \quad j.$

$$
\begin{array}{lcll}
\overline{\mathsf{ite}}\,\overline{\mathsf{false}}\ i\ j & \to & (\lambda b.\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b)\ \overline{\mathsf{false}}\ i\ j & \to \\
(\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ b)\ [b := \overline{\mathsf{false}}]\ i\ j & \to & (\lambda i.\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{false}})\ i\ j & \to \\
(\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{false}})\ [i := i]\ j & \to & (\lambda j.\,\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{false}})\ j & \to \\
\mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{false}}\ [j := j] & \to & \mathsf{MIt}\ \ s_1\ s_2\ \overline{\mathsf{false}} & \to \\
\mathsf{MIt}\ \ s_1\ s_2\ (\mathsf{in}_{2,2}) & \mapsto_\beta & s_2\ (\mathsf{MIt}\ \ s_1\ s_2\ ) & \to \\
(\lambda p.j)(\mathsf{MIt}\ \ s_1\ s_2\ ) & \to & j[p := (\mathsf{MIt}\, s_1\ s_2\ )] & \to \\
j & & &
\end{array}
$$

$$\therefore\ \overline{\mathsf{ite}}\,\overline{\mathsf{false}}\ i\ j \quad = \quad j$$

### 4.1.2. Números naturales

- **Definición**. Se define el conjunto de los números naturales como:

$$\mathbb{N} =_{def} \mu(\Phi) \quad \text{con} \quad \Phi(X) = \{0\} \cup X^{suc}$$

- **Especificación funcional**

$$data\ Nat\ =\ Zero \mid Suc\ Nat$$

- **Implementación**. Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle -, \mathsf{zero}\rangle$, $\mathcal{C}_2 = \langle X, \mathsf{suc}\rangle$. Se definen los números naturales como:

$$\mathbb{N} =_{def} \mu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2).$$

- **Reglas de construcción**

$$\frac{}{\Gamma \vdash \mathsf{in}_{2,1} : \mathbb{N}(\mathsf{zero})}\ (\mu I)$$

$$\frac{\Gamma \vdash r_1 : \mathbb{N}(t_1)}{\Gamma \vdash \mathsf{in}_{2,2}\, r_1 : \mathbb{N}(\mathsf{suc}\ t_1)}\ (\mu I)$$

- **Constructores**

  - $\overline{\mathsf{zero}} = \mathsf{in}_{2,1}$

    *Demostración.*

$$1.\quad \Gamma \vdash \mathsf{in}_{2,1} : \mathbb{N}(\mathsf{zero}) \quad (\mu I)$$

$$\therefore\ \overline{\mathsf{zero}} = \mathsf{in}_{2,1}$$

□

  - $\overline{\mathsf{suc}} = \lambda x.\, \mathsf{in}_{2,2}\, x$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ x)$

$$\begin{array}{lll}
1. & x : \mathbb{N}(x) \vdash x : \mathbb{N}(n) & (Var) \\
2. & x : \mathbb{N}(x) \vdash \mathsf{in}_{2,2}\, x : \mathbb{N}(\mathsf{suc}\ x) & (\mu I), 1 \\
3. & \vdash \lambda x.\, \mathsf{in}_{2,2}\, x : \mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ x) & (\to I), 2 \\
4. & \vdash \lambda x.\, \mathsf{in}_{2,2}\, x : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ x) & (\forall I), 3
\end{array}$$

$$\therefore\ \overline{\mathsf{suc}} = \lambda x.\, \mathsf{in}_{2,2}\, x$$

□

- **Semántica operacional**

$$\begin{array}{rcl}
\mathsf{MIt}\ \vec{s}\,(\mathsf{zero}) & \mapsto_\beta & s_1 \\
\mathsf{MIt}\ \vec{s}\,(\mathsf{suc}\ n) & \mapsto_\beta & s_2\ \mathsf{MIt}\ n
\end{array}$$

Algunos ejemplos de extracción de programas son:

- **Suma**. Sea $\mathsf{sum}$ un símbolo de función y $\mathbb{E}_{\mathsf{sum}} = \{\mathsf{sum}\ n\ \mathsf{zero} = n,\ \mathsf{sum}\, n\ \mathsf{suc}\ m = \mathsf{suc}\ \mathsf{sum}\, n\ m\}$. Entonces

$$\vdash \overline{\mathsf{sum}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}\ n\ m)$$

donde $\overline{\mathsf{sum}} = \lambda x.\lambda y.\, \mathsf{MIt}\ s_1\ s_2\ y$, $s_1 = \lambda z.x$ y $s_2 = \lambda z.\lambda w.\overline{\mathsf{suc}}\ (zw)$ .

*Demostración.* Sea $\mathsf{f} =_{def} \mathsf{sum}\ n$, $\Gamma =_{def} \{x : \mathbb{N}(n),\ y : \mathbb{N}(m)\}$, $\Omega =_{def} \{z : X \subseteq_f \mathbb{N}\}$ y $\Delta =_{def} \Omega \cup \{w : X(x)\}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{sum}} = \{\mathsf{f}\ \mathsf{zero} = n,\ \mathsf{f}\ (\mathsf{suc}\ m) = \mathsf{suc}(\mathsf{f}\ m)\}$. Entonces hay que demostrar que

$$x : \mathbb{N}(n)\ \vdash \bar{\mathsf{f}} : \forall m.\mathbb{N}(m) \to \mathbb{N}(\mathsf{f}\ m)$$

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle -, \mathsf{zero}\rangle \subseteq_f \mathbb{N}\big)$

$$\begin{array}{lll}
1. & \Gamma,\ \Omega \vdash x : \mathbb{N}(n) & (Var)\\
2. & \Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{sum}}} x : \mathbb{N}(\mathsf{f}\ \mathsf{zero}) & (Eq), 1\\
3. & \Gamma,\ \Omega \vdash x : \langle -, \mathsf{zero}\rangle \subseteq_f \mathbb{N} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2\\
4. & \Gamma,\ \Omega \vdash z : X \subseteq_f \mathbb{N} & (Var)\\
5. & \Gamma \vdash \lambda z.x : X \subseteq_f \mathbb{N} \to \mathbb{N}(\mathsf{f}\ \mathsf{zero}) & (\to I), 4\\
6. & \Gamma \vdash \lambda z.x : \forall X\big(X \subseteq_f \mathbb{N} \to \mathbb{N}(\mathsf{f}\ \mathsf{zero})\big) & (\forall^2 I), 3
\end{array}$$

$$\therefore\ s_1 = \lambda z.x$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big)$

$$\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash z : X \subseteq_f \mathbb{N} & (Var)\\
2. & \Gamma,\ \Delta \vdash z : \forall x.X(x) \to \mathbb{N}(\mathsf{f}\ x) & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 1\\
3. & \Gamma,\ \Delta \vdash z : X(x) \to \mathbb{N}(\mathsf{f}\ x) & (\forall E), 2, [x := x]\\
4. & \Gamma,\ \Delta \vdash w : X(x) & (Var), 3\\
5. & \Gamma,\ \Delta \vdash zw : \mathbb{N}(\mathsf{f}\ x) & (\to E), 4\\
6. & \Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ x) & (Hip)\\
7. & \Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \mathbb{N}(\mathsf{f}\ x) \to \mathbb{N}(\mathsf{suc}\ (\mathsf{f}\ x)) & (\forall E), 6, [x := \mathsf{f}\ x]\\
8. & \Gamma,\ \Delta \vdash \overline{\mathsf{suc}}\ (zw) : \mathbb{N}(\mathsf{suc}\ (\mathsf{f}\ x)) & (\to E), 5, 7\\
9. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{sum}}} \overline{\mathsf{suc}}\ (zw) : \mathbb{N}(\mathsf{f}\ (\mathsf{suc}\ x)) & (Eq), 8\\
10. & \Gamma,\ \Omega\ \vdash \lambda w.\overline{\mathsf{suc}}\ (zw) : X(x) \to \mathbb{N}(\mathsf{f}\ (\mathsf{suc}\ x)) & (\to I), 9\\
11. & \Gamma,\ \Omega\ \vdash \lambda w.\overline{\mathsf{suc}}\ (zw) : \forall x.X(x) \to \mathbb{N}(\mathsf{f}\ (\mathsf{suc}\ x)) & (\forall I), 10\\
12. & \Gamma,\ \Omega\ \vdash \lambda w.\overline{\mathsf{suc}}\ (zw) : \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 11\\
13. & \Gamma\ \vdash \lambda z \lambda w.\overline{\mathsf{suc}}\ (zw) : X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N} & (\to I), 12\\
14. & \Gamma\ \vdash \lambda z \lambda w.\overline{\mathsf{suc}}\ (zw) : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big) & (\forall^2 I), 13
\end{array}$$

$$\therefore\ s_2 = \lambda z.\lambda w.\overline{\mathsf{suc}}\ (zw)$$

- $\vdash \mathsf{sum} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}\ m)$

$$\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle -, \mathsf{zero}\rangle \subseteq_f \mathbb{N}\big) & (Hip)\\
2. & \Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big) & (Hip)\\
3. & \Gamma \vdash y : \mathbb{N}(m) & (Var)\\
4. & \Gamma \vdash \mathsf{MIt}\ s_1\ s_2\ y : \mathbb{N}(\mathsf{f}(m)) & (\mu E^-), 1, 2, 3\\
5. & x : \mathbb{N}(n) \vdash \lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y : \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}\ m) & (\to I)), 4\\
6. & \vdash \lambda x.\lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y : \mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}\ m) & (\to I)), 5\\
7. & \vdash \lambda x.\lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y : \forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}\ m) & (\forall I)), 6\\
8. & \vdash \lambda x.\lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}\ m) & (\forall I)), 7
\end{array}$$

$\therefore \vdash \overline{\mathsf{sum}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}\ \ n\ m)$, donde $\overline{\mathsf{sum}} = \lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y$.

□

- **Semántica operacional de la función $\overline{\mathsf{sum}}$**
  - $\overline{\mathsf{sum}}\ n\ \overline{\mathsf{zero}} \quad = \quad n.$

$$\begin{array}{llll}
\overline{\mathsf{sum}}\ n\ \overline{\mathsf{zero}} & \to & (\lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ n\ \overline{\mathsf{zero}} & \to \\
(\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ [x := n]\ \overline{\mathsf{zero}} & \to & (\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ \overline{\mathsf{zero}} & \to \\
(\mathsf{MIt}\ \ s_1\ s_2\ y)\ [y := \overline{\mathsf{zero}}] & \to & \mathsf{MIt}\ \ s_1\ s_2\ \ \overline{\mathsf{zero}} & \to \\
\mathsf{MIt}\ \ s_1\ s_2\ \ \mathsf{in}_{2,1} & \mapsto_\beta & s_1\ (\mathsf{MIt}\ \ s_1\ s_2\ ) & \to \\
(\lambda z.n)(\mathsf{MIt}\ \ s_1\ s_2\ ) & \to & n[z := \mathsf{MIt}\ \ s_1\ s_2\ ] & \to \\
n & & &
\end{array}$$

$$\therefore\ \overline{\mathsf{sum}}\ n\ \overline{\mathsf{zero}} \quad = \quad n$$

  - $\overline{\mathsf{sum}}\ n\ (\overline{\mathsf{suc}}\ m) \quad = \quad \overline{\mathsf{suc}}\ (\overline{\mathsf{sum}}\ n\ m)$. La prueba se divide en dos partes: se demuestra primero que $\overline{\mathsf{sum}}\ n\ (\overline{\mathsf{suc}}\ m) \quad = \quad \overline{\mathsf{suc}}((\mathsf{MIt}\ \ s_1\ s_2)\ m)$ y después que $\overline{\mathsf{suc}}\ (\overline{\mathsf{sum}}\ n\ m) \quad = \overline{\mathsf{suc}}\ (\mathsf{MIt}\ \ s_1\ s_2\ m)$.

$$\begin{array}{llll}
\overline{\mathsf{sum}}\ n\ (\overline{\mathsf{suc}}\ m) & \to & (\lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ n\ (\overline{\mathsf{suc}}\ m) & \to \\
(\lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ n\ ((\lambda x.\,\mathsf{in}_{2,2}\,x)\ m) & \to & (\lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ n\ (\mathsf{in}_{2,2}\,x\ [x := m]) & \to \\
(\lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ n\ (\mathsf{in}_{2,2}\,m) & \to & (\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ [x := n]\ (\mathsf{in}_{2,2}\,m) & \to \\
(\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ (\mathsf{in}_{2,2}\,m) & \to & (\mathsf{MIt}\ \ s_1\ s_2\ y)\ [y := \mathsf{in}_{2,2}\,m] & \to \\
\mathsf{MIt}\ \ s_1\ s_2\ (\mathsf{in}_{2,2}\,m) & \mapsto_\beta & s_2\ (\mathsf{MIt}\ \ s_1\ s_2)\ m & \to \\
(\lambda z.\lambda w.\overline{\mathsf{suc}}(zw))\ (\mathsf{MIt}\ \ s_1\ s_2)\ m & \to & (\lambda w.\overline{\mathsf{suc}}(zw))\ [z := \mathsf{MIt}\ \ s_1\ s_2]\ m & \to \\
(\lambda w.\overline{\mathsf{suc}}((\mathsf{MIt}\ \ s_1\ s_2)\ w))m & \to & (\overline{\mathsf{suc}}((\mathsf{MIt}\ \ s_1\ s_2)\ w))[w := m] & \to \\
\overline{\mathsf{suc}}((\mathsf{MIt}\ \ s_1\ s_2)\ m) & & &
\end{array}$$

$$\begin{array}{llll}
\overline{\mathsf{suc}}\ (\overline{\mathsf{sum}}\ n\ m) & \to & \overline{\mathsf{suc}}\ ((\lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ n\ m) & \to \\
\overline{\mathsf{suc}}\ ((\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ [x := n]\ m) & \to & \overline{\mathsf{suc}}\ ((\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y)\ m) & \to \\
\overline{\mathsf{suc}}\ ((\mathsf{MIt}\ \ s_1\ s_2\ y)[y := m]) & \to & \overline{\mathsf{suc}}\ (\mathsf{MIt}\ \ s_1\ s_2\ m) &
\end{array}$$

$$\therefore\ \overline{\mathsf{sum}}\ n\ (\overline{\mathsf{suc}}\ m) \quad = \quad \overline{\mathsf{suc}}\ (\overline{\mathsf{sum}}\ n\ m)$$

- **Producto**. Sea $\mathsf{prod}$ un símbolo de función y $\mathbb{E}_{\mathsf{prod}} = \{\mathsf{prod}\ \ n\ \ \mathsf{zero} = \mathsf{zero},\ \mathsf{prod}\ \ n\ (\mathsf{suc}\,m) = \mathsf{sum}\ (\mathsf{prod}\ \ n\ m)\ n\}$. Entonces

$$\vdash \overline{\mathsf{prod}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{prod}\ \ n\ m)$$

donde $\overline{\mathsf{prod}} = \lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ y$, $s_1 = \lambda w.x$ y $s_2 = \lambda w.\lambda u.\overline{\mathsf{sum}}(wu)(x)$.

*Demostración.* Sea $\mathsf{f} =_{def} \mathsf{prod}\ \ n$, $\Gamma =_{def} \{x : \mathbb{N}(n),\ y : \mathbb{N}(m)\}$, $\Omega =_{def} \{w : X \subseteq_f \mathbb{N}\}$ y $\Delta =_{def} \Omega\ \cup\ \{u : X(x)\}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{prod}} = \{\mathsf{f}(\mathsf{zero}) = \mathsf{zero},\ \mathsf{f}(\mathsf{suc}\ \ m) = \mathsf{sum}((\mathsf{f}\ \ m)\ n)\}$. Entonces hay que demostrar que

$$x : \mathbb{N}(n)\ \vdash \overline{\mathsf{f}} : \forall m.\mathbb{N}(m) \to \mathbb{N}(\mathsf{f}\ \ m)$$

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle -, \mathsf{zero}\rangle \subseteq_f \mathbb{N}\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Omega \vdash x : \mathbb{N}(n)$ | $(Var)$ |
| 2. | $\Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{prod}}} x : \mathbb{N}(\mathsf{f\ zero})$ | $(Eq), 1$ |
| 3. | $\Gamma,\ \Omega \vdash w : X \subseteq_f \mathbb{N}$ | $(Var)$ |
| 4. | $\Gamma \vdash \lambda w.x : X \subseteq_f \mathbb{N} \to \mathbb{N}(\mathsf{f\ zero})$ | $(\to I), 3$ |
| 5. | $\Gamma \vdash \lambda w.x : \forall X\big(X \subseteq_f \mathbb{N} \to \mathbb{N}(\mathsf{f\ zero})\big)$ | $(\forall^2 I), 5$ |

$$\therefore\ s_1 = \lambda w.x$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash w : X \subseteq_f \mathbb{N}$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash w : \forall x.X(x) \to \mathbb{N}(\mathsf{f}\ x)$ | $(Def\ \mathcal{C} \subseteq \mathcal{K}), 1$ |
| 3. | $\Gamma,\ \Delta \vdash w : X(x) \to \mathbb{N}(\mathsf{f}\ x)$ | $(\forall E), 2, [x := x]$ |
| 4. | $\Gamma,\ \Delta \vdash u : X(x)$ | $(Var)$ |
| 5. | $\Gamma,\ \Delta \vdash wu : \mathbb{N}(\mathsf{f}\ x)$ | $(\to E), 3, 4$ |
| 6. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}\ n\ m)$ | $(Hip)$ |
| 7. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \forall m.\mathbb{N}(\mathsf{f}\ x) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}\ (\mathsf{f}\ x)\ m)$ | $(\forall E), [n := \mathsf{f}\ x], 6$ |
| 8. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \mathbb{N}(\mathsf{f}\ x) \to \mathbb{N}(n) \to \mathbb{N}(\mathsf{sum}\ (\mathsf{f}\ x)\ n)$ | $(\forall E), [m := n], 7$ |
| 9. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}}(wu) : \mathbb{N}(n) \to \mathbb{N}(\mathsf{sum}\ (\mathsf{f}\ x)\ n)$ | $(\to E), 5, 8$ |
| 10. | $\Gamma,\ \Delta \vdash x : \mathbb{N}(n)$ | $(Var)$ |
| 11. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}}(wu)(x) : \mathbb{N}(\mathsf{sum}\ (\mathsf{f}\ x)\ n)$ | $(\to E), 9, 10$ |
| 12. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{prod}}} \overline{\mathsf{sum}}(wu)(x) : \mathbb{N}(\mathsf{f}\ (\mathsf{suc}\ x))$ | $(Eq), 11$ |
| 13. | $\Gamma,\ \Omega \vdash \lambda u.\overline{\mathsf{sum}}(wu)(x) : X(x) \to \mathbb{N}(\mathsf{f}\ (\mathsf{suc}\ x))$ | $(\to I), 12$ |
| 14. | $\Gamma,\ \Omega \vdash \lambda u.\overline{\mathsf{sum}}(wu)(x) : \forall x.X(x) \to \mathbb{N}(\mathsf{f}\ (\mathsf{suc}\ x))$ | $(\forall I), 13$ |
| 15. | $\Gamma,\ \Omega \vdash \lambda u.\overline{\mathsf{sum}}(wu)(x) : \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 14$ |
| 16. | $\Gamma \vdash \lambda w.\lambda u.\overline{\mathsf{sum}}(wu)(x) : X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}$ | $(\to I), 15$ |
| 17. | $\Gamma \vdash \lambda w.\lambda u.\overline{\mathsf{sum}}(wu)(x) : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big)$ | $(\forall^2 I), 16$ |

$$\therefore\ s_2 = \lambda w.\lambda u.\overline{\mathsf{sum}}(wu)(x)$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle -, \mathsf{zero}\rangle \subseteq_f \mathbb{N}\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash y : \mathbb{N}(m)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MIt}\ s_1\ s_2\ y : \mathbb{N}(\mathsf{f}(m))$ | $(\mu E^-), 1, 2, 3$ |
| 5. | $x : \mathbb{N}(n) \vdash \lambda y.\,\mathsf{MIt}\ s_1\ s_2\ b : \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}(m))$ | $(\to I)), 4$ |
| 6. | $\vdash \lambda x.\lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y : \mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}(m))$ | $(\to I)), 5$ |
| 7. | $\vdash \lambda x.\lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y : \forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}(m))$ | $(\forall I)), 6$ |
| 8. | $\vdash \lambda x.\lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{f}(m))$ | $(\forall I)), 7$ |

$\therefore \vdash \overline{\mathsf{prod}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{prod}\ n\ m)$, donde $\overline{\mathsf{prod}} = \lambda x.\lambda y.\,\mathsf{MIt}\ s_1\ s_2\ y$.

□

- **Factorial**. Sea $\mathsf{fac}$ un símbolo de función y $\mathbb{E}_{\mathsf{fac}} = \{\mathsf{fac\ zero} = \mathsf{suc\ zero},\ \mathsf{fac\ suc}\ n = \mathsf{prod}\ (\mathsf{suc}\ n)\ (\mathsf{fac}\, n)\}$. Entonces

$$\vdash \overline{\mathsf{fac}} : \forall n.\mathbb{N}(n) \to \mathbb{N}(\mathsf{fac}\ n)$$

donde $\overline{\mathsf{fac}} = \lambda x.\,\mathsf{MRec}\ s_1\ s_2\ x$, $s_1 = \lambda z.\lambda w.\overline{\mathsf{suc}}(\overline{\mathsf{zero}})$ y $s_2 = \lambda z.\lambda w.\lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu)$.

*Demostración.* Sea $\mathsf{f} =_{def} \mathsf{fac}$, $\Gamma =_{def} \{x : \mathbb{N}(n)\}$, $\Omega =_{def} \{z : X \subseteq \mu(\Phi),\ w : X \subseteq_f \mathbb{N}\}$ y $\Delta =_{def} \Omega \cup \{u : X(x)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \langle -, \mathsf{zero}\rangle \subseteq_f \mathbb{N}\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Omega \vdash \overline{\mathsf{zero}} : \mathbb{N}(\mathsf{zero})$ | $(Hip)$ |
| 2. | $\Gamma,\ \Omega \vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\, x)$ | $(Hip)$ |
| 3. | $\Gamma,\ \Omega \vdash \overline{\mathsf{suc}} : \mathbb{N}(\mathsf{zero}) \to \mathbb{N}(\mathsf{suc}\ \mathsf{zero})$ | $(\forall E), 2, [x := zero]$ |
| 4. | $\Gamma,\ \Omega \vdash \overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : \mathbb{N}(\mathsf{suc}\ \mathsf{zero})$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{fac}}} \overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : \mathbb{N}(\mathsf{f}\ \mathsf{zero})$ | $(Eq), 4$ |
| 6. | $\Gamma,\ z : X \subseteq \mu(\Phi) \vdash \lambda w.\overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : X \subseteq_f \mathbb{N} \to \mathbb{N}(\mathsf{f}\ \mathsf{zero})$ | $(\to I), 5$ |
| 7. | $\Gamma \vdash \lambda z.\lambda w.\overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \mathbb{N}(\mathsf{f}\ \mathsf{zero})$ | $(\to I), 6$ |
| 8. | $\Gamma \vdash \lambda z.\lambda w.\overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : \forall X\big(X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \mathbb{N}(\mathsf{f}\ \mathsf{zero})\big)$ | $(\forall^2 I), 7$ |

$$\therefore\ s_1 = \lambda z.\lambda w.\overline{\mathsf{suc}}(\overline{\mathsf{zero}})$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big)$.

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash w : X \subseteq_f \mathbb{N}$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash w : \forall x.X(x) \to \mathbb{N}(\mathsf{f}\ x)$ | $(Def\ \mathcal{C} \subseteq \mathcal{K}), 1$ |
| 3. | $\Gamma,\ \Delta \vdash w : X(x) \to \mathbb{N}(\mathsf{f}\ x)$ | $(\forall E), 2, [x := x]$ |
| 4. | $\Gamma,\ \Delta \vdash u : X(x)$ | $(Var)$ |
| 5. | $\Gamma,\ \Delta \vdash wu : \mathbb{N}(\mathsf{f}\ x)$ | $(\to E), 3, 4$ |
| 6. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ x)$ | $(Hip)$ |
| 7. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ x)$ | $(\forall E), 6, [x := x]$ |
| 8. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}}\ x : \mathbb{N}(\mathsf{suc}\ x)$ | $(\to E), 7$ |
| 9. | $\Gamma,\ \Delta \vdash \overline{\mathsf{prod}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{prod}\ n\ m)$ | $(Hip)$ |
| 10. | $\Gamma,\ \Delta \vdash \overline{\mathsf{prod}} : \forall m.\mathbb{N}(\mathsf{suc}\ x) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{prod}\ (\mathsf{suc}\ x)\ m)$ | $(\forall E), 9, [n := \mathsf{suc}\ x]$ |
| 11. | $\Gamma,\ \Delta \vdash \overline{\mathsf{prod}} : \mathbb{N}(\mathsf{suc}\ x) \to \mathbb{N}(\mathsf{f}\ x) \to \mathbb{N}(\mathsf{prod}\ (\mathsf{suc}\ x)\ (\mathsf{f}\ x))$ | $(\forall E), 10, [m := \mathsf{f}\ x]$ |
| 12. | $\Gamma,\ \Delta \vdash \overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x) : \mathbb{N}(\mathsf{f}\ x) \to \mathbb{N}(\mathsf{prod}\ (\mathsf{suc}\ x)\ (\mathsf{f}\ x))$ | $(\to E), 8, 11$ |
| 13. | $\Gamma,\ \Delta \vdash \overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : \mathbb{N}(\mathsf{prod}\ (\mathsf{suc}\ x)\ (\mathsf{f}\ x))$ | $(\to E), 5, 12$ |
| 14. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{fac}}} \overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : \mathbb{N}(\mathsf{f}(\mathsf{suc}\ x))$ | $(Eq), 13$ |
| 15. | $\Gamma,\ \Omega \vdash \lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : X(x) \to \mathbb{N}(\mathsf{f}(\mathsf{suc}\ x))$ | $(\to I), 14$ |
| 16. | $\Gamma,\ \Omega \vdash \lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : \forall x.X(x) \to \mathbb{N}(\mathsf{f}(\mathsf{suc}\ x))$ | $(\forall I), 15$ |
| 17. | $\Gamma,\ \Omega \vdash \lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 16$ |
| 18. | $\Gamma,\ z : X \subseteq \mu(\Phi) \vdash \lambda w.\lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}$ | $(\to I), 17$ |
| 19. | $\Gamma \vdash \lambda z.\lambda w.\lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}$ | $(\to I), 18$ |
| 20. | $\Gamma \vdash \lambda z.\lambda w.\lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu) : \forall X\big(X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big)$ | $(\forall^2 I), 19$ |

$$\therefore\ s_2 = \lambda z.\lambda w.\lambda u.\overline{\mathsf{prod}}(\overline{\mathsf{suc}}\ x)(wu)$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \langle -, \mathsf{zero}\rangle \subseteq_f \mathbb{N}\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(X \subseteq \mu(\Phi) \to X \subseteq_f \mathbb{N} \to \langle X, \mathsf{suc}\rangle \subseteq_f \mathbb{N}\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash x : \mathbb{N}(n)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MRec}\ s_1\ s_2\ x : \mathbb{N}(\mathsf{f}(n))$ | $(\mu E), 1, 2, 3$ |
| 5. | $\vdash \lambda x.\,\mathsf{MRec}\ s_1\ s_2\ x : \mathbb{N}(n) \to \mathbb{N}(\mathsf{f}(n))$ | $(\to I)), 4$ |
| 6. | $\vdash \lambda x.\,\mathsf{MRec}\ s_1\ s_2\ x : \forall n.\mathbb{N}(n) \to \mathbb{N}(\mathsf{f}(n))$ | $(\forall I)), 5$ |

$\therefore \vdash \overline{\mathsf{fac}} : \forall n.\mathbb{N}(n) \to \mathbb{N}(\mathsf{fac}\ n)$, donde $\overline{\mathsf{fac}} = \lambda x.\,\mathsf{MRec}\ s_1\ s_2\ x$.

□

### 4.1.3. Listas finitas sobre A

- **Definición**. Se definen las listas sobre un conjunto A como:

$$\mathbb{L}_{\mathcal{A}} =_{def} \mu(\Phi) \quad \text{con} \quad \Phi(X) = \{nil\} \cup X^{cons}$$

- **Especificación funcional**

$$data\ List\ a\ =\ Nil \mid Cons\ a\ (List\ a)$$

- **Implementación**. Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle -, \mathsf{nil}\rangle$, $\mathcal{C}_2 = \langle \mathcal{A}, X, \mathsf{cons}\rangle$. Las listas finitas sobre A se definen como:

$$\mathbb{L}_{\mathcal{A}} =_{def} \mu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2).$$

- **Reglas de construcción**

$$\frac{}{\Gamma \vdash \mathsf{in}_{2,1} : \mathbb{L}_{\mathcal{A}}(\mathsf{nil})}\ (\mu I)$$

$$\frac{\Gamma \vdash r_1 : \mathcal{A}(t_1) \qquad \Gamma \vdash r_2 : \mathbb{L}_{\mathcal{A}}(t_2)}{\Gamma \vdash \mathsf{in}_{2,2}\, r_1 r_2 : \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ t_1\ t_2)}\ (\mu I)$$

- **Constructores**
  - $\overline{\mathsf{nil}} = \mathsf{in}_{2,1}$

    *Demostración.*

    1. $\Gamma \vdash \mathsf{in}_{2,1} : \mathbb{L}_{\mathcal{A}}(\mathsf{nil}) \quad (\mu I)$

$$\therefore\ \overline{\mathsf{nil}} = \mathsf{in}_{2,1}$$

□

- $\overline{\mathsf{cons}} = \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy$

  *Demostración.* Hay que probar que $\vdash \overline{\mathsf{cons}} : \forall x.\forall y.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$. Sea $\Gamma =_{def} \{x : \mathcal{A}(x), y : \mathbb{L}_{\mathcal{A}}(y)\}$

$$\begin{array}{lll}
1. & \Gamma \vdash x : \mathcal{A}(x) & (Var) \\
2. & \Gamma \vdash y : \mathbb{L}_{\mathcal{A}}(y) & (Var) \\
3. & \Gamma \vdash \mathsf{in}_{2,2}\, xy : \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y) & (\mu I), 1, 2 \\
4. & x : \mathcal{A}(x) \vdash \lambda y.\, \mathsf{in}_{2,2}\, xy : \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y) & (\to I), 3 \\
5. & \vdash \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy : \mathcal{A}(X) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y) & (\to I), 4 \\
6. & \vdash \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy : \forall y.\mathcal{A}(X) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y) & (\forall I), 5 \\
7. & \vdash \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy : \forall x.\forall y.\mathcal{A}(X) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y) & (\forall I), 6
\end{array}$$

$$\therefore\ \overline{\mathsf{cons}} = \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy$$

□

- **Semántica operacional**

$$\begin{array}{rcl}
\mathsf{MIt}\ \vec{s}\,(\mathsf{nil}) & \mapsto_\beta & s_1 \\
\mathsf{MIt}\ \vec{s}\,(\mathsf{cons}\ \ a\ l) & \mapsto_\beta & s_2\ \mathsf{MIt}\ \ a\ l
\end{array}$$

  Algunos ejemplos de extracción de programas son:

- **Append**. Sea $\mathsf{app}$ un símbolo de función y $\mathbb{E}_{\mathsf{app}} = \{\mathsf{app}\ \ \mathsf{nil}\ \ l_2 = l_2,\ \ \mathsf{app}\ \ (\mathsf{cons}\ \ a\ \ l_1)\ \ l_2 = \mathsf{cons}\ \ a\ (\mathsf{app}\ \ l_1\ l_2)\}$.

  Entonces

$$\vdash \overline{\mathsf{app}} : \forall l_1.\forall l_2.\mathbb{L}_{\mathcal{A}}(l_1) \to \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}\ \ l_1\ l_2)$$

  donde $\overline{\mathsf{app}} = \lambda x.\lambda y.\, \mathsf{MIt}\ \ s_1\ s_2\ x$, $s_1 = \lambda u.y$ y $s_2 = \lambda u.\lambda v.\lambda w.\overline{\mathsf{cons}}\ v\ (u\ w)$.

  *Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f}\ \ l_1 =_{def} \mathsf{app}\ \ l_1\ l_2$, $\Gamma =_{def} \{x : \mathbb{L}_{\mathcal{A}}(l_1),\ \ y : \mathbb{L}_{\mathcal{A}}(l_2)\}$, $\Omega =_{def} \{u : X \subseteq_f \mathbb{L}_{\mathcal{A}}\}$ y $\Delta =_{def} \Omega\ \cup\ \{v : \mathcal{A}(x),\ \ w : X(y)\}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{app}} = \{\mathsf{f}\ \ \mathsf{nil} = l_2,\ \mathsf{f}\ (\mathsf{cons}\ \ a\ l) = \mathsf{cons}\ \ a\ (\mathsf{f}\ \ l)\}$.

  - $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$

$$\begin{array}{lll}
1. & \Gamma, \Omega \vdash y : \mathbb{L}_{\mathcal{A}}(l_2) & (Var) \\
2. & \Gamma, \Omega \vdash_{\mathbb{E}_{\mathsf{app}}} y : \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ \mathsf{nil}) & (Eq), 1 \\
3. & \Gamma, \Omega \vdash y : \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2 \\
4. & \Gamma \vdash \lambda u.y : X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}} & (\to I), 3 \\
5. & \Gamma \vdash \lambda u.y : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big) & (\forall^2 I), 4
\end{array}$$

$$\therefore\ s_1 = \lambda u.y$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$.

$$
\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash w : X(y) & (Var)\\
2. & \Gamma,\ \Delta \vdash u : X \subseteq_f \mathbb{L}_{\mathcal{A}} & (Var)\\
3. & \Gamma,\ \Delta \vdash u : \forall x.X(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ x) & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2\\
4. & \Gamma,\ \Delta \vdash u : X(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ y) & (\forall E), 3, [x := y]\\
5. & \Gamma,\ \Delta \vdash u\ w : \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ y) & (\to E), 1, 4\\
6. & \Gamma,\ \Delta \vdash v : \mathcal{A}(x) & (Var)\\
7. & \Gamma,\ \Delta \vdash \overline{\mathsf{cons}} : \forall x.\forall y.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y) & (Hip)\\
8. & \Gamma,\ \Delta \vdash \overline{\mathsf{cons}} : \forall y.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y) & (\forall E), 7, [x := x]\\
9. & \Gamma,\ \Delta \vdash \overline{\mathsf{cons}} : \mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ (\mathsf{f}\ \ y)) & (\forall E), 8, [y := \mathsf{f}\ \ y]\\
10. & \Gamma,\ \Delta \vdash \overline{\mathsf{cons}}\, v : \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ (\mathsf{f}\ \ y)) & (\to E), 6, 9\\
11. & \Gamma,\ \Delta \vdash \overline{\mathsf{cons}}\, v\ (u\ w) : \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ (\mathsf{f}\ \ y)) & (\to E), 5, 10\\
12. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{app}}} \overline{\mathsf{cons}}\, v\ (u\ w) : \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}\ \ x\ \ y)) & (Eq), 11
\end{array}
$$

$$
\begin{array}{lll}
13. & \Gamma,\ \Omega,\ v : \mathcal{A}(x) \vdash \lambda w.\overline{\mathsf{cons}}\, v\ (u\ w) : X(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}\ \ x\ \ y)) & (\to I), 12\\
14. & \Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{cons}}\, v\ (u\ w) : \mathcal{A}(x) \to X(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}\ \ x\ \ y)) & (\to I), 13\\
15. & \Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{cons}}\, v\ (u\ w) : \forall y.\mathcal{A}(x) \to X(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}\ \ x\ \ y)) & (\forall I), 14\\
16. & \Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{cons}}\, v\ (u\ w) : \forall x.\forall y.\mathcal{A}(x) \to X(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}\ \ x\ \ y)) & (\forall I), 15\\
17. & \Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{cons}}\, v\ (u\ w) : \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 16\\
18. & \Gamma \vdash \lambda u.\lambda v.\lambda w.\overline{\mathsf{cons}}\, v\ (u\ w) : X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}} & (\to I), 17\\
19. & \Gamma \vdash \lambda u.\lambda v.\lambda w.\overline{\mathsf{cons}}\, v\ (u\ w) : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big) & (\forall^2 I), 18
\end{array}
$$

$$\therefore\ s_2 = \lambda u.\lambda v.\lambda w.\overline{\mathsf{cons}}\, v\ (u\ w)$$

-

$$
\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big) & (Hip)\\
2. & \Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big) & (Hip)\\
3. & \Gamma \vdash x : \mathbb{L}_{\mathcal{A}}(l_1) & (Var)\\
4. & \Gamma \vdash \mathsf{MIt}\ \ s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(\mathsf{f}(l_1)) & (\mu E^-), 1, 2, 3\\
5. & \Gamma \vdash \mathsf{MIt}\ \ s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(\mathsf{app}\ \ l_1\ l_2) & (Def\ \ \mathsf{f}), 4\\
6. & x : \mathbb{L}_{\mathcal{A}}(l_1) \vdash \lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}\ \ l_1\ l_2) & (\to I)), 5\\
7. & \vdash \lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(l_1) \to \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}\ \ l_1\ l_2) & (\to I)), 6\\
8. & \vdash \lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ x : \forall l_2.\mathbb{L}_{\mathcal{A}}(l_1) \to \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}\ \ l_1\ l_2) & (\forall I)), 7\\
9. & \vdash \lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ x : \forall l_1.\forall l_2.\mathbb{L}_{\mathcal{A}}(l_1) \to \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}\ \ l_1\ l_2) & (\forall I)), 8
\end{array}
$$

$\therefore \vdash \overline{\mathsf{app}} : \forall l_1.\forall l_2.\mathbb{L}_{\mathcal{A}}(l_1) \to \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}\ \ l_1\ l_2)$, donde $\overline{\mathsf{app}} = \lambda x.\lambda y.\,\mathsf{MIt}\ \ s_1\ s_2\ x$ $\square$

- **Semántica operacional de la función** $\overline{\mathsf{app}}$
  - $\overline{\mathsf{app}}\ \mathsf{nil}\ l_2 \ = \ l_2$.

$$
\begin{array}{llll}
\overline{\mathsf{app}}\ \overline{\mathsf{nil}}\ l_2 & \to & (\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ x)\ \overline{\mathsf{nil}}\ l_2 & \to\\
(\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ x)\ [x := \overline{\mathsf{nil}}]\ l_2 & \to & (\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nil}})\ l_2 & \to\\
(\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nil}})\ [y := l_2] & \to & (\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nil}}) & \to\\
(\mathsf{MIt}\, s_1\ s_2\ \ \mathsf{in}_{2,1}) & \mapsto_\beta & s_1\ (\mathsf{MIt}\, s_1\ s_2\ ) & \to\\
(\lambda u.l_2)\ (\mathsf{MIt}\, s_1\ s_2\ ) & \to & l_2\ [u = \mathsf{MIt}\, s_1\ s_2\ ] & \to\\
l_2 & & &
\end{array}
$$

$$\therefore\ \overline{\mathsf{app}}\ \mathsf{nil}\ l_2 \ = \ l_2$$

- $\overline{\mathsf{app}}\ \overline{\mathsf{cons}}\ (a\ l_1)\ l_2 \ = \ \overline{\mathsf{cons}}\ a\ (\overline{\mathsf{app}}\ l_1\ l_2)$. La prueba se divide en dos partes: se demuestra primero que $\overline{\mathsf{app}}\ \overline{\mathsf{cons}}\ (a\ l_1)\ l_2 \ = \ \overline{\mathsf{cons}}\ (a)\ ((\mathsf{MIt}\, s_1\ s_2)\ l_1)$ y después que $\overline{\mathsf{cons}}\ a\ (\overline{\mathsf{app}}\ l_1\ l_2) \ = \ \overline{\mathsf{cons}}\ (a)\ ((\mathsf{MIt}\, s_1\ s_2)\ l_1)$.

$$\begin{array}{lll}
\overline{\mathsf{app}}\ \overline{\mathsf{cons}}\ (a\ l_1)\ l_2 & \to & (\lambda x.\lambda y.\, \mathsf{MIt}\, s_1\ s_2\ x)\ \overline{\mathsf{cons}}\ (a\ l_1)\ l_2 \quad \to \\
(\lambda y.\, \mathsf{MIt}\, s_1\ s_2\ x)\ [x := \overline{\mathsf{cons}}\ (a\ l_1)]\ l_2 & \to & (\lambda y.\, \mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{cons}}\ (a\ l_1))\ l_2 \quad \to \\
(\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{cons}}\ (a\ l_1))\ [y := l_2] & \to & \mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{cons}}\ a\ l_1 \quad \to \\
\mathsf{MIt}\, s_1\ s_2\ (\lambda x.\lambda y.\, \mathsf{in}_{2,2}\ \ x\ y)\ a\ l_1 & \to & \mathsf{MIt}\, s_1\ s_2\ (\lambda y.\, \mathsf{in}_{2,2}\ \ x\ y)\ [x := a]\ l_1 \quad \to \\
\mathsf{MIt}\, s_1\ s_2\ (\lambda y.\, \mathsf{in}_{2,2}\ \ a\ y)\ l_1 & \to & \mathsf{MIt}\, s_1\ s_2\ (\mathsf{in}_{2,2}\ \ a\ y)[y := l_1] \quad \to \\
\mathsf{MIt}\, s_1\ s_2\ (\mathsf{in}_{2,2}\ \ a\ l_1) & \mapsto_\beta & s_2\ (\mathsf{MIt}\, s_1\ s_2)\ a\ l_1 \quad \to \\
(\lambda u.\lambda v.\lambda w.\overline{\mathsf{cons}}\ v\ (u\ w))(\mathsf{MIt}\, s_1\ s_2)\ a\ l_1 & \to & (\lambda v.\lambda w.\overline{\mathsf{cons}}\ v\ (u\ w))[u := \mathsf{MIt}\, s_1\ s_2]\ a\ l_1 \quad \to \\
(\lambda v.\lambda w.\overline{\mathsf{cons}}\ v\ ((\mathsf{MIt}\, s_1\ s_2)\ w))\ a\ l_1 & \to & (\lambda w.\overline{\mathsf{cons}}\ v\ ((\mathsf{MIt}\, s_1\ s_2)\ w))[v := a]\ l_1 \quad \to \\
(\lambda w.\overline{\mathsf{cons}}\ a\ ((\mathsf{MIt}\, s_1\ s_2)\ w))\ l_1 & \to & \overline{\mathsf{cons}}\ a\ ((\mathsf{MIt}\, s_1\ s_2)\ w)\ [w := l_1] \quad \to \\
\overline{\mathsf{cons}}\ a\ ((\mathsf{MIt}\, s_1\ s_2)\ l_1) & &
\end{array}$$

$$\begin{array}{lll}
\overline{\mathsf{cons}}\ a\ (\overline{\mathsf{app}}\ l_1\ l_2) & \to & \overline{\mathsf{cons}}\ a\ ((\lambda x.\lambda y.\, \mathsf{MIt}\, s_1\ s_2\ x)\ l_1\ l_2) \quad \to \\
\overline{\mathsf{cons}}\ a\ ((\lambda y.\, \mathsf{MIt}\, s_1\ s_2\ x)\ [x := l_1]\ l_2) & \to & \overline{\mathsf{cons}}\ a\ ((\lambda y.\, \mathsf{MIt}\, s_1\ s_2\ l_1)\ l_2) \quad \to \\
\overline{\mathsf{cons}}\ a\ ((\mathsf{MIt}\, s_1\ s_2\ l_1)\ [y :=; l_2]) & \to & \overline{\mathsf{cons}}\ a\ (\mathsf{MIt}\, s_1\ s_2\ l_1)
\end{array}$$

$$\therefore\ \overline{\mathsf{app}}\ \overline{\mathsf{cons}}\ (a\ l_1)\ l_2 \ = \ \overline{\mathsf{cons}}\ a\ (\overline{\mathsf{app}}\ l_1\ l_2)$$

- **Wrap**. Sea $\mathsf{wrap}$ un símbolo de función tal que $\mathbb{E}_{\mathsf{wrap}} = \{\mathsf{wrap}(x) = \mathsf{cons}(x\ \mathsf{nil})\}$. Entonces

$$\vdash \overline{\mathsf{wrap}} : \forall x.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{wrap}\ \ x)$$

donde $\overline{\mathsf{wrap}} = \lambda x.\overline{\mathsf{cons}}(x)(\overline{\mathsf{nil}})$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f} = \mathsf{wrap}$.

| | | |
|---|---|---|
| 1. | $x : \mathcal{A}(x) \vdash x : \mathcal{A}(x)$ | $(Var)$ |
| 2. | $x : \mathcal{A}(x) \vdash \overline{\mathsf{cons}} : \forall x.\forall y.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$ | $(Hip)$ |
| 3. | $x : \mathcal{A}(x) \vdash \overline{\mathsf{cons}} : \forall y.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(y) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$ | $(\forall E), 2, [x := x]$ |
| 4. | $x : \mathcal{A}(x) \vdash \overline{\mathsf{cons}} : \mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(nil) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ nil)$ | $(\forall E), 3, [y := nil]$ |
| 5. | $x : \mathcal{A}(x) \vdash \overline{\mathsf{nil}} : \mathbb{L}_{\mathcal{A}}(nil)$ | $(Hip)$ |
| 6. | $x : \mathcal{A}(x) \vdash \overline{\mathsf{cons}}(x) : \mathbb{L}_{\mathcal{A}}(nil) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ nil)$ | $(\to E), 1, 4$ |
| 7. | $x : \mathcal{A}(x) \vdash \overline{\mathsf{cons}}(x)(\overline{\mathsf{nil}}) : \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ nil)$ | $(\to E), 5, 6$ |
| 8. | $\vdash \lambda x.\overline{\mathsf{cons}}(x)(\overline{\mathsf{nil}}) : \mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{cons}\ \ x\ nil)$ | $(\to I), 7$ |
| 9. | $\vdash \lambda x.\overline{\mathsf{cons}}(x)(\overline{\mathsf{nil}}) : \mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ x)$ | $(Eq), 8$ |
| 10. | $\vdash \lambda x.\overline{\mathsf{cons}}(x)(\overline{\mathsf{nil}}) : \forall x.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ x)$ | $(\forall I), 9$ |

$$\therefore\ \overline{\mathsf{wrap}} = \lambda x.\overline{\mathsf{cons}}(x)(\overline{\mathsf{nil}})$$

□

- **Reverse**. Sea $\mathsf{rev}$ un símbolo de función y $\mathbb{E}_{\mathsf{rev}} = \{\mathsf{rev}(\mathsf{nil}) = \mathsf{nil},\ \mathsf{rev}(\mathsf{cons}(a\ l)) = \mathsf{app}(\mathsf{rev}(l)\ \ \mathsf{wrap}(a))\}$. Entonces

$$\vdash \overline{\mathsf{rev}} : \forall l.\mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{rev}\ \ l)$$

donde $\overline{\mathsf{rev}} = \lambda x.\,\mathsf{MIt}\ \ s_1\ s_2\ x,\ s_1 = \lambda u.\overline{\mathsf{nil}}$ y $s_2 = \lambda u.\lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v)$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f} = \mathsf{rev}$, $\Gamma := \{x : \mathbb{L}_{\mathcal{A}}(l)\}$, $\Omega := \{u : X \subseteq_f \mathbb{L}_{\mathcal{A}}\}$ y $\Delta := \Omega \cup \{v : \mathcal{A}(a),\ w : X(l)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Omega \vdash \overline{\mathsf{nil}} : \mathbb{L}_{\mathcal{A}}(nil)$ | $(Hip)$ |
| 2. | $\Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{rev}}} \overline{\mathsf{nil}} : \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ \mathsf{nil})$ | $(Eq), 1$ |
| 3. | $\Gamma,\ \Omega \vdash \overline{\mathsf{nil}} : \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2$ |
| 4. | $\Gamma \vdash \lambda u.\overline{\mathsf{nil}} : X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}$ | $(\to I), 3$ |
| 5. | $\Gamma \vdash \lambda u.\overline{\mathsf{nil}} : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$ | $(\forall^2 I), 4$ |

$$\therefore\ s_1 = \lambda u.\overline{\mathsf{nil}}$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$.

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash v : \mathcal{A}(a)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash \overline{\mathsf{wrap}} : \forall x.\mathcal{A}(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{wrap}\ \ x)$ | $(Hip)$ |
| 3. | $\Gamma,\ \Delta \vdash \overline{\mathsf{wrap}} : \mathcal{A}(a) \to \mathbb{L}_{\mathcal{A}}(\mathsf{wrap}\ \ a)$ | $(\forall E), 2, [x := a]$ |
| 4. | $\Gamma,\ \Delta \vdash \overline{\mathsf{wrap}}v : \mathbb{L}_{\mathcal{A}}(\mathsf{wrap}\ \ a)$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\ \Delta \vdash w : X(l)$ | $(Var)$ |
| 6. | $\Gamma,\ \Delta \vdash u : X \subseteq_f \mathbb{L}_{\mathcal{A}}$ | $(Var)$ |
| 7. | $\Gamma,\ \Delta \vdash u : \forall x.X(x) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ x)$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 6$ |
| 8. | $\Gamma,\ \Delta \vdash u : X(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ l)$ | $(\forall E), 7, [x := l]$ |
| 9. | $\Gamma,\ \Delta \vdash uw : \mathbb{L}_{\mathcal{A}}(\mathsf{f}\ \ l)$ | $(\to E), 5, 8$ |
| 10. | $\Gamma,\ \Delta \vdash \overline{\mathsf{app}} : \forall l_1.\forall l_2.\mathbb{L}_{\mathcal{A}}(l_1) \to \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}(l_1\ l_2))$ | $(Hip)$ |
| 11. | $\Gamma,\ \Delta \vdash \overline{\mathsf{app}} : \forall l_2.\mathbb{L}_{\mathcal{A}}(f\ l) \to \mathbb{L}_{\mathcal{A}}(l_2) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}(f\ l\ \ l_2))$ | $(\forall E), 10, [l_1 := f\ l]$ |
| 12. | $\Gamma,\ \Delta \vdash \overline{\mathsf{app}} : \mathbb{L}_{\mathcal{A}}(f\ l) \to \mathbb{L}_{\mathcal{A}}(wrap\ a) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}(f\ l\ \ wrap\ a))$ | $(\forall E), 11, [l_2 := wrap\ a]$ |
| 13. | $\Gamma,\ \Delta \vdash \overline{\mathsf{app}}(uw) : \mathbb{L}_{\mathcal{A}}(wrap\ a) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}(f\ l\ \ wrap\ a))$ | $(\to E), 9, 12$ |
| 14. | $\Gamma,\ \Delta \vdash \overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : \mathbb{L}_{\mathcal{A}}(\mathsf{app}(f\ l\ \ wrap\ a))$ | $(\to E), 4, 13$ |
| 15. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{rev}}} \overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(Eq), 14$ |
| 16. | $\Gamma,\ \Omega,\ v : \mathcal{A}(a) \vdash \lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : X(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\to I), 15$ |
| 17. | $\Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : \mathcal{A}(a) \to X(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\to I), 16$ |
| 18. | $\Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : \forall l.\mathcal{A}(a) \to X(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\forall I), 17$ |
| 19. | $\Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : \forall a.\forall l.\mathcal{A}(a) \to X(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\forall I), 18$ |
| 20. | $\Gamma,\ \Omega \vdash \lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 19$ |

| | | |
|---|---|---|
| 21. | $\Gamma \vdash \lambda u.\lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}$ | $(\to I), 20$ |
| 22. | $\Gamma \vdash \lambda u.\lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v) : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$ | $(\forall^2 I), 21$ |

$$\therefore\ s_2 = \lambda u.\lambda v.\lambda w.\overline{\mathsf{app}}(uw)(\overline{\mathsf{wrap}}v)$$

- | | | |
  |---|---|---|
  | 1. | $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$ | $(Hip)$ |
  | 2. | $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{A}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{A}}\big)$ | $(Hip)$ |
  | 3. | $\Gamma \vdash x : \mathbb{L}_{\mathcal{A}}(l)$ | $(Var)$ |
  | 4. | $\Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(\mathsf{f}(l))$ | $(\mu E^-), 1, 2, 3$ |
  | 5. | $\Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(\mathsf{rev}\ \ l)$ | $(Def\ \ \mathsf{f}), 4$ |
  | 6. | $\vdash \lambda x.\,\mathsf{MIt}\, s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{rev}\ \ l)$ | $(\to I)), 5$ |
  | 7. | $\vdash \lambda x.\,\mathsf{MIt}\, s_1\ s_2\ x : \forall l.\mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{rev}\ \ l)$ | $(\forall I), 7$ |

$\therefore \vdash \overline{\mathsf{rev}} : \forall l.\mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{A}}(\mathsf{app}(l))$, donde $\overline{\mathsf{rev}} = \lambda x.\,\mathsf{MIt}\, s_1\ s_2\ x$ □

- **Map**. Sean $\mathsf{map}$, $\mathsf{g}$ símbolos de función tales $\mathbb{E}_{\mathsf{map}} = \{\mathsf{map}(\mathsf{g}\ \ \mathsf{nil}) = \mathsf{nil},\ \mathsf{map}(\mathsf{g}\ \ \mathsf{cons}(a\ l)) = \mathsf{cons}(\mathsf{g}\ \ a\ \ \ \mathsf{map}(\mathsf{g}\ \ \ l))\}$ y $\mathsf{g} : \mathcal{A}(a) \to \mathcal{B}(\mathsf{g}(a))$. Entonces

$$\vdash \overline{\mathsf{map}} : \forall l.\big(\forall a.\mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a)\big) \to \mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{map}\ \ l)$$

donde $\overline{\mathsf{map}} = \lambda p.\lambda x.\,\mathsf{MIt}\ \ s_1\ s_2\ x$, $s_1 = \lambda q.\overline{\mathsf{nil}}$ y $s_2 = \lambda q.\lambda v.\lambda w.\overline{\mathsf{cons}}(pr)(qs)$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f} = \mathsf{map}\, g$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{map}} = \{\mathsf{f}(\mathsf{nil}) = \mathsf{nil},\ \mathsf{f}(\mathsf{cons}(a\ l)) = \mathsf{cons}(\mathsf{g}\ \ a\ \ \ \mathsf{f}\ \ l))\}$. Sea $\Gamma := \{p : \forall a.\mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a),\ x : \mathbb{L}_{\mathcal{A}}(l)\}$, $\Omega := \{q : X \subseteq_f \mathbb{L}_{\mathcal{B}}\}$ y $\Delta := \Omega \cup \{r : \mathcal{A}(a),\ s : X(l)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}\big)$

  | | | |
  |---|---|---|
  | 1. | $\Gamma,\ \Omega \vdash \overline{\mathsf{nil}} : \mathbb{L}_{\mathcal{B}}(\mathsf{nil})$ | $(Hip)$ |
  | 2. | $\Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{map}}} \overline{\mathsf{nil}} : \mathbb{L}_{\mathcal{B}}(\mathsf{f}\ \ \mathsf{nil})$ | $(Eq), 1$ |
  | 3. | $\Gamma,\ \Omega \vdash \overline{\mathsf{nil}} : \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2$ |
  | 4. | $\Gamma \vdash \lambda q.\overline{\mathsf{nil}} : X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}$ | $(\to I), 3$ |
  | 5. | $\Gamma \vdash \lambda q.\overline{\mathsf{nil}} : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}\big)$ | $(\forall^2 I), 4$ |

$$\therefore\ s_1 = \lambda q.\overline{\mathsf{nil}}$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}\big)$.

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash r : \mathcal{A}(a)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash p : \forall a.\mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a)$ | $(Var)$ |
| 3. | $\Gamma,\ \Delta \vdash p : \mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a)$ | $(\forall E), 2, [a := a]$ |
| 4. | $\Gamma,\ \Delta \vdash pr : \mathcal{B}(\mathsf{g}\ \ a)$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\ \Delta \vdash s : X(l)$ | $(\to E), 1, 3$ |
| 6. | $\Gamma,\ \Delta \vdash q : X \subseteq_f \mathbb{L}_{\mathcal{B}}$ | $(Var)$ |
| 7. | $\Gamma,\ \Delta \vdash q : \forall x.X(x) \to \mathbb{L}_{\mathcal{B}}(\mathsf{f}\ \ x)$ | $(Def\ \mathcal{C} \subseteq_f\ \ K), 6$ |
| 8. | $\Gamma,\ \Delta \vdash q : X(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{f}\ \ l)$ | $(\forall E), 7, [x := l]$ |
| 9. | $\Gamma,\ \Delta \vdash qs : \mathbb{L}_{\mathcal{B}}(\mathsf{f}\ \ l)$ | $(\to E), 5, 8$ |

| | | |
|---|---|---|
| 10. | $\Gamma,\ \Delta \vdash \overline{\mathsf{cons}} : \forall x.\forall y.\mathcal{B}(x) \to \mathbb{L}_{\mathcal{B}}(y) \to \mathbb{L}_{\mathcal{B}}(\mathsf{cons}\ \ x\ y)$ | $(Hip)$ |
| 11. | $\Gamma,\ \Delta \vdash \overline{\mathsf{cons}} : \forall y.\mathcal{B}(\mathsf{g}\ \ a) \to \mathbb{L}_{\mathcal{B}}(y) \to \mathbb{L}_{\mathcal{B}}(\mathsf{cons}\ \ \mathsf{g}\ \ a\ \ y)$ | $(\forall E), 10, [x := \mathsf{g}\ \ a]$ |
| 12. | $\Gamma,\ \Delta \vdash \overline{\mathsf{cons}} : \mathcal{B}(\mathsf{g}\ \ a) \to \mathbb{L}_{\mathcal{B}}(\mathsf{f}\ \ l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{cons}\ \ \mathsf{g}\ \ a\ \ \mathsf{f}\ \ l))$ | $(\forall E), 11, [y := \mathsf{f}\ \ l]$ |
| 13. | $\Gamma,\ \Delta \vdash \overline{\mathsf{cons}}(pr) : \mathbb{L}_{\mathcal{B}}(\mathsf{f}\ \ l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{cons}\ \ \mathsf{g}\ \ a\ \ \mathsf{f}\ \ l)$ | $(\to E), 4, 12$ |
| 14. | $\Gamma,\ \Delta \vdash \overline{\mathsf{cons}}(pr)(qs) : \mathbb{L}_{\mathcal{B}}(\mathsf{cons}\ \ \mathsf{g}\ \ a\ \ \mathsf{f}\ \ l)$ | $(\to E), 9, 13$ |
| 15. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{map}}} \overline{\mathsf{cons}}(pr)(qs) : \mathbb{L}_{\mathcal{B}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(Eq), 14$ |

| | | |
|---|---|---|
| 16. | $\Gamma,\ \Omega,\ r : \mathcal{A}(a) \vdash \lambda s.\overline{\mathsf{cons}}(pr)(qs) : X(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\to I), 15$ |
| 17. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{cons}}(pr)(qs) : A(a) \to X(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\to I), 16$ |
| 18. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{cons}}(pr)(qs) : \forall l.A(a) \to X(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\forall I), 17$ |
| 19. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{cons}}(pr)(qs) : \forall a.\forall l.A(a) \to X(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{f}(\mathsf{cons}(a\ l)))$ | $(\forall I), 18$ |

| | | |
|---|---|---|
| 20. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{cons}}(pr)(qs) : \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}$ | $(Def\ \mathcal{C} \subseteq_f\ \ K), 19$ |
| 21. | $\Gamma \vdash \lambda q.\lambda r.\lambda s.\overline{\mathsf{cons}}(pr)(qs) : X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}$ | $(\to I), 20$ |
| 22. | $\Gamma \vdash \lambda q.\lambda r.\lambda s.\overline{\mathsf{cons}}(pr)(qs) : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}\big)$ | $(\forall^2 I), 21$ |

$$\therefore\ s_2 = \lambda q.\lambda v.\lambda w.\overline{\mathsf{cons}}(pr)(qs)$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle -, \mathsf{nil}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{L}_{\mathcal{B}} \to \langle \mathcal{A}, X, \mathsf{cons}\rangle \subseteq_f \mathbb{L}_{\mathcal{B}}\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash x : \mathbb{L}_{\mathcal{A}}(l)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{L}_{\mathcal{B}}(\mathsf{f}(l))$ | $(\mu E^-), 1, 2, 3$ |
| 5. | $\Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{L}_{\mathcal{B}}(\mathsf{map}\ \ l)$ | $(Def\ \ \mathsf{f}), 4$ |
| 6. | $p : \forall a.\mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a) \vdash \lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{map}\ \ l)$ | $(\to I), 5$ |
| 7. | $\vdash \lambda p.\lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x : \big(\forall a.\mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a)\big) \to \mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{map}\ \ l)$ | $(\to I), 6$ |
| 8. | $\vdash \lambda p.\lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x : \forall l.\big(\forall a.\mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a)\big) \to \mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{map}\ \ l)$ | $(\forall I), 7$ |

$$\therefore \vdash \overline{\mathsf{map}} : \forall l.\big(\forall a.\mathcal{A}(a) \to \mathcal{B}(\mathsf{g}\ \ a)\big) \to \mathbb{L}_{\mathcal{A}}(l) \to \mathbb{L}_{\mathcal{B}}(\mathsf{map}\ \ l),\ \text{ donde }\ \ \overline{\mathsf{map}} = \lambda p.\lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x$$

□

### 4.1.4. Maybe

- **Definición**. Sea $P^{just} = \{just\ x | x \in P\}$. El tipo Maybe que encapsula a otro tipo opcional P se define como:

$$\mathsf{Mb} =_{def} \mu(\Phi) \quad \text{con} \quad \Phi(X) = \{nothing\} \cup P^{just}$$

- **Especificación funcional**

$$data\ Maybe\ P =\ Nothing |\ Just\ P$$

- **Implementación**. Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle -, \mathsf{nothing}\rangle, \mathcal{C}_2 = \langle \mathcal{P}, \mathsf{just}\rangle$. Se define el predicado inductivo Maybe como:

$$\mathsf{Mb}\ \mathcal{P} =_{def} \mu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2).$$

- **Reglas de construcción**

$$\frac{}{\Gamma \vdash \mathsf{in}_{2,1} : \mathsf{Mb}\ \mathcal{P}(\mathsf{nothing})}\ (\mu I)$$

$$\frac{\Gamma \vdash r : \mathcal{P}(x)}{\Gamma \vdash \mathsf{in}_{2,2}\, r : \mathsf{Mb}\ \mathcal{P}(\mathsf{just}\ \ x)}\ (\mu I)$$

- **Constructores**
  - $\overline{\mathsf{nothing}} = \mathsf{in}_{2,1}$

    *Demostración.*

    $$1. \quad \Gamma \vdash \mathsf{in}_{2,1} : \mathsf{Mb}\ \mathcal{P}(\mathsf{nothing}) \quad (Var)$$

    $$\therefore\ \overline{\mathsf{nothing}} = \mathsf{in}_{2,1}$$

    □

  - $\overline{\mathsf{just}} = \lambda x.\, \mathsf{in}_{2,2}\, x$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{just}} : \forall x.\mathcal{P}(x) \to \mathsf{Mb}\ \mathcal{P}(\mathsf{just}\ \ x)$

    $$\begin{array}{lll} 1. & x : \mathcal{P}(x) \vdash x : \mathcal{P}(x) & (Var) \\ 2. & x : \mathcal{P}(x) \vdash \mathsf{in}_{2,2}\, x : \mathsf{Mb}\ \mathcal{P}(\mathsf{just}\ \ x) & (\mu I), 1 \\ 3. & \vdash \lambda x.\, \mathsf{in}_{2,2}\, x : \mathcal{P}(x) \to \mathsf{Mb}\ \mathcal{P}(\mathsf{just}\ \ x) & (\to I), 2 \\ 4. & \vdash \lambda x.\, \mathsf{in}_{2,2}\, x : \forall x.\mathcal{P}(x) \to \mathsf{Mb}\ \mathcal{P}(\mathsf{just}\ \ x) & (\forall I), 3 \end{array}$$

    $$\therefore\ \overline{\mathsf{just}} = \lambda x.\, \mathsf{in}_{2,2}\, x$$

    □

- **Semántica operacional**

$$\begin{array}{rcl} \mathsf{MIt}\ \vec{s}\,(\mathsf{nothing}) & \mapsto_\beta & s_1 \\ \mathsf{MIt}\ \vec{s}\,(\mathsf{just}\ x) & \mapsto_\beta & s_2\ \mathsf{MIt}\ x \end{array}$$

Un ejemplo de extracción de programa es:

- **Condicional**. Sean $\mathsf{iMb}$ un símbolo de función y $\mathbb{E}_{\mathsf{iMb}} = \{\mathsf{iMb}\ \mathsf{nothing}\ x\ y\ = x,\ \mathsf{iMb}\ (\mathsf{just}\ z)\ x\ y = y\}$. La función que determina si el elemento de un producto es $\mathsf{nothing}$ o $\mathsf{just}$ se define como

$$\vdash \overline{\mathsf{iMb}} : \forall r.\forall x.\forall y.\mathsf{Mb}\ \mathcal{A}(r) \to \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}(r\ x\ y))$$

donde $\overline{\mathsf{iMb}} = \lambda r.\lambda x.\lambda y.\, \mathsf{MIt}\ s_1\ s_2\ r$, $s_1 = \lambda p.x$ y $s_2 = \lambda p.\lambda z.y$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f}\ r = \mathsf{iMb}\ r\ x\ y$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{iMb}} = \{\mathsf{f}\ \mathsf{nothing}\ x\ y = x,\ \mathsf{f}\ (\mathsf{just}\ z)\ x\ y = y\}$. Sea $\Gamma := \{r : \mathsf{Mb}\ \mathcal{A}(r), x : \mathbb{B}(x), y : \mathbb{B}(y)\}$, $\Omega := \{p : X \subseteq_f \mathbb{B}\}$ y $\Delta := \Omega \cup \{z : X(z)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{B} \to \langle -, \mathsf{nothing}\rangle \subseteq_f \mathbb{B}\big)$

$$\begin{array}{lll} 1. & \Gamma,\ \Omega \vdash x : \mathbb{B}(x) & (Var) \\ 2. & \Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{iMb}}} x : \mathbb{B}(\mathsf{f}(\mathsf{nothing})) & (Eq), 1 \\ 3. & \Gamma,\ \Omega \vdash x : \langle -, \mathsf{nothing}\rangle \subseteq_f \mathbb{B} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2 \\ 4. & \Gamma \vdash \lambda p.x : X \subseteq_f \mathbb{B} \to \langle -, \mathsf{nothing}\rangle \subseteq_f \mathbb{B} & (\to I), 3 \\ 5. & \Gamma \vdash \lambda p.x : \forall X\big(X \subseteq_f \mathbb{B} \to \langle -, \mathsf{nothing}\rangle \subseteq_f \mathbb{B}\big) & (\forall^2 I), 4 \end{array}$$

$$\therefore\ s_1 = \lambda p.x$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{B} \to \langle X, \mathsf{just}\rangle \subseteq_f \mathbb{B}\big)$

$$\begin{array}{lll} 1. & \Gamma,\ \Delta \vdash y : \mathbb{B}(y) & (Var) \\ 2. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{iMb}}} y : \mathbb{B}(\mathsf{f}(\mathsf{just}\ z)) & (Eq), 1 \\ 3. & \Gamma,\ \Omega \vdash \lambda z.y : X(z) \to \mathbb{B}(\mathsf{f}(\mathsf{just}\ z)) & (\to I), 2 \\ 4. & \Gamma,\ \Omega \vdash \lambda z.y : \forall z.X(z) \to \mathbb{B}(\mathsf{f}(\mathsf{just}\ z)) & (\forall I), 3 \\ 5. & \Gamma,\ \Omega \vdash \lambda z.y : \langle X, \mathsf{just}\rangle \subseteq_f \mathbb{B} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 4 \\ 6. & \Gamma \vdash \lambda p.\lambda z.y : X \subseteq_f \mathbb{B} \to \langle X, \mathsf{just}\rangle \subseteq_f \mathbb{B} & (\to I), 5 \\ 7. & \Gamma \vdash \lambda p.\lambda z.y : \forall X\big(X \subseteq_f \mathbb{B} \to \langle X, \mathsf{just}\rangle \subseteq_f \mathbb{B}\big) & (\forall^2 I), 6 \end{array}$$

$$\therefore\ s_2 = \lambda p.\lambda z.y$$

- 

$$\begin{array}{lll} 1. & \Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{B} \to \langle -, \mathsf{nothing}\rangle \subseteq_f \mathbb{B}\big) & (Hip) \\ 2. & \Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{B} \to \langle X, \mathsf{just}\rangle \subseteq_f \mathbb{B}\big) & (Hip) \\ 3. & \Gamma \vdash r : \mathsf{Mb}\ \mathcal{A}(r) & (Var) \\ 4. & \Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ r : \mathbb{B}(\mathsf{f}\ r) & (\mu E^-), 1, 2, 3 \\ 5. & \Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ r : \mathbb{B}(\mathsf{iMb}\ r\ x\ y) & (Def\ \mathsf{f}), 4 \end{array}$$

$$
\begin{array}{lll}
6. & r : \mathsf{Mb}\ \mathcal{A}(r),\ x : \mathbb{B}(x) \vdash \lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r : \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}\ \ r\ x\ y) & (\to I), 5 \\
7. & r : \mathsf{Mb}\ \mathcal{A}(r) \vdash \lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r : \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}\ \ r\ x\ y) & (\to I), 6 \\
8. & \vdash \lambda r.\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r : \mathsf{Mb}\ \mathcal{A}(r) \to \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}\ \ r\ x\ y) & (\to I), 7
\end{array}
$$

$$
\begin{array}{lll}
9. & \vdash \lambda r.\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r : \forall y.\mathsf{Mb}\ \mathcal{A}(r) \to \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}\ \ r\ x\ y) & (\forall I), 8 \\
10. & \vdash \lambda r.\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r : \forall x.\forall y.\mathsf{Mb}\ \mathcal{A}(r) \to \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}\ \ r\ x\ y) & (\forall I), 9 \\
11. & \vdash \lambda r.\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r : \forall r.\forall x.\forall y.\mathsf{Mb}\ \mathcal{A}(r) \to \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}\ \ r\ x\ y) & (\forall I), 10
\end{array}
$$

$$\therefore \vdash \overline{\mathsf{iMb}} : \forall r.\forall x.\forall y.\mathsf{Mb}\ \mathcal{A}(r) \to \mathbb{B}(x) \to \mathbb{B}(y) \to \mathbb{B}(\mathsf{iMb}(r\ x\ y)),\ \text{donde}\ \ \overline{\mathsf{iMb}} = \lambda r.\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r$$

$\square$

- **Semántica operacional de la función** $\overline{\mathsf{iMb}}$
  - $\overline{\mathsf{iMb}}\ \overline{\mathsf{nothing}}\ x\ y \ = \ x.$

$$
\begin{array}{lll}
\overline{\mathsf{iMb}}\ \overline{\mathsf{nothing}}\ x\ y & \to & (\lambda r.\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r)\overline{\mathsf{nothing}}\ x\ y \ \to \\
(\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r)\ [r := \overline{\mathsf{nothing}}]\ x\ y & \to & (\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nothing}})\ x\ y \ \to \\
(\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nothing}})\ [x := x]\ y & \to & (\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nothing}})\ y \ \to \\
(\mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nothing}})\ [y := y] & \to & \mathsf{MIt}\, s_1\ s_2\ \overline{\mathsf{nothing}} \ \to \\
\mathsf{MIt}\, s_1\ s_2\ \ \mathsf{in}_{2,1} & \mapsto_\beta & s_1\ \ \mathsf{MIt}\, s_1\ s_2 \ \to \\
s_1\ (\mathsf{MIt}\, s_1\ s_2) & \to & (\lambda p.x)\ (\mathsf{MIt}\, s_1\ s_2) \ \to \\
x\ [p := \mathsf{MIt}\, s_1\ s_2] & \to & x
\end{array}
$$

$$\therefore\ \overline{\mathsf{iMb}}\ \overline{\mathsf{nothing}}\ x\ y \ = \ x$$

  - $\overline{\mathsf{iMb}}\ (\overline{\mathsf{just}}\ z)\ x\ y \ = \ y$

$$
\begin{array}{lll}
\overline{\mathsf{iMb}}\ (\overline{\mathsf{just}}\ z)\ x\ y & \to & (\lambda r.\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r)(\overline{\mathsf{just}}\ z)\ x\ y \ \to \\
(\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ r)\ [r := \overline{\mathsf{just}}\ z]\ x\ y & \to & (\lambda x.\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ (\overline{\mathsf{just}}\ z))\ x\ y \ \to \\
(\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ (\overline{\mathsf{just}}\ z))\ [x := x]\ y & \to & (\lambda y.\,\mathsf{MIt}\, s_1\ s_2\ (\overline{\mathsf{just}}\ z))\ y \ \to \\
(\mathsf{MIt}\, s_1\ s_2\ (\overline{\mathsf{just}}\ z))\ [y := y] & \to & \mathsf{MIt}\, s_1\ s_2\ (\overline{\mathsf{just}}\ z) \ \to \\
\mathsf{MIt}\, s_1\ s_2\ ((\lambda x.\,\mathsf{in}_{2,2}\, x)\ z) & \to & \mathsf{MIt}\, s_1\ s_2\ (\mathsf{in}_{2,2}\, x[x := z]) \ \to \\
\mathsf{MIt}\, s_1\ s_2\ (\mathsf{in}_{2,2}\, z) & \mapsto_\beta & s_2\ (\mathsf{MIt}\, s_1\ s_2)\ z \ \to \\
(\lambda p.\lambda z.y)\ (\mathsf{MIt}\, s_1\ s_2)\ z & \to & (\lambda z.y)\ [p := \mathsf{MIt}\, s_1\ s_2]\ z \ \to \\
(\lambda z.y)\ z & \to & y\ [z := z] \ \to \\
y & &
\end{array}
$$

$$\therefore\ \overline{\mathsf{iMb}}\ (\overline{\mathsf{just}}\ z)\ x\ y \ = \ y$$

### 4.1.5. Árboles binarios sin etiqueta

- **Definición**. Sea $X^{makebt} = \{makebt\ x\ y | x \in X \wedge y \in X\}$. El conjunto de los árboles binarios sin etiqueta se define como:

$$\mathsf{BTree} =_{def} \mu(\Phi) \quad \text{con} \quad \Phi(X) = \{node\} \cup X^{makebt}$$

- **Especificación funcional**.

$$data\ BinTree\ =\ Node \mid Makebt\ (BinTree)\ (BinTree)$$

- **Implementación**. Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle -, \mathsf{node}\rangle, \mathcal{C}_2 = \langle X, X, \mathsf{mBT}\rangle$. Los árboles binarios no etiquetados se definen como:

$$\mathsf{BTree} =_{def} \mu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2).$$

- **Reglas de construcción**.

$$\frac{}{\Gamma \vdash \mathsf{in}_{2,1} : \mathsf{BTree}(\mathsf{node})}\ (\mu I)$$

$$\frac{\Gamma \vdash r_1 : \mathsf{BTree}(t_1) \quad \Gamma \vdash r_2 : \mathsf{BTree}(t_2)}{\Gamma \vdash \mathsf{in}_{2,2}\, r_1 r_2 : \mathsf{BTree}(\mathsf{mBT}\ \ t_1\ t_2)}\ (\mu I)$$

- **Constructores**.
  - $\overline{\mathsf{node}} = \mathsf{in}_{2,1}$

    *Demostración.*

    $$1. \quad \Gamma \vdash \mathsf{in}_{2,1} : \mathsf{BTree}(\mathsf{node}) \quad (\mu I)$$

    $$\therefore\ \overline{\mathsf{node}} = \mathsf{in}_{2,1}$$

    □

  - $\overline{\mathsf{mBT}} = \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{mBT}} : \forall x.\forall y.\mathsf{BTree}(x) \to \mathsf{BTree}(y) \to \mathsf{BTree}(\mathsf{mBT}\ \ x\ y)$

    $$\begin{array}{lll}
1. & x : \mathsf{BTree}(x),\ y : \mathsf{BTree}(y) \vdash x : \mathsf{BTree}(x) & (Var)\\
2. & x : \mathsf{BTree}(x),\ y : \mathsf{BTree}(y) \vdash y : \mathsf{BTree}(y) & (Var)\\
3. & x : \mathsf{BTree}(x),\ y : \mathsf{BTree}(y) \vdash \mathsf{in}_{2,2}\, xy : \mathsf{BTree}(\mathsf{mBT}\ \ x\ y) & (\mu I), 1, 2\\
4. & x : \mathsf{BTree}(x) \vdash \lambda y.\, \mathsf{in}_{2,2}\, xy : \mathsf{BTree}(y) \to \mathsf{BTree}(\mathsf{mBT}\ \ x\ y) & (\to I), 3\\
5. & \vdash \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy : \mathsf{BTree}(x) \to \mathsf{BTree}(y) \to \mathsf{BTree}(\mathsf{mBT}\ \ x\ y) & (\to I), 4\\
6. & \vdash \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy : \forall y.\mathsf{BTree}(x) \to \mathsf{BTree}(y) \to \mathsf{BTree}(\mathsf{mBT}\ \ x\ y) & (\forall I), 5\\
7. & \vdash \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy : \forall x.\forall y.\mathsf{BTree}(x) \to \mathsf{BTree}(y) \to \mathsf{BTree}(\mathsf{mBT}\ \ x\ y) & (\forall I), 6
\end{array}$$

    $$\therefore\ \overline{\mathsf{mBT}} = \lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy$$

    □

- **Semántica operacional**

$$\begin{array}{rcl} \mathsf{MRec}\ \vec{s}\,(\mathsf{node}) & \mapsto_\beta & s_1 \\ \mathsf{MRec}\ \vec{s}\,(\mathsf{mBT}\ \ x\ y) & \mapsto_\beta & s_2\ \mathsf{MRec}\ x\ y \end{array}$$

Algunos ejemplos de extracción de programas son:

- **Destructor**. Sean $\mathsf{sts}$ un símbolo de función y $\mathbb{E}_{\mathsf{sts}} = \{\mathsf{sts}\ \ \mathsf{node} = \mathsf{nothing},\ \mathsf{sts}\ (\mathsf{mBT}\ \ x\ y) = \mathsf{just}\ (\mathsf{pair}\ \ x\ y)\}$. Entonces

$$\vdash \overline{\mathsf{sts}} : \forall x.\mathsf{BTree}(x) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{sts}\ \ x)$$

donde $\overline{\mathsf{sts}} = \lambda x.\,\mathsf{MRec}\ \ s_1\ s_2\ x,\ s_1 = \lambda p.\lambda q.\overline{\mathsf{nothing}}$ y $s_2 = \lambda p.\lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps))$.

*Demostración.* Sea $\mathsf{f} = \mathsf{sts}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{sts}} = \{\mathsf{f}\ \ \mathsf{node} = \mathsf{nothing},\ \mathsf{f}\ (\mathsf{mBT}\ \ x\ y) = \mathsf{just}(\mathsf{pair}\ \ x\ y)\}$. Sean $\Gamma := \{x : \mathsf{BTree}(x)\}$, $\Omega := \{p : X \subseteq \mathsf{BTree},\ q : X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})\}$ y $\Delta := \Omega \cup \{r : X(x),\ s : X(y)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle -, \mathsf{node}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})\big)$

$$\begin{array}{lll} 1. & \Gamma,\ \Omega \vdash \overline{\mathsf{nothing}} : \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{nothing}) & (Hip) \\ 2. & \Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{sts}}} \overline{\mathsf{nothing}} : \mathsf{Mb}\ \ \mathsf{f}(\mathsf{node}) & (Eq), 1 \\ 3. & \Gamma,\ \Omega \vdash \overline{\mathsf{nothing}} : \langle -, \mathsf{node}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2 \end{array}$$

4. $\Gamma,\ p : X \subseteq \mathsf{BTree} \vdash \lambda q.\overline{\mathsf{nothing}} : X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle -, \mathsf{node}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})$ $(\to I), 3$
5. $\Gamma \vdash \lambda p.\lambda q.\overline{\mathsf{nothing}} : X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle -, \mathsf{node}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})$ $(\to I), 4$

6. $\Gamma \vdash \lambda p.\lambda q.\overline{\mathsf{nothing}} : \forall X\big(X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle -, \mathsf{node}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})\big)$ $(\forall^2 I), 5$

$$\therefore\ s_1 = \lambda p.\lambda q.\overline{\mathsf{nothing}}$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})\big)$.

$$\begin{array}{lll} 1. & \Gamma,\ \Delta \vdash r : X(x) & (Hip) \\ 2. & \Gamma,\ \Delta \vdash p : X \subseteq \mathsf{BTree} & (Var) \\ 3. & \Gamma,\ \Delta \vdash p : \forall x.X(x) \to \mathsf{BTree}(x) & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2 \\ 4. & \Gamma,\ \Delta \vdash p : X(x) \to \mathsf{BTree}(x) & (\forall E), 3, [x := x] \quad lll \\ 5. & \Gamma,\ \Delta \vdash pr : \mathsf{BTree}(x) & (\to E), 1, 4 \\ 6. & \Gamma,\ \Delta \vdash s : X(y) & (Var) \\ 7. & \Gamma,\ \Delta \vdash p : X(y) \to \mathsf{BTree}(y) & (\forall E), 3, [x := y] \\ 8. & \Gamma,\ \Delta \vdash ps : \mathsf{BTree}(y) & (\to E), 6, 7 \end{array}$$

| | | | |
|---|---|---|---|
| 9. | $\Gamma,\ \Delta \vdash \overline{\mathsf{pair}} : \forall x.\forall y.\mathsf{BTree}(x) \to \mathsf{BTree}(y) \to (\mathsf{BTreeXBTree})(\mathsf{pair}\ \ x\ y)$ | $(Hip)$ |
| 10. | $\Gamma,\ \Delta \vdash \overline{\mathsf{pair}} : \forall y.\mathsf{BTree}(x) \to \mathsf{BTree}(y) \to (\mathsf{BTreeXBTree})(\mathsf{pair}\ \ x\ y)$ | $(\forall E), 9, [x := x]$ |
| 11. | $\Gamma,\ \Delta \vdash \overline{\mathsf{pair}} : \mathsf{BTree}(x) \to \mathsf{BTree}(y) \to (\mathsf{BTreeXBTree})(\mathsf{pair}\ \ x\ y)$ | $(\forall E), 10, [y := y]$ |
| 12. | $\Gamma,\ \Delta \vdash \overline{\mathsf{pair}}(pr) : \mathsf{BTree}(y) \to (\mathsf{BTreeXBTree})(\mathsf{pair}\ \ x\ y)$ | $(\to E), 5, 11$ |
| 13. | $\Gamma,\ \Delta \vdash \overline{\mathsf{pair}}(pr)(ps) : (\mathsf{BTreeXBTree})(\mathsf{pair}\ \ x\ y)$ | $(\to E), 8, 12$ |
| 14. | $\Gamma,\ \Delta \vdash \overline{\mathsf{just}} : \forall x.\,(\mathsf{BTreeXBTree})(x) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{just}\ \ x)$ | $(Var)$ |
| 15. | $\Gamma,\ \Delta \vdash \overline{\mathsf{just}} : (\mathsf{BTreeXBTree})(\mathsf{pair}\ \ x\ y) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{just}(\mathsf{pair}\ \ x\ y))$ | $(\forall E), 14, [x := \mathsf{pair}\ \ x\ y]$ |
| 16. | $\Gamma,\ \Delta \vdash \overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{just}(\mathsf{pair}\ \ x\ y))$ | $(\to E), 13, 15$ |
| 17. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{sts}}} \overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(Eq), 16$ |

| | | |
|---|---|---|
| 18. | $\Gamma,\ \Omega,\ r : X(x) \vdash \lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : X(y) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\to I), 17$ |
| 19. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : X(x) \to X(y) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\to I), 18$ |
| 20. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : \forall y.X(x) \to X(y) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\forall I), 19$ |

| | | |
|---|---|---|
| 21. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : \forall x.\forall y.X(x) \to X(y) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\forall I), 20$ |
| 22. | $\Gamma,\ \Omega \vdash \lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 21$ |

| | | |
|---|---|---|
| 23. | $\Gamma,\ p : X \subseteq \mathsf{BTree} \vdash \lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})$ | $(\to I), 22$ |
| 24. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})$ | $(\to I), 23$ |
| 25. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps)) : \forall X\big(X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})\big)$ | $(\forall^2 I), 24$ |

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}(\overline{\mathsf{pair}}(pr)(ps))$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle -, \mathsf{node}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(X \subseteq \mathsf{BTree} \to X \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree}) \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash x : \mathsf{BTree}(x)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MRec}\, s_1\ s_2\ x : \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{f}\ \ x)$ | $(\mu E), 1, 2, 3$ |
| 5. | $\Gamma \vdash \mathsf{MRec}\, s_1\ s_2\ x : \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{sts}\ \ x)$ | $(Def\ \ \mathsf{f}), 4$ |
| 6. | $\vdash \lambda x.\,\mathsf{MRec}\, s_1\ s_2\ x : \mathsf{BTree}(x) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{sts}\ \ x)$ | $(\to I), 5$ |
| 7. | $\vdash \lambda x.\,\mathsf{MRec}\, s_1\ s_2\ x : \forall x.\mathsf{BTree}(x) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{sts}\ \ x)$ | $(\forall I), 6$ |

$$\therefore \vdash \overline{\mathsf{sts}} : \forall x.\mathsf{BTree}(x) \to \mathsf{Mb}\ \ (\mathsf{BTreeXBTree})(\mathsf{sts}\ \ x),\ \ \text{donde}\ \ \ \overline{\mathsf{sts}} = \lambda x.\,\mathsf{MRec}\ \ s_1\ s_2\ x.$$

□

- **Semántica operacional de la función** $\overline{\mathsf{sts}}$

- $\overline{\mathsf{sts}}\ \overline{\mathsf{node}} \ = \ \overline{\mathsf{nothing}}$.

$$\begin{array}{llll}
\overline{\mathsf{sts}}\ \overline{\mathsf{node}} & \to & (\lambda x.\, \mathsf{MRec}\ \ s_1\ s_2\ x)\ \overline{\mathsf{node}} & \to \\
(\mathsf{MRec}\ \ s_1\ s_2\ x)\ [x := \overline{\mathsf{node}}] & \to & \mathsf{MRec}\ \ s_1\ s_2\ \overline{\mathsf{node}} & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ \ \mathsf{in}_{2,1} & \mapsto_\beta & s_1\ (\lambda x.x)\ (\mathsf{MRec}\ \ s_1\ s_2) & \to \\
(\lambda p.\lambda q.\overline{\mathsf{nothing}})\ (\lambda x.x)\ (\mathsf{MRec}\ \ s_1\ s_2) & \to & (\lambda q.\overline{\mathsf{nothing}})\ [p := \lambda x.x]\ (\mathsf{MRec}\ \ s_1\ s_2) & \to \\
(\lambda q.\overline{\mathsf{nothing}})\ (\mathsf{MRec}\ \ s_1\ s_2) & \to & \overline{\mathsf{nothing}}\ [q := \mathsf{MRec}\ \ s_1\ s_2] & \to \\
\overline{\mathsf{nothing}} & & &
\end{array}$$

$$\therefore\ \overline{\mathsf{sts}}\ \overline{\mathsf{node}} \ = \ \overline{\mathsf{nothing}}$$

- $\overline{\mathsf{sts}}\ (\overline{\mathsf{mBT}}\ x\ y) \ = \ \overline{\mathsf{just}}\ (\overline{\mathsf{pair}}\ x\ y)$.

$$\begin{array}{ll}
\overline{\mathsf{sts}}\ (\overline{\mathsf{mBT}}\ x\ y) & \to \\
(\lambda x.\, \mathsf{MRec}\ \ s_1\ s_2\ x)\ (\overline{\mathsf{mBT}}\ x\ y) & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ x\ [x := (\overline{\mathsf{mBT}}\ x\ y)] & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ (\overline{\mathsf{mBT}}\ x\ y) & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ ((\lambda x.\lambda y.\, \mathsf{in}_{2,2}\, xy)\ x\ y) & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ ((\lambda y.\, \mathsf{in}_{2,2}\, xy)\ [x := x]\ y) & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ ((\lambda y.\, \mathsf{in}_{2,2}\, xy)\ y) & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ ((\mathsf{in}_{2,2}\, xy)\ [y := y]) & \to \\
\mathsf{MRec}\ \ s_1\ s_2\ (\mathsf{in}_{2,2}\, xy) & \mapsto_\beta \\
s_2\ (\lambda z.z)\ (\mathsf{MRec}\, s_1\ s_2)\ x\ y & \to \\
(\lambda p.\lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}(pr)(ps)))\ (\lambda z.z)\ (\mathsf{MRec}\, s_1\ s_2)\ x\ y & \to \\
(\lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}(pr)(ps)))\ [p := \lambda z.z]\ (\mathsf{MRec}\, s_1\ s_2)\ x\ y & \to \\
(\lambda q.\lambda r.\lambda s.\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}((\lambda z.z)r)((\lambda z.z)s)))\ (\mathsf{MRec}\, s_1\ s_2)\ x\ y & \to \\
(\lambda r.\lambda s.\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}((\lambda z.z)r)((\lambda z.z)s)))\ [q := \mathsf{MRec}\, s_1\ s_2]\ x\ y & \to \\
(\lambda r.\lambda s.\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}((\lambda z.z)r)((\lambda z.z)s)))\ x\ y & \to \\
(\lambda s.\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}((\lambda z.z)r)((\lambda z.z)s)))\ [r := x]\ y & \to \\
(\lambda s.\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}((\lambda z.z)x)((\lambda z.z)s)))\ y & \to \\
(\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}((\lambda z.z)x)((\lambda z.z)s)))\ [s := y] & \to \\
\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}((\lambda z.z)x)((\lambda z.z)y)) & \to \\
\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}(z[z := x])((\lambda z.z)y)) & \to \\
\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}\ x\ ((\lambda z.z)y)) & \to \\
\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}\ x\ (z[z := y])) & \to \\
\overline{\mathsf{just}}\ (\overline{\mathsf{pair}}\ x\ y) &
\end{array}$$

$$\therefore\ \overline{\mathsf{sts}}\ (\overline{\mathsf{mBT}}\ x\ y) \ = \ \overline{\mathsf{just}}\ (\overline{\mathsf{pair}}\ x\ y)$$

- **Isnode**. Sean $\mathsf{isn}$ un símbolo de función y $\mathbb{E}_{\mathsf{isn}} = \{\mathsf{isn}(\mathsf{node}) = \mathsf{true},\ \mathsf{isn}(\mathsf{mBT}\ \ x\ y) = \mathsf{false}\}$. La función que determina si dado un árbol binario éste es un nodo se define como:

$$\vdash \overline{\mathsf{isn}} : \forall x.\mathsf{BTree}(x) \to \mathbb{B}(\mathsf{isn}\ \ x)$$

donde $\overline{\mathsf{isn}} = \lambda x.\, \mathsf{MIt}\ \ s_1\ s_2\ x$, $s_1 = \lambda p.\overline{\mathsf{true}}$ y $s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{false}}$.

*Demostración.* Sea $\mathsf{f} = \mathsf{isn}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{isn}} = \{\mathsf{f}(\mathsf{node}) = \mathsf{true},\ \mathsf{f}(\mathsf{mBT}\ \ x\ y) = \mathsf{false}\}$. Sea $\Gamma := \{x : \mathsf{BTree}(x)\}$, $\Omega := \{p : X \subseteq_f \mathbb{B}\}$ y $\Delta := \Omega \cup \{q : X(x),\ r : X(y)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{B}\ \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{B}\big)$

$$\begin{array}{lll}
1. & \Gamma,\ \Omega \vdash \overline{\mathsf{true}} : \mathbb{B}(\mathsf{true}) & (Hip) \\
2. & \Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{isn}}} \overline{\mathsf{true}} : \mathbb{B}(\mathsf{f}(\mathsf{node})) & (Eq), 1 \\
3. & \Gamma,\ \Omega \vdash \overline{\mathsf{true}} : \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{B} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 2 \\
4. & \Gamma \vdash \lambda p.\overline{\mathsf{true}} : X \subseteq_f \mathbb{B} \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{B} & (\to I), 3 \\
5. & \Gamma \vdash \lambda p.\overline{\mathsf{true}} : \forall X\big(X \subseteq_f \mathbb{B} \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{B}\big) & (\forall^2 I), 4
\end{array}$$

$$\therefore\ s_1 = \lambda p.\overline{\mathsf{true}}$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{B} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{B}\big)$

$$\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash \overline{\mathsf{false}} : \mathbb{B}(\mathsf{false}) & (Hip) \\
2. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{isn}}} \overline{\mathsf{false}} : \mathbb{B}(\mathsf{f}(\mathsf{mBT}\ \ x\ y)) & (Eq), 1 \\
3. & \Gamma,\ \Omega,\ q : X(x) \vdash \lambda r.\overline{\mathsf{false}} : X(y) \to \mathbb{B}(\mathsf{f}(\mathsf{mBT}\ \ x\ y)) & (\to I), 2 \\
4. & \Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{false}} : X(x) \to X(y) \to \mathbb{B}(\mathsf{f}(\mathsf{mBT}\ \ x\ y)) & (\to I), 3 \\
5. & \Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{false}} : \forall y.X(x) \to X(y) \to \mathbb{B}(\mathsf{f}(\mathsf{mBT}\ \ x\ y)) & (\forall I), 4 \\
6. & \Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{false}} : \forall x.\forall y.X(x) \to X(y) \to \mathbb{B}(\mathsf{f}(\mathsf{mBT}\ \ x\ y)) & (\forall I), 5 \\
7. & \Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{false}} : \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{B} & (Def\ \mathcal{C} \subseteq_f \mathcal{K}), 6 \\
8. & \Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{false}} : X \subseteq_f \mathbb{B} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{B} & (\to I), 7 \\
9. & \Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{false}} : \forall X\big(X \subseteq_f \mathbb{B} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{B}\big) & (\forall^2 I), 8
\end{array}$$

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{false}}$$

-
$$\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{B} \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{B}\big) & (Hip) \\
2. & \Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{B} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{B}\big) & (Hip) \\
3. & \Gamma \vdash x : \mathsf{BTree}(x) & (Var) \\
4. & \Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{B}(\mathsf{f}\ \ x) & (\mu E^-), 1, 2, 3 \\
5. & \Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{B}(\mathsf{isn}\ \ x) & (Def\ \ \mathsf{f}), 4 \\
6. & \vdash \lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x : \mathsf{BTree}(x) \to \mathbb{B}(\mathsf{isn}\ \ x) & (\to I), 5 \\
7. & \vdash \lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x : \forall x.\mathsf{BTree}(x) \to \mathbb{B}(\mathsf{isn}\ \ x) & (\forall I), 6
\end{array}$$

$\therefore \vdash \overline{\mathsf{isn}} : \forall x.\mathsf{BTree}(x) \to \mathbb{B}(\mathsf{isn}\ \ x)$, donde $\overline{\mathsf{isn}} = \lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x$

□

- **Número de nodos**. Sean $\mathsf{nn}$ un símbolo de función y $\mathbb{E}_{\mathsf{nn}} = \{\mathsf{nn}(\mathsf{node}) = \mathsf{suc}(\mathsf{zero}),\ \mathsf{nn}(\mathsf{mBT}\ \ x\ y) = \mathsf{suc}(\mathsf{sum}(\mathsf{nn}\ \ x \quad \mathsf{nn}\ \ y))\}$. La función que determina el número de nodos de un árbol binario se define como:

$$\vdash \overline{\mathsf{nn}} : \forall x.\mathsf{BTree}(x) \to \mathbb{N}(\mathsf{nn}\ \ x)$$

donde $\overline{\mathsf{nn}} = \lambda x.\, \mathsf{MIt}\ \ s_1\ s_2\ x$, $s_1 = \lambda p.\overline{\mathsf{suc}}(\overline{\mathsf{zero}})$ y $s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr))$.

*Demostración.* Sea $\mathsf{f} = \mathsf{nn}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{nn}} = \{\mathsf{f}(\mathsf{node}) = \mathsf{suc}(\mathsf{zero}),\ \mathsf{f}(\mathsf{mBT}\ \ x\ y) = \mathsf{suc}(\mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y))\}$. Sea $\Gamma := \{x : \mathsf{BTree}(x)\}$, $\Omega := \{p : X \subseteq_f \mathbb{N}\}$ y $\Delta := \{q : X(x),\ r : X(y)\}$.

- $\Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{N}\ \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{N}\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Omega \vdash \overline{\mathsf{zero}} : \mathbb{N}(\mathsf{zero})$ | $(Hip)$ |
| 2. | $\Gamma,\ \Omega \vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ \ x)$ | $(Hip)$ |
| 3. | $\Gamma,\ \Omega \vdash \overline{\mathsf{suc}} : \mathbb{N}(\mathsf{zero}) \to \mathbb{N}(\mathsf{suc}\ \ \mathsf{zero})$ | $(\forall E), 2, [x := \mathsf{zero}]$ |
| 4. | $\Gamma,\ \Omega \vdash \overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : \mathbb{N}(\mathsf{suc}\ \ \mathsf{zero})$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\ \Omega \vdash_{\mathbb{E}_{\mathsf{nn}}} \overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : \mathbb{N}(\mathsf{f}(\mathsf{node}))$ | $(Eq), 4$ |
| 6. | $\Gamma,\ \Omega \vdash \overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{N}$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 5$ |
| 7. | $\Gamma \vdash \lambda p.\overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : X \subseteq_f \mathbb{N} \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{N}$ | $(\to I), 6$ |
| 8. | $\Gamma \vdash \lambda p.\overline{\mathsf{suc}}(\overline{\mathsf{zero}}) : \forall X\big(X \subseteq_f \mathbb{N} \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{N}\big)$ | $(\forall^2 I), 7$ |

$$\therefore\ s_1 = \lambda p.\overline{\mathsf{suc}}(\overline{\mathsf{zero}})$$

- $\Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{N}\big)$.

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash p : X \subseteq_f \mathbb{N}$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash p : \forall x.X(x) \to \mathbb{N}(\mathsf{f}\ \ x)$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 1$ |
| 3. | $\Gamma,\ \Delta \vdash p : X(x) \to \mathbb{N}(\mathsf{f}\ \ x)$ | $(\forall E), 2, [x := x]$ |
| 4. | $\Gamma,\ \Delta \vdash q : X(x)$ | $(Var)$ |
| 5. | $\Gamma,\ \Delta \vdash pq : \mathbb{N}(\mathsf{f}\ \ x)$ | $(\to E), 3, 4$ |
| 6. | $\Gamma,\ \Delta \vdash p : X(y) \to \mathbb{N}(\mathsf{f}\ \ y)$ | $(\forall E), 2, [x := y]$ |
| 7. | $\Gamma,\ \Delta \vdash r : X(y)$ | $(Var)$ |
| 8. | $\Gamma,\ \Delta \vdash pr : \mathbb{N}(\mathsf{f}\ \ y)$ | $(\to E), 6, 7$ |
| 9. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}(n\ m))$ | $(Hip)$ |
| 10. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \forall m.\mathbb{N}(\mathsf{f}\ \ x) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}(\mathsf{f}\ \ x\ \ m))$ | $(\forall E), 9, [n := \mathsf{f}\ \ x]$ |
| 11. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \mathbb{N}(\mathsf{f}\ \ x) \to \mathbb{N}(\mathsf{f}\ \ y) \to \mathbb{N}(\mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y))$ | $(\forall E), 10, [m := \mathsf{f}\ \ y]$ |
| 12. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}}(pq) : \mathbb{N}(\mathsf{f}\ \ y) \to \mathbb{N}(\mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y))$ | $(\to E), 5, 11$ |
| 13. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}}(pq)(pr) : \mathbb{N}(\mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y))$ | $(\to E), 8, 12$ |
| 14. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ \ x)$ | $(Hip)$ |

| | | |
|---|---|---|
| 15. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \mathbb{N}(\mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y)) \to \mathbb{N}(\mathsf{suc}(\mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y)))$ | $(\forall E), 14, [x := \mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y)]$ |
| 16. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : \mathbb{N}(\mathsf{suc}(\mathsf{sum}(\mathsf{f}\ \ x\ \ \ \mathsf{f}\ \ y)))$ | $(\to E), 13, 15$ |

| | | |
|---|---|---|
| 17. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : \mathbb{N}(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(Eq), 16$ |
| 18. | $\Gamma,\ \Omega,\ q : X(x) \vdash \lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : X(y) \to \mathbb{N}(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\to I), 17$ |
| 19. | $\Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : X(x) \to X(y) \to \mathbb{N}(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\to I), 18$ |
| 20. | $\Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : \forall y.X(x) \to X(y) \to \mathbb{N}(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\forall I), 19$ |
| 21. | $\Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : \forall x.\forall y.X(x) \to X(y) \to \mathbb{N}(\mathsf{f}(\mathsf{mBT}\ \ x\ y))$ | $(\forall I), 20$ |

| | | |
|---|---|---|
| 22. | $\Gamma,\ \Omega \vdash \lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{N}$ | $(Def\ \mathcal{C} \subseteq_f \mathcal{K}), 21$ |
| 23. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : X \subseteq_f \mathbb{N} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{N}$ | $(\to I), 22$ |
| 24. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr)) : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{N}\big)$ | $(\forall^2 I), 23$ |

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{suc}}(\overline{\mathsf{sum}}(pq)(pr))$$

- 

$$
\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle -, \mathsf{node}\rangle \subseteq_f \mathbb{N}\big) & (Hip) \\
2. & \Gamma \vdash s_2 : \forall X\big(X \subseteq_f \mathbb{N} \to \langle X, X, \mathsf{mBT}\rangle \subseteq_f \mathbb{N}\big) & (Hip) \\
3. & \Gamma \vdash x : \mathsf{BTree}(x) & (Var) \\
4. & \Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{N}(\mathsf{f}\ \ x) & (\mu E^-), 1, 2, 3 \\
5. & \Gamma \vdash \mathsf{MIt}\, s_1\ s_2\ x : \mathbb{N}(\mathsf{nn}\ \ x) & (Def\ \ \mathsf{f}), 4 \\
6. & \vdash \lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x : \mathsf{BTree}(x) \to \mathbb{N}(\mathsf{nn}\ \ x) & (\to I), 5 \\
7. & \vdash \lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x : \forall x.\mathsf{BTree}(x) \to \mathbb{N}(\mathsf{nn}\ \ x) & (\forall I), 6
\end{array}
$$

$\therefore \vdash \overline{\mathsf{nn}} : \forall x.\mathsf{BTree}(x) \to \mathbb{N}(\mathsf{nn}\ \ x)$, donde $\overline{\mathsf{nn}} = \lambda x.\, \mathsf{MIt}\, s_1\ s_2\ x$.

$\square$

Con esto terminamos nuestra serie de ejemplos con tipos inductivos.

## 4.2. Extracción de programas con tipos coinductivos

En esta sección desarrollamos ejemplos que involucran a los siguientes tipos:

- Listas estrictamente infinitas (streams) sobre un tipo A
- Colistas (listas potencialmente infinitas) no vacías sobre un tipo C
- Producto cartesiano de dos tipos
- Árboles binarios completos estrictamente infinitos con etiquetas sobre un tipo A

### 4.2.1. Streams sobre A

- **Definición**. Los streams sobre un conjunto A se define como:

$$\mathbb{S}_{\mathcal{A}} =_{def} \nu(\Phi) \quad \text{con} \quad \Phi(X) = A^{head} \cap X^{tail}$$

- **Especificación funcional**

$$codata\ Stream\ a =\ Head\ a \mid Tail\ (Stream\ a)$$

- **Implementación**. Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle \mathcal{A}, \mathsf{head}\rangle$, $\mathcal{C}_2 = \langle X, \mathsf{tail}\rangle$. Los streams sobre A se definen como:

$$\mathbb{S}_{\mathcal{A}} =_{def} \nu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2).$$

- **Reglas de construcción**

$$\frac{\Gamma \vdash r : \mathbb{S}_{\mathcal{A}}(t_1)}{\Gamma \vdash \mathsf{out}_{2,1}\, r : \mathcal{A}(\mathsf{head}\ \ t_1)}\ (\nu E)$$

$$\frac{\Gamma \vdash r : \mathbb{S}_{\mathcal{A}}(t_2)}{\Gamma \vdash \mathsf{out}_{2,2}\, r : \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ \ t_2)}\ (\nu E)$$

- **Destructores**
  - $\overline{\mathsf{head}} = \lambda x.\, \mathsf{out}_{2,1}\, x$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{head}} : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathcal{A}(\mathsf{head}\ \ x)$

$$\begin{array}{lll}
1. & x : \mathbb{S}_{\mathcal{A}}(x) \vdash x : \mathbb{S}_{\mathcal{A}}(x) & (Var) \\
2. & x : \mathbb{S}_{\mathcal{A}}(x) \vdash \mathsf{out}_{2,1}\, x : \mathcal{A}(\mathsf{head}\ \ x) & (\nu E), 1 \\
3. & \vdash \lambda x.\, \mathsf{out}_{2,1}\, x : \mathbb{S}_{\mathcal{A}}(x) \to \mathcal{A}(\mathsf{head}\ \ x) & (\to I), 2 \\
4. & \vdash \lambda x.\, \mathsf{out}_{2,1}\, x : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{head}\ \ x) & (\forall I), 3
\end{array}$$

$$\therefore\ \overline{\mathsf{head}} = \lambda x.\, \mathsf{out}_{2,1}\, x$$

□

  - $\overline{\mathsf{tail}} = \lambda x.\, \mathsf{out}_{2,2}\, x$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{tail}} : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ \ x)$

$$\begin{array}{lll}
1. & x : \mathbb{S}_{\mathcal{A}}(x) \vdash x : \mathbb{S}_{\mathcal{A}}(x) & (Var) \\
2. & x : \mathbb{S}_{\mathcal{A}}(x) \vdash \mathsf{out}_{2,2}\, x : \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ \ x) & (\nu E), 1 \\
3. & \vdash \lambda x.\, \mathsf{out}_{2,2}\, x : \mathbb{S}_{\mathcal{A}}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ \ x) & (\to I), 2 \\
4. & \vdash \lambda x.\, \mathsf{out}_{2,2}\, x : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ \ x) & (\forall I), 3
\end{array}$$

$$\therefore\ \overline{\mathsf{tail}} = \lambda x.\, \mathsf{out}_{2,2}\, x$$

□

- **Semántica operacional**

$$\begin{array}{rcl}
\mathsf{head}(\mathsf{MCoIt}\ \ \vec{s}\, x) & \mapsto_\beta & s_1\ \ \mathsf{MCoIt}\ \ x \\
\mathsf{tail}(\mathsf{MCoIt}\ \ \vec{s}\, x) & \mapsto_\beta & s_2\ \ \mathsf{MCoIt}\ \ x
\end{array}$$

Algunos ejemplos de extracción de programas son:

- **Constructor**. Sean $\mathsf{cons}$ un símbolo de función y $\mathbb{E}_{\mathsf{cons}} = \{\mathsf{head}(\mathsf{cons}\ \ x\ y) = x,\ \mathsf{tail}(\mathsf{cons}\ \ x\ y) = y\}$. Entonces

$$\vdash \mathsf{cons} : \forall x.\forall y.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(y) \to \mathbb{S}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$$

donde $\vdash \overline{\mathsf{cons}} = \lambda x.\lambda y.\, \mathsf{MCoRec}\ \ s_1\ s_2\ y,\ s_1 = \lambda p.\lambda q.\lambda r.x$ y $s_2 = \lambda p.\lambda q.\lambda r.pr$.

*Demostración.* Sea $\mathsf{f}\ \ x = \mathsf{cons}$ tal que $\mathsf{f}\ \ x = \mathsf{cons}\ \ x\ y$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{cons}} = \{\mathsf{head}(\mathsf{f}\ \ x\ \ y) = x,\ \mathsf{tail}(\mathsf{f}\ \ x\ \ y) = y\}$. Sea $\Gamma := \{x : \mathcal{A}(x),\ y : \mathbb{S}_{\mathcal{A}}(y)\}$, $\Omega := \{p : \mathbb{S}_{\mathcal{A}} \subseteq X,\ q : \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X\}$ y $\Delta := \Omega \cup \{r : \mathbb{S}_{\mathcal{A}}(y)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle \mathcal{A}, \mathsf{head}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash x : \mathcal{A}(x)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{cons}}} x : \mathcal{A}(\mathsf{head}(\mathsf{f}\ \ x\ \ y))$ | $(Eq), 1$ |
| 3. | $\Gamma,\ \Omega \vdash \lambda r.x : \mathbb{S}_{\mathcal{A}}(y) \to \mathcal{A}(\mathsf{head}(\mathsf{f}\ \ x\ \ y))$ | $(\to I), 2$ |
| 4. | $\Gamma,\ \Omega \vdash \lambda r.x : \forall y.\mathbb{S}_{\mathcal{A}}(y) \to \mathcal{A}(\mathsf{head}(\mathsf{f}\ \ x\ \ y))$ | $(\forall I), 3$ |
| 5. | $\Gamma,\ \Omega \vdash \lambda r.x : \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle \mathcal{A}, \mathsf{head}\rangle$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 4$ |
| 6. | $\Gamma,\ p : \mathbb{S}_{\mathcal{A}} \subseteq X \vdash \lambda q.\lambda r.x : \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle \mathcal{A}, \mathsf{head}\rangle$ | $(\to I), 5$ |
| 7. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.x : \mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle \mathcal{A}, \mathsf{head}\rangle$ | $(\to I), 6$ |
| 8. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.x : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle \mathcal{A}, \mathsf{head}\rangle\big)$ | $(\forall^2 I), 7$ |

$$\therefore\ s_1 = \lambda p.\lambda q.\lambda r.x$$

- $\Gamma \vdash s_2 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle X, \mathsf{tail}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash r : \mathbb{S}_{\mathcal{A}}(y)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash p : \mathbb{S}_{\mathcal{A}} \subseteq X$ | $(Var)$ |
| 3. | $\Gamma,\ \Delta \vdash p : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to X(x)$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 2$ |
| 4. | $\Gamma,\ \Delta \vdash p : \mathbb{S}_{\mathcal{A}}(y) \to X(y)$ | $(\forall E), 3, [x := y]$ |
| 5. | $\Gamma,\ \Delta \vdash pr : X(y)$ | $(\to E), 1, 4$ |
| 6. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{cons}}} pr : X(\mathsf{tail}(\mathsf{f}\ \ x\ \ y))$ | $(Eq), 5$ |
| 7. | $\Gamma,\ \Omega \vdash \lambda r.pr : \mathbb{S}_{\mathcal{A}}(y) \to X(\mathsf{tail}(\mathsf{f}\ \ x\ \ y))$ | $(\to I), 6$ |
| 8. | $\Gamma,\ \Omega \vdash \lambda r.pr : \forall y.\mathbb{S}_{\mathcal{A}}(y) \to X(\mathsf{tail}(\mathsf{f}\ \ x\ \ y))$ | $(\forall I), 7$ |
| 9. | $\Gamma,\ \Omega \vdash \lambda r.pr : \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle X, \mathsf{tail}\rangle$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 8$ |
| 10. | $\Gamma,\ p : \mathbb{S}_{\mathcal{A}} \subseteq X \vdash \lambda q.\lambda r.pr : \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle X, \mathsf{tail}\rangle$ | $(\to I), 9$ |
| 11. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.pr : \mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle X, \mathsf{tail}\rangle$ | $(\to I), 10$ |
| 12. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.pr : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle X, \mathsf{tail}\rangle\big)$ | $(\forall^2 I)11$ |

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.pr$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle \mathcal{A}, \mathsf{head}\rangle\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} X \to \mathbb{S}_{\mathcal{A}} \subseteq_{f\ x} \langle X, \mathsf{tail}\rangle\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash y : \mathbb{S}_{\mathcal{A}}(y)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{A}}(\mathsf{f}\ \ x\ \ y)$ | $(\nu I), 1, 2, 3$ |
| 5. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$ | $(Def\ \ \mathsf{f}), 4$ |
| 6. | $x : \mathcal{A}(x) \vdash \lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{A}}(y) \to \mathbb{S}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$ | $(\to I), 5$ |
| 7. | $\vdash \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : \mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(y) \to \mathbb{S}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$ | $(\to I), 6$ |
| 8. | $\vdash \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : \forall y.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(y) \to \mathbb{S}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$ | $(\forall I), 7$ |
| 9. | $\vdash \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : \forall x.\forall y.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(y) \to \mathbb{S}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$ | $(\forall I), 8$ |

$\therefore \vdash \overline{\mathsf{cons}} : \forall x.\forall y.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(y) \to \mathbb{S}_{\mathcal{A}}(\mathsf{cons}\ \ x\ y)$, donde $\overline{\mathsf{cons}} = \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y$.

$\square$

- **From**. Sean $\mathsf{from}$ un símbolo de función y $\mathbb{E}_{\mathsf{from}} = \{\mathsf{head}\ (\mathsf{from}\ \ x) = x,\ \mathsf{tail}\ (\mathsf{from}\ \ x) = \mathsf{from}\ (\mathsf{suc}\ \ x)\}$. Entonces

$$\vdash \overline{\mathsf{from}} : \forall x.\mathbb{N}(x) \to \mathbb{S}_{\mathcal{N}}(\mathsf{from}\ \ x)$$

donde $\overline{\mathsf{from}} = \lambda x.\,\mathsf{MCoIt}\ \ s_1\ s_2\ x,\ s_1 = \lambda p.\lambda q.x$ y $s_2 = \lambda p.\lambda q.p(\overline{\mathsf{suc}}q)$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f} = \mathsf{from}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{f}} = \{\mathsf{head}\ (\mathsf{f}\ \ x) = x,\ \mathsf{tail}\ (\mathsf{f}\ \ x) = \mathsf{f}\ (\mathsf{suc}\ \ x)\}$. Sea $\Gamma := \{x : \mathbb{N}(x)\}$, $\Omega := \{p : \mathbb{N} \subseteq_f X\}$ y $\Delta := \Omega \cup \{q : \mathbb{N}(x)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle \mathbb{N}, \mathsf{head}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash x : \mathbb{N}(x)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{from}}} x : \mathbb{N}(\mathsf{head}(\mathsf{f}\ \ x))$ | $(Eq), 1$ |
| 3. | $\Gamma,\ \Omega \vdash \lambda q.x : \mathbb{N}(x) \to \mathbb{N}(\mathsf{head}(\mathsf{f}\ \ x))$ | $(\to I), 2$ |
| 4. | $\Gamma,\ \Omega \vdash \lambda q.x : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{head}(\mathsf{f}\ \ x))$ | $(\forall I), 3$ |
| 5. | $\Gamma,\ \Omega \vdash \lambda q.x : \mathbb{N} \subseteq_f \langle \mathbb{N}, \mathsf{head}\rangle$ | $(Def\ \mathcal{K} \subseteq_f \mathcal{C}), 4$ |
| 6. | $\Gamma \vdash \lambda p.\lambda q.x : \mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle \mathbb{N}, \mathsf{head}\rangle$ | $(\to I), 5$ |
| 7. | $\Gamma \vdash \lambda p.\lambda q.x : \forall X\big(\mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle \mathbb{N}, \mathsf{head}\rangle\big)$ | $(\forall^2 I), 6$ |

$$\therefore\ \ s_1 = \lambda p.\lambda q.x$$

- $\Gamma \vdash s_2 : \forall X\big(\mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash q : \mathbb{N}(x)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ \ x)$ | $(Var)$ |
| 3. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ \ x)$ | $(\forall E), 2, [x := x]$ |
| 4. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}}q : \mathbb{N}(\mathsf{suc}\ \ x)$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\ \Delta \vdash p : \mathbb{N} \subseteq_f X$ | $(Var)$ |
| 6. | $\Gamma,\ \Delta \vdash p : \forall x.\mathbb{N}(x) \to X(\mathsf{f}\ \ x)$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 5$ |
| 7. | $\Gamma,\ \Delta \vdash p : \mathbb{N}(\mathsf{suc}\ \ x) \to X(\mathsf{f}(\mathsf{suc}\ \ x))$ | $(\forall E), 6, [x := \mathsf{suc}\ \ x]$ |
| 8. | $\Gamma,\ \Delta \vdash p(\overline{\mathsf{suc}}q) : X(\mathsf{f}(\mathsf{suc}\ \ x))$ | $(\to E), 4, 7$ |
| 9. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{from}}} p(\overline{\mathsf{suc}}q) : X(\mathsf{tail}(\mathsf{from}\ \ x))$ | $(Eq), 8$ |
| 10. | $\Gamma,\ \Omega \vdash \lambda q.p(\overline{\mathsf{suc}}q) : \mathbb{N}(x) \to X(\mathsf{tail}(\mathsf{from}\ \ x))$ | $(\to I), 9$ |
| 11. | $\Gamma,\ \Omega \vdash \lambda q.p(\overline{\mathsf{suc}}q) : \forall x.\mathbb{N}(x) \to X(\mathsf{tail}(\mathsf{from}\ \ x))$ | $(\forall I), 10$ |
| 12. | $\Gamma,\ \Omega \vdash \lambda q.p(\overline{\mathsf{suc}}q) : \mathbb{N} \subseteq_f \langle X, \mathsf{tail}\rangle$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 11$ |
| 13. | $\Gamma \vdash \lambda p.\lambda q.p(\overline{\mathsf{suc}}q) : \mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle X, \mathsf{tail}\rangle$ | $(\to I), 12$ |
| 14. | $\Gamma \vdash \lambda p.\lambda q.p(\overline{\mathsf{suc}}q) : \forall X\big(\mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$ | $(\forall^2 I), 13$ |

$$\therefore\ \ s_2 = \lambda p.\lambda q.p(\overline{\mathsf{suc}}q)$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(\mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle \mathbb{N}, \mathsf{head}\rangle\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(\mathbb{N} \subseteq_f X \to \mathbb{N} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash x : \mathbb{N}(x)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MCoIt}\, s_1\ s_2\ x : \mathbb{S}_{\mathcal{N}}(\mathsf{f}\ \ x)$ | $(\nu I^-), 1, 2, 3$ |
| 5. | $\Gamma \vdash \mathsf{MCoIt}\, s_1\ s_2\ x : \mathbb{S}_{\mathcal{N}}(\mathsf{from}\ \ x)$ | $(Def\ \ \mathsf{f}), 4$ |
| 6. | $\vdash \lambda x.\,\mathsf{MCoIt}\, s_1\ s_2\ x : \mathbb{N}(x) \to \mathbb{S}_{\mathcal{N}}(\mathsf{from}\ \ x)$ | $(\to I), 5$ |
| 7. | $\vdash \lambda x.\,\mathsf{MCoIt}\, s_1\ s_2\ x : \forall x.\mathbb{N}(x) \to \mathbb{S}_{\mathcal{N}}(\mathsf{from}\ \ x)$ | $(\forall I), 6$ |

$\therefore \vdash \overline{\mathsf{from}} : \forall x.\mathbb{N}(x) \to \mathbb{S}_{\mathcal{N}}(\mathsf{from}\ \ x)$, donde $\overline{\mathsf{from}} = \lambda x.\,\mathsf{MColt}\, s_1\ s_2\ x$.

$\square$

- **Semántica operacional de la función** $\overline{\mathsf{from}}$
  - $\overline{\mathsf{head}}\ (\overline{\mathsf{from}}\ x) \ = \ x$

$$
\begin{array}{llll}
\overline{\mathsf{head}}\ (\overline{\mathsf{from}}\ x) & \to & \overline{\mathsf{head}}\ ((\lambda x.\,\mathsf{MColt}\, s_1\ s_2\ x)\ x) & \to \\
\overline{\mathsf{head}}\ ((\mathsf{MColt}\, s_1\ s_2\ x)\ [x := x]) & \to & \overline{\mathsf{head}}\ (\mathsf{MColt}\, s_1\ s_2\ x) & \to \\
(\lambda x.\,\mathsf{out}_{2,1}\ \ x)\ (\mathsf{MColt}\, s_1\ s_2\ x) & \to & (\mathsf{out}_{2,1}\ \ x)\ [x := (\mathsf{MColt}\, s_1\ s_2\ x)] & \to \\
\mathsf{out}_{2,1}\ (\mathsf{MColt}\, s_1\ s_2\ x) & \mapsto_\beta & s_1\ (\mathsf{MColt}\, s_1\ s_2\ )\ x & \to \\
(\lambda p.\lambda q.x)\ (\mathsf{MColt}\, s_1\ s_2\ )\ x & \to & (\lambda q.x)\ [p := (\mathsf{MColt}\, s_1\ s_2\ )]\ x & \to \\
(\lambda q.x)\ x & \to & x\ [q := x] & \to \\
x & & &
\end{array}
$$

$$\therefore\ \overline{\mathsf{head}}\ (\overline{\mathsf{from}}\ x) \ = \ x$$

  - $\overline{\mathsf{tail}}\ (\overline{\mathsf{from}}\ \ x) \ = \ \overline{\mathsf{from}}\ (\overline{\mathsf{suc}}\ \ x)$. La prueba se divide en dos partes: se demuestra primero que $\overline{\mathsf{tail}}\ (\overline{\mathsf{from}}\ x) \ = \ \mathsf{MColt}\, s_1\ s_2\ (\overline{\mathsf{suc}}\ \ x)$ y después que $\overline{\mathsf{from}}\ (\overline{\mathsf{suc}}\ x) \ = \ \mathsf{MColt}\, s_1\ s_2\ (\overline{\mathsf{suc}}\ x)$.

$$
\begin{array}{llll}
\overline{\mathsf{tail}}\ (\overline{\mathsf{from}}\ x) & \to & \overline{\mathsf{tail}}\ ((\lambda x.\,\mathsf{MColt}\, s_1\ s_2\ x)\ x) & \to \\
\overline{\mathsf{tail}}((\mathsf{MColt}\, s_1\ s_2\ x)[x := x]) & \to & \overline{\mathsf{tail}}(\mathsf{MColt}\, s_1\ s_2\ x) & \to \\
(\lambda x.\,\mathsf{out}_{2,2}\ \ x)\ (\mathsf{MColt}\, s_1\ s_2\ x) & \to & (\mathsf{out}_{2,2}\ \ x)\ [x := (\mathsf{MColt}\, s_1\ s_2\ x)] & \to \\
\mathsf{out}_{2,2}\ (\mathsf{MColt}\, s_1\ s_2\ x) & \mapsto_\beta & s_2\ (\mathsf{MColt}\, s_1\ s_2\ )\ x & \to \\
(\lambda p.\lambda q.p\ (\overline{\mathsf{suc}}\ q))\ (\mathsf{MColt}\, s_1\ s_2\ )\ x & \to & (\lambda q.p\ (\overline{\mathsf{suc}}\ q))\ [p := \mathsf{MColt}\, s_1\ s_2]\ x & \to \\
(\lambda q.\,\mathsf{MColt}\, s_1\ s_2\ (\overline{\mathsf{suc}}\ q))\ x & \to & \mathsf{MColt}\, s_1\ s_2\ (\overline{\mathsf{suc}}\ q)\ [q := x] & \to \\
\mathsf{MColt}\, s_1\ s_2\ (\overline{\mathsf{suc}}\ x) & & &
\end{array}
$$

$$
\begin{array}{llll}
\overline{\mathsf{from}}\ (\overline{\mathsf{suc}}\ x) & \to & (\lambda x.\,\mathsf{MColt}\, s_1\ s_2\ x)\ (\overline{\mathsf{suc}}\ x) & \to \\
\mathsf{MColt}\, s_1\ s_2\ x\ [x := \overline{\mathsf{suc}}\ x] & \to & \mathsf{MColt}\, s_1\ s_2\ (\overline{\mathsf{suc}}\ x) &
\end{array}
$$

$$\therefore\ \overline{\mathsf{tail}}\ (\overline{\mathsf{from}}\ x) \ = \ \overline{\mathsf{from}}\ (\overline{\mathsf{suc}}\ x).$$

- **Iteración**. Sean $\mathsf{iter}$ un símbolo de función y $\mathbb{E}_{\mathsf{iter}} = \{\mathsf{head}\ (\mathsf{iter}\ \ f\ \ x) = x,\ \mathsf{tail}\ (\mathsf{iter}\ \ f\ \ x) = \mathsf{iter}\ \ f\ \ (f\ x)\}$. Entonces

$$\vdash \overline{\mathsf{iter}} : \big(\forall x.\mathcal{A}(x) \to \mathcal{A}(f\ x)\big) \to \forall x.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{iter}\ \ f\ \ x)$$

donde $\overline{\mathsf{iter}} = \lambda s.\lambda x.\,\mathsf{MColt}\ \ s_1\ s_2\ x$, $s_1 = \lambda p.\lambda q.q$ y $s_2 = \lambda p.\lambda q.p(sq)$.

*Demostración.* Sea $\overline{\mathsf{f}} = \mathsf{iter}\ \ f$ tal que $\overline{\mathsf{f}}\ \ x = \mathsf{iter}\ \ f\ \ x$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{iter}} = \{\mathsf{head}(\overline{\mathsf{f}}\ x) = x,\ \mathsf{tail}(\overline{\mathsf{f}}\ x) = \overline{\mathsf{f}}(f\ x)\}$. Sea $\Gamma := \{s : \forall x.\mathcal{A}(x) \to \mathcal{A}(f\ x),\ x : \mathcal{A}(x)\}$, $\Omega := \{p : \mathcal{A} \subseteq_{\overline{\mathsf{f}}} X\}$ y $\Delta := \Omega \cup \{q : \mathcal{A}(x)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{\mathsf{f}}} \langle \mathcal{A}, \mathsf{head}\rangle\big)$

$$\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash q : \mathcal{A}(x) & (Var) \\
2. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{iter}}} q : \mathcal{A}(\mathsf{head}(\bar{\mathsf{f}}\ x)) & (Eq), 1 \\
3. & \Gamma,\ \Omega \vdash \lambda q.q : \mathcal{A}(x) \to \mathcal{A}(\mathsf{head}(\bar{\mathsf{f}}\ x)) & (\to I), 2 \\
4. & \Gamma,\ \Omega \vdash \lambda q.q : \forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{head}(\bar{\mathsf{f}}\ x)) & (\forall I), 3 \\
5. & \Gamma,\ \Omega \vdash \lambda q.q : \mathcal{A} \subseteq_{\bar{\mathsf{f}}} \langle \mathcal{A}, \mathsf{head}\rangle & (Def\ \mathcal{K} \subseteq_f\ \ C), 4 \\
6. & \Gamma \vdash \lambda p.\lambda q.q : \mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{\mathsf{f}}} \langle \mathcal{A}, \mathsf{head}\rangle & (\to I), 5 \\
7. & \Gamma \vdash \lambda p.\lambda q.q : \forall X\big(\mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{\mathsf{f}}} \langle \mathcal{A}, \mathsf{head}\rangle\big) & (\forall^2 I), 6
\end{array}$$

$$\therefore\ s_1 = \lambda p.\lambda q.q$$

- $\Gamma \vdash s_2 : \forall X\big(\mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{\mathsf{f}}} \langle X, \mathsf{tail}\rangle\big)$

$$\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash q : \mathcal{A}(x) & (Var) \\
2. & \Gamma,\ \Delta \vdash s : \forall x.\mathcal{A}(x) \to \mathcal{A}(f\ x) & (Var) \\
3. & \Gamma,\ \Delta \vdash s : \mathcal{A}(x) \to \mathcal{A}(f\ x) & (\forall E), 2, [x := x] \\
4. & \Gamma,\ \Delta \vdash sq : \mathcal{A}(f\ x) & (\to E), 1, 3 \\
5. & \Gamma,\ \Delta \vdash p : \mathcal{A} \subseteq_{\bar{\mathsf{f}}} X & (Var) \\
6. & \Gamma,\ \Delta \vdash p : \forall x.\mathcal{A}(x) \to X(\bar{\mathsf{f}}(x)) & (Def\ \mathcal{K} \subseteq_f\ \ C), 5 \\
7. & \Gamma,\ \Delta \vdash p : \mathcal{A}(f\ x) \to X(\bar{\mathsf{f}}(f\ x)) & (\forall E), 6, [x := f\ x] \\
8. & \Gamma,\ \Delta \vdash p(sq) : X(\bar{\mathsf{f}}(f\ x)) & (\to E), 4, 7 \\
9. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{iter}}} p(sq) : X(\mathsf{tail}(\bar{\mathsf{f}}\ x)) & (Eq), 8 \\
10. & \Gamma,\ \Omega \vdash \lambda q.p(sq) : \mathcal{A}(x) \to X(\mathsf{tail}(\bar{\mathsf{f}}\ x)) & (\to I), 9 \\
11. & \Gamma,\ \Omega \vdash \lambda q.p(sq) : \forall x.\mathcal{A}(x) \to X(\mathsf{tail}(\bar{\mathsf{f}}\ x)) & (\forall I), 10 \\
12. & \Gamma,\ \Omega \vdash \lambda q.p(sq) : \mathcal{A} \subseteq_{\bar{f}} \langle X, \mathsf{tail}\rangle & (Def\ \mathcal{K} \subseteq_f\ \ C), 11 \\
13. & \Gamma \vdash \lambda p.\lambda q.p(sq) : \mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{f}} \langle X, \mathsf{tail}\rangle & (\to I), 12 \\
14. & \Gamma \vdash \forall X\big(\lambda p.\lambda q.p(sq) : \mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{f}} \langle X, \mathsf{tail}\rangle\big) & (\forall^2 I), 13
\end{array}$$

$$\therefore\ s_2 = \lambda p.\lambda q.p(sq)$$

-

$$\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(\mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{\mathsf{f}}} \langle \mathcal{A}, \mathsf{head}\rangle\big) & (Hip) \\
2. & \Gamma \vdash s_2 : \forall X\big(\mathcal{A} \subseteq_{\bar{\mathsf{f}}} X \to \mathcal{A} \subseteq_{\bar{\mathsf{f}}} \langle X, \mathsf{tail}\rangle\big) & (Hip) \\
3. & \Gamma \vdash x : \mathcal{A}(x) & (Var) \\
4. & \Gamma \vdash \mathsf{MColt}\, s_1\ s_2\ x : \mathbb{S}_{\mathcal{A}}(\bar{\mathsf{f}}\ x) & (\nu I^-), 1, 2, 3 \\
5. & s : \forall x.\mathcal{A}(x) \to \mathcal{A}(f\ x) \vdash \lambda x.\, \mathsf{MColt}\, s_1\ s_2\ x : \mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(\bar{\mathsf{f}}\ x) & (\to I), 4 \\
6. & s : \forall x.\mathcal{A}(x) \to \mathcal{A}(f\ x) \vdash \lambda x.\, \mathsf{MColt}\, s_1\ s_2\ x : \forall x.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(\bar{\mathsf{f}}\ x) & (\forall I), 5 \\
7. & \vdash \lambda s.\lambda x.\, \mathsf{MColt}\, s_1\ s_2\ x : (\forall x.\mathcal{A}(x) \to \mathcal{A}(f\ x)) \to \forall x.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(\bar{\mathsf{f}}\ x) & (\to I), 6
\end{array}$$

$\therefore \vdash \overline{\mathsf{iter}} : \big(\forall x.\mathcal{A}(x) \to \mathcal{A}(f\ x)\big) \to \forall x.\mathcal{A}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{iter}\ \ f\ \ x)$, donde $\overline{\mathsf{iter}} = \lambda s.\lambda x.\, \mathsf{MColt}\, s_1\ s_2\ x$.

$\square$

- **Map**. Sean $\mathsf{map}, \mathsf{g}$ símbolos de función tales que $\mathbb{E}_{\mathsf{map}} = \{\mathsf{head}(\mathsf{map}\ \ \mathsf{g}\ \ l) = \mathsf{g}(\mathsf{head}\ \ l),\ \mathsf{tail}(\mathsf{map}\ \ \mathsf{g}\ \ l) = \mathsf{map}\ \ \mathsf{g}(\mathsf{tail}\ \ l)\}$ y $\mathsf{g} : \mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x))$. Entonces

$$\vdash \overline{\mathsf{map}} : \big(\forall x.\mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x))\big) \to \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{B}}(\mathsf{map}\ \ \mathsf{g}\ \ \ z)$$

donde $\overline{\mathsf{map}} = \lambda x.\lambda y.\,\mathsf{MCoIt}\;\; s_1\; s_2\; y$, $s_1 = \lambda p.\lambda q.x(\overline{\mathsf{head}}q)$ y $s_2 = \lambda p.\lambda q.p(\overline{\mathsf{tail}}q)$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f} = \mathsf{map}\;\; g$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{map}} = \{\mathsf{head}(\mathsf{f}\;\; l) = \mathsf{g}(\mathsf{head}\;\; l),\; \mathsf{tail}(\mathsf{f}\;\; l) = \mathsf{f}(\mathsf{tail}\;\; l)\}$. Sea $\Gamma := \{x : \forall x.\mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x)),\; y : \mathbb{S}_{\mathcal{A}}(z)\}$, $\Omega := \{p : \mathbb{S}_{\mathcal{A}} \subseteq_f X\}$ y $\Delta := \Omega \cup \{q : \mathbb{S}_{\mathcal{A}}(r)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{B}, \mathsf{head}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\; \Delta \vdash q : \mathbb{S}_{\mathcal{A}}(r)$ | $(Var)$ |
| 2. | $\Gamma,\; \Delta \vdash \overline{\mathsf{head}} : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathcal{A}(\mathsf{head}\;\; x)$ | $(Hip)$ |
| 3. | $\Gamma,\; \Delta \vdash \overline{\mathsf{head}} : \mathbb{S}_{\mathcal{A}}(r) \to \mathcal{A}(\mathsf{head}\;\; r)$ | $(\forall E), 2, [x := r]$ |
| 4. | $\Gamma,\; \Delta \vdash \overline{\mathsf{head}}q : A(\mathsf{head}\;\; r)$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\; \Delta \vdash x : \forall x.\mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x))$ | $(Var)$ |
| 6. | $\Gamma,\; \Delta \vdash x : \mathcal{A}(\mathsf{head}\;\; r) \to \mathcal{B}(\mathsf{g}(\mathsf{head}\;\; r))$ | $(\forall E), 5, [x := \mathsf{head}\;\; r]$ |
| 7. | $\Gamma,\; \Delta \vdash x(\overline{\mathsf{head}}q) : \mathcal{B}(\mathsf{g}(\mathsf{head}\;\; r))$ | $(\to E), 4, 6$ |
| 8. | $\Gamma,\; \Delta \vdash_{\mathbb{E}_{\mathsf{map}}} x(\overline{\mathsf{head}}q) : \mathcal{B}(\mathsf{head}(\mathsf{f}\;\; r))$ | $(Eq), 7$ |
| 9. | $\Gamma,\; \Omega \vdash \lambda q.x(\overline{\mathsf{head}}q) : \mathbb{S}_{\mathcal{A}}(r) \to \mathcal{B}(\mathsf{head}(\mathsf{f}\;\; r))$ | $(\to I), 8$ |
| 10. | $\Gamma,\; \Omega \vdash \lambda q.x(\overline{\mathsf{head}}q) : \forall r.\mathbb{S}_{\mathcal{A}}(r) \to \mathcal{B}(\mathsf{head}(\mathsf{f}\;\; r))$ | $(\forall I), 9$ |
| 11. | $\Gamma,\; \Omega \vdash \lambda q.x(\overline{\mathsf{head}}q) : \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{B}, \mathsf{head}\rangle$ | $(Def\; \mathcal{K} \subseteq_f\;\; C), 10$ |
| 12. | $\Gamma \vdash \lambda p.\lambda q.x(\overline{\mathsf{head}}q) : \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{B}, \mathsf{head}\rangle$ | $(\to I), 11$ |
| 13. | $\Gamma \vdash \lambda p.\lambda q.x(\overline{\mathsf{head}}q) : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{B}, \mathsf{head}\rangle\big)$ | $(\forall^2 I), 12$ |

$$\therefore\;\; s_1 = \lambda p.\lambda q.x(\overline{\mathsf{head}}q)$$

- $\Gamma \vdash s_2 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\; \Delta \vdash q : \mathbb{S}_{\mathcal{A}}(r)$ | $(Var)$ |
| 2. | $\Gamma,\; \Delta \vdash \overline{\mathsf{tail}} : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\;\; x)$ | $(Hip)$ |
| 3. | $\Gamma,\; \Delta \vdash \overline{\mathsf{tail}} : \mathbb{S}_{\mathcal{A}}(r) \to \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\;\; r)$ | $(\forall E), 2, [x := r]$ |
| 4. | $\Gamma,\; \Delta \vdash \overline{\mathsf{tail}}q : \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\;\; r)$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\; \Delta \vdash p : \mathbb{S}_{\mathcal{A}} \subseteq_f X$ | $(Var)$ |
| 6. | $\Gamma,\; \Delta \vdash p : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to X(\mathsf{f}\;\; x)$ | $(Def\; \mathcal{K} \subseteq_f\;\; C), 5$ |
| 7. | $\Gamma,\; \Delta \vdash p : \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\;\; r) \to X(\mathsf{f}(\mathsf{tail}\;\; r))$ | $(\forall E), 6, [x := \mathsf{tail}\;\; r]$ |
| 8. | $\Gamma,\; \Delta \vdash p(\overline{\mathsf{tail}}q) : X(\mathsf{f}(\mathsf{tail}\;\; r))$ | $(\to E), 4, 7$ |
| 9. | $\Gamma,\; \Delta \vdash_{\mathbb{E}_{\mathsf{map}}} p(\overline{\mathsf{tail}}q) : X(\mathsf{tail}(\mathsf{f}\;\; r))$ | $(Eq), 8$ |
| 10. | $\Gamma,\; \Omega \vdash \lambda q.p(\overline{\mathsf{tail}}q) : \mathbb{S}_{\mathcal{A}}(r) \to X(\mathsf{tail}(\mathsf{f}\;\; r))$ | $(\to I), 9$ |
| 11. | $\Gamma,\; \Omega \vdash \lambda q.p(\overline{\mathsf{tail}}q) : \forall r.\mathbb{S}_{\mathcal{A}}(r) \to X(\mathsf{tail}(\mathsf{f}\;\; r))$ | $(\forall I), 10$ |
| 12. | $\Gamma,\; \Omega \vdash \lambda q.p(\overline{\mathsf{tail}}q) : \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle$ | $(Def\; \mathcal{K} \subseteq_f\;\; C), 11$ |
| 13. | $\Gamma \vdash \lambda p.\lambda q.p(\overline{\mathsf{tail}}q) : \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle$ | $(\to I), 12$ |
| 14. | $\Gamma \vdash \lambda p.\lambda q.p(\overline{\mathsf{tail}}q) : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$ | $(\forall^2 I), 13$ |

$$\therefore\;\; s_2 = \lambda p.\lambda q.p(\overline{\mathsf{tail}}q)$$

- 

$$\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{B}, \mathsf{head}\rangle\big) & (Hip)\\
2. & \Gamma \vdash s_2 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle\big) & (Hip)\\
3. & \Gamma \vdash y : \mathbb{S}_{\mathcal{A}}(z) & (Var)\\
4. & \Gamma \vdash \mathsf{MColt}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{B}}(\mathsf{f}\ \ z) & (\nu I^-), 1, 2, 3\\
5. & \Gamma \vdash \mathsf{MColt}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{B}}(\mathsf{map}\ \ \mathsf{g}\ \ \ z) & (Def\ \ \mathsf{f}), 4
\end{array}$$

$$\begin{array}{lll}
6. & x : \forall x.\mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x)) \vdash \lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{B}}(\mathsf{map}\ \ \mathsf{g}\ \ \ z) & (\to I), 5\\
7. & x : \forall x.\mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x)) \vdash \lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y : \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{B}}(\mathsf{map}\ \ \mathsf{g}\ \ \ z) & (\forall I), 6\\
8. & \vdash \lambda x.\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y : \big(\forall x.\mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x))\big) \to \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{B}}(\mathsf{map}\ \ \mathsf{g}\ \ \ z) & (\to I), 7
\end{array}$$

$\therefore \vdash \overline{\mathsf{map}} : \big(\forall x.\mathcal{A}(x) \to \mathcal{B}(\mathsf{g}(x))\big) \to \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{B}}(\mathsf{map}\ \ \mathsf{g}\ \ \ z)$, donde $\overline{\mathsf{map}} = \lambda x.\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y$.

□

- **Maphd**. Sean $\mathsf{maphd}$, $\mathsf{g}$ símbolos de función tales que $\mathbb{E}_{\mathsf{maphd}} = \{\mathsf{head}(\mathsf{maphd}\ \ \mathsf{g}\ \ l) = \mathsf{g}(\mathsf{head}\ \ l),\ \mathsf{tail}(\mathsf{maphd}\ \ \mathsf{g}\ \ l) = \mathsf{tail}\ \ l\}$ y $\mathsf{g} : \mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x))$. Entonces

$$\vdash \overline{\mathsf{maphd}} : \big(\forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x))\big) \to \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{A}}(\mathsf{maphd}\ \ \mathsf{g}\ \ \ z)$$

donde $\overline{\mathsf{maphd}} = \lambda x.\lambda y.\, \mathsf{MCoRec}\ \ s_1\ s_2\ y$, $s_1 = \lambda p.\lambda q.\lambda r.x(\overline{\mathsf{head}}r)$ y $s_2 = \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{tail}}r)$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f} = \mathsf{maphd}\ \ g$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{maphd}} = \{\mathsf{head}(\mathsf{f}\ \ l) = \mathsf{g}(\mathsf{head}\ \ l),\ \mathsf{tail}(\mathsf{f}\ \ l) = \mathsf{tail}\ \ l\}$. Sea $\Gamma := \{x : \forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x)),\ y : \mathbb{S}_{\mathcal{A}}(z)\}$, $\Omega := \{p : \mathbb{S}_{\mathcal{A}} \subseteq X,\ q : \mathbb{S}_{\mathcal{A}} \subseteq_f X\}$ y $\Delta := \Omega \cup \{r : \mathbb{S}_{\mathcal{A}}(r)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{A}, \mathsf{head}\rangle\big)$

$$\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash r : \mathbb{S}_{\mathcal{A}}(r) & (Var)\\
2. & \Gamma,\ \Delta \vdash \overline{\mathsf{head}} : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathcal{A}(\mathsf{head}\ \ x) & (Hip)\\
3. & \Gamma,\ \Delta \vdash \overline{\mathsf{head}} : \mathbb{S}_{\mathcal{A}}(r) \to \mathcal{A}(\mathsf{head}\ \ r) & (\forall E), 2, [x := r]\\
4. & \Gamma,\ \Delta \vdash \overline{\mathsf{head}}r : A(\mathsf{head}\ \ r) & (\to E), 1, 3\\
5. & \Gamma,\ \Delta \vdash x : \forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x)) & (Var)\\
6. & \Gamma,\ \Delta \vdash x : \mathcal{A}(\mathsf{head}\ \ r) \to \mathcal{A}(\mathsf{g}(\mathsf{head}\ \ r)) & (\forall E), 5, [x := \mathsf{head}\ \ r]\\
7. & \Gamma,\ \Delta \vdash x(\overline{\mathsf{head}}r) : \mathcal{A}(\mathsf{g}(\mathsf{head}\ \ r)) & (\to E), 4, 6\\
8. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{map}}} x(\overline{\mathsf{head}}r) : \mathcal{A}(\mathsf{head}(\mathsf{f}\ \ r)) & (Eq), 7\\
9. & \Gamma,\ \Omega \vdash \lambda r.x(\overline{\mathsf{head}}r) : \mathbb{S}_{\mathcal{A}}(r) \to \mathcal{B}(\mathsf{head}(\mathsf{f}\ \ r)) & (\to I), 8\\
10. & \Gamma,\ \Omega \vdash \lambda r.x(\overline{\mathsf{head}}r) : \forall r.\mathbb{S}_{\mathcal{A}}(r) \to \mathcal{B}(\mathsf{head}(\mathsf{f}\ \ r)) & (\forall I), 9\\
11. & \Gamma,\ \Omega \vdash \lambda r.x(\overline{\mathsf{head}}r) : \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{A}, \mathsf{head}\rangle & (Def\ \mathcal{K} \subseteq_f\ \ C), 10
\end{array}$$

$$\begin{array}{lll}
12. & \Gamma,\ p : \mathbb{S}_{\mathcal{A}} \subseteq X \vdash \lambda q.\lambda r.x(\overline{\mathsf{head}}r) : \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{A}, \mathsf{head}\rangle & (\to I), 11\\
13. & \Gamma \vdash \lambda p.\lambda q.\lambda r.x(\overline{\mathsf{head}}r) : \mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{A}, \mathsf{head}\rangle & (\to I), 12\\
14. & \Gamma \vdash \lambda p.\lambda q.\lambda r.x(\overline{\mathsf{head}}r) : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{A}, \mathsf{head}\rangle\big) & (\forall^2 I), 13
\end{array}$$

$$\therefore\ s_1 = \lambda p.\lambda q.\lambda r.x(\overline{\mathsf{head}}r)$$

- $\Gamma \vdash s_2 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash r : \mathbb{S}_{\mathcal{A}}(r)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash \overline{\mathsf{tail}} : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ x)$ | $(Hip)$ |
| 3. | $\Gamma,\ \Delta \vdash \overline{\mathsf{tail}} : \mathbb{S}_{\mathcal{A}}(r) \to \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ r)$ | $(\forall E), 2, [x := r]$ |
| 4. | $\Gamma,\ \Delta \vdash \overline{\mathsf{tail}} r : \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ r)$ | $(\to), 1, 3$ |
| 5. | $\Gamma,\ \Delta \vdash p : \mathbb{S}_{\mathcal{A}} \subseteq X$ | $(Var)$ |
| 6. | $\Gamma,\ \Delta \vdash p : \forall x.\mathbb{S}_{\mathcal{A}}(x) \to X(x)$ | $(Def\ \mathcal{K} \subseteq_f\ C), 5$ |
| 7. | $\Gamma,\ \Delta \vdash p : \mathbb{S}_{\mathcal{A}}(\mathsf{tail}\ r) \to X(\mathsf{tail}\ r)$ | $(\forall E), 6, [x := \mathsf{tail}\ r]$ |
| 8. | $\Gamma,\ \Delta \vdash p(\overline{\mathsf{tail}} r) : X(\mathsf{tail}\ r)$ | $(\to E), 4, 7$ |
| 9. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{maphd}}} p(\overline{\mathsf{tail}} r) : X(\mathsf{tail}(\mathsf{f}\ r))$ | $(Eq), 8$ |
| 10. | $\Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{tail}} r) : \mathbb{S}_{\mathcal{A}}(r) \to X(\mathsf{tail}(\mathsf{f}\ r))$ | $(\to I), 9$ |
| 11. | $\Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{tail}} r) : \forall r.\mathbb{S}_{\mathcal{A}}(r) \to X(\mathsf{tail}(\mathsf{f}\ r))$ | $(\forall I), 10$ |
| 12. | $\Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{tail}} r) : \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle$ | $(Def\ \mathcal{K} \subseteq_f\ C), 11$ |

| | | |
|---|---|---|
| 13. | $\Gamma,\ p : \mathbb{S}_{\mathcal{A}} \subseteq X \vdash \lambda q.\lambda r.p(\overline{\mathsf{tail}} r) : \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle$ | $(\to I), 12$ |
| 14. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{tail}} r) : \mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle$ | $(\to I), 13$ |
| 15. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{tail}} r) : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$ | $(\forall^2 I), 14$ |

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{tail}} r)$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle \mathcal{A}, \mathsf{head}\rangle\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(\mathbb{S}_{\mathcal{A}} \subseteq X \to \mathbb{S}_{\mathcal{A}} \subseteq_f X \to \mathbb{S}_{\mathcal{A}} \subseteq_f \langle X, \mathsf{tail}\rangle\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash y : \mathbb{S}_{\mathcal{A}}(z)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{A}}(\mathsf{f}\ z)$ | $(\nu I), 1, 2, 3$ |
| 5. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{A}}(\mathsf{maphd}\ \mathsf{g}\ \ z)$ | $(Def\ \ \mathsf{f}), 4$ |

| | | |
|---|---|---|
| 6. | $x : \forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x)) \vdash \lambda y.\,\mathsf{MCoRec}\, s_1\ s_2\ y : \mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{A}}(\mathsf{maphd}\ \mathsf{g}\ \ z)$ | $(\to I), 5$ |
| 7. | $x : \forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x)) \vdash \lambda y.\,\mathsf{MCoRec}\, s_1\ s_2\ y : \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{A}}(\mathsf{maphd}\ \mathsf{g}\ \ z)$ | $(\forall I), 6$ |
| 8. | $\vdash \lambda x.\lambda y.\,\mathsf{MCoRec}\, s_1\ s_2\ y : \big(\forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x))\big) \to \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{A}}(\mathsf{maphd}\ \mathsf{g}\ \ z)$ | $(\to I), 7$ |

$\therefore \vdash \overline{\mathsf{maphd}} : \big(\forall x.\mathcal{A}(x) \to \mathcal{A}(\mathsf{g}(x))\big) \to \forall z.\mathbb{S}_{\mathcal{A}}(z) \to \mathbb{S}_{\mathcal{A}}(\mathsf{maphd}\ \mathsf{g}\ \ z)$, donde $\overline{\mathsf{maphd}} = \lambda x.\lambda y.\,\mathsf{MCoRec}\, s_1\ s_2\ y$.

$\square$

### 4.2.2. Colistas de elementos de C no vacías potencialmente infinitas

- **Definición**. Sean $C^{head} = \{x \in X | head(x) \in C\}$ y $X^{tail} = \{x \in X | tail(Maybe\, x) \in X\}$. El conjunto de colistas de elementos de C no vacías potencialmente infinitas se define como:

$$\mathsf{Colist}\ C\ =_{def} \nu(\Phi) \quad \text{con} \quad \Phi(X) = C^{head} \cap X^{tail}$$

- **Especificación funcional**

$$codata\ Colist\ c = \ Head\ c \mid Tail\ (Maybe\ Colist\ c)$$

- **Implementación**. Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle C, \mathsf{head}\rangle, \mathcal{C}_2 = \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle$. Las colistas de elementos de C no vacías potencialmente infinitas se definen como:

$$\mathsf{Colist}\ C =_{def} \nu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2).$$

- **Reglas de construcción**

$$\frac{\Gamma \vdash r : \mathsf{Colist}\ C(t) \quad \mathcal{C}_1 = \langle C, \mathsf{head}\rangle}{\Gamma \vdash \mathsf{out}_{2,1}\, r : C(\mathsf{head}\ \ t)}\ (\nu E)$$

$$\frac{\Gamma \vdash r : \mathsf{Colist}\ C(t) \quad \mathcal{C}_2 = \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle}{\Gamma \vdash \mathsf{out}_{2,2}\, r : \mathsf{Mb}\ \mathsf{Colist}\ C(\mathsf{tail}\ \ t)}\ (\nu E)$$

- **Destructores**

  - $\overline{\mathsf{head}} = \lambda x.\, \mathsf{out}_{2,1}\, x$

    Demostración. Hay que demostrar que $\vdash \overline{\mathsf{head}} : \forall x.\mathsf{Colist}\ C(x) \to C(\mathsf{head}\ \ x)$.

$$\begin{array}{lll}
1. & x : \mathsf{Colist}\ C(x) \vdash x : \mathsf{Colist}\ C(x) & (Var) \\
2. & x : \mathsf{Colist}\ C(x) \vdash \mathsf{out}_{2,1}\, x : \mathsf{Colist}\ C(\mathsf{head}\ \ x) & (\nu E), 1 \\
3. & \vdash \lambda x.\, \mathsf{out}_{2,1}\, x : \mathsf{Colist}\ C(x) \to C(\mathsf{head}\ \ x) & (\to I), 2 \\
4. & \vdash \lambda x.\, \mathsf{out}_{2,1}\, x : \forall x.\mathsf{Colist}\ C(x) \to C(\mathsf{head}\ \ x) & (\forall I), 3
\end{array}$$

$$\therefore\ \overline{\mathsf{head}} = \lambda x.\, \mathsf{out}_{2,1}\, x$$

  - $\overline{\mathsf{tail}} = \lambda x.\, \mathsf{out}_{2,2}\, x$

    Demostración. Hay que demostrar que $\vdash \overline{\mathsf{tail}} : \forall x.\mathsf{Colist}\ C(x) \to \mathsf{Mb}\ \mathsf{Colist}\ C(\mathsf{tail}\ \ x)$.

$$\begin{array}{lll}
1. & x : \mathsf{Colist}\ C(x) \vdash x : \mathsf{Colist}\ C(x) & (Var) \\
2. & x : \mathsf{Colist}\ C(x) \vdash \mathsf{out}_{2,2}\, x : \mathsf{Mb}\ \mathsf{Colist}\ C(\mathsf{tail}\ \ x) & (\nu E), 1 \\
3. & \vdash \lambda x.\, \mathsf{out}_{2,2}\, x : \mathsf{Colist}\ C(x) \to \mathsf{Mb}\ \mathsf{Colist}\ C(\mathsf{tail}\ \ x) & (\to I), 2 \\
4. & \vdash \lambda x.\, \mathsf{out}_{2,2}\, x : \forall x.\mathsf{Colist}\ C(x) \to \mathsf{Mb}\ \mathsf{Colist}\ C(\mathsf{tail}\ \ x) & (\forall I), 3
\end{array}$$

$$\therefore\ \overline{\mathsf{tail}} = \lambda x.\, \mathsf{out}_{2,2}\, x$$

- **Semántica operacional**

$$\begin{array}{rcl}
\mathsf{head}(\mathsf{MCoRec}\ \vec{s}\, x) & \mapsto_\beta & s_1\ \ \mathsf{MCoRec}\ \ x \\
\mathsf{tail}(\mathsf{MCoRec}\ \vec{s}\,\mathsf{Mb}\ x) & \mapsto_\beta & s_2\ \ \mathsf{MCoRec}\ \ x
\end{array}$$

Algunos ejemplos de extracción de programas son:

- **Constructor**. Sean $\mathsf{cons}$ un símbolo de función y $\mathbb{E}_{\mathsf{cons}} = \{\mathsf{head}\ (\mathsf{cons}\ \ x\ y) = x,\ \mathsf{tail}\ (\mathsf{cons}\ \ x\ y) = \mathsf{just}\ \ y\}$. Entonces

$$\vdash \overline{\mathsf{cons}} : \forall x.\forall y.C(x) \to \mathsf{Colist}\ C(y) \to \mathsf{Colist}\ C(\mathsf{cons}\ \ x\ y)$$

donde $\overline{\mathsf{cons}} = \lambda x.\lambda y.\,\mathsf{MCoRec}\ s_1\ s_2\ y$, $s_1 = \lambda p.\lambda q.\lambda r.x$ y $s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{just}}(pr)$.

*Demostración.* Sea $\mathsf{f}\ \ x = \mathsf{cons}$ tal que $\mathsf{f}\ \ x = \mathsf{cons}\ \ x\ \ y$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{cons}} = \{\mathsf{head}\ (\mathsf{f}\ \ x\ \ \ y) = x,\ \mathsf{tail}\ (\mathsf{f}\ \ x\ \ \ y) = \mathsf{just}\ \ y\}$. Sea $\Gamma := \{x : C(x),\ y : \mathsf{Colist}\ C(y)\}$, $\Omega := \{p : \mathsf{Colist}\ C \subseteq X,\ q : \mathsf{Colist}\ C \subseteq_{f\ x} X\}$ y $\Delta := \Omega \cup \{r : \mathsf{Colist}\ C(y)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash x : \mathcal{C}(x)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{cons}}} x : C(\mathsf{head}(\mathsf{f}\ \ x\ \ y))$ | $(Eq), 1$ |
| 3. | $\Gamma,\ \Omega \vdash \lambda r.x : \mathsf{Colist}\ C(y) \to C(\mathsf{head}(\mathsf{f}\ \ x\ \ y))$ | $(\to I), 2$ |
| 4. | $\Gamma,\ \Omega \vdash \lambda r.x : \forall y.\mathsf{Colist}\ C(y) \to C(\mathsf{head}(\mathsf{f}\ \ x\ \ y))$ | $(\forall I), 3$ |
| 5. | $\Gamma,\ \Omega \vdash \lambda r.x : \mathsf{Colist}\ C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 4$ |

| | | |
|---|---|---|
| 6. | $\Gamma,\ p : \mathsf{Colist}\ C \subseteq X \vdash \lambda q.\lambda r.x : \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle$ | $(\to I), 5$ |
| 7. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.x : \mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle$ | $(\to I), 6$ |
| 8. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.x : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle\big)$ | $(\forall^2 I), 7$ |

$$\therefore\ s_1 = \lambda p.\lambda q.\lambda r.x$$

- $\Gamma \vdash s_2 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash r : \mathsf{Colist}\ C(y)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash p : \mathsf{Colist}\ C \subseteq X$ | $(Var)$ |
| 3. | $\Gamma,\ \Delta \vdash p : \forall x.\mathsf{Colist}\ C(x) \to X(x)$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 2$ |
| 4. | $\Gamma,\ \Delta \vdash p : \mathsf{Colist}\ C(y) \to X(y)$ | $(\forall E), 3, [x := y]$ |
| 5. | $\Gamma,\ \Delta \vdash pr : X(y)$ | $(\to E), 1, 4$ |
| 6. | $\Gamma,\ \Delta \vdash \overline{\mathsf{just}} : \forall x.X(x) \to \mathsf{Mb}\ X(\mathsf{just}\ \ x)$ | $(Hip)$ |
| 7. | $\Gamma,\ \Delta \vdash \overline{\mathsf{just}} : X(y) \to \mathsf{Mb}\ X(\mathsf{just}\ \ y)$ | $(\forall E), 6, [x := y]$ |
| 8. | $\Gamma,\ \Delta \vdash \overline{\mathsf{just}}(pr) : \mathsf{Mb}\ X(\mathsf{just}\ \ y)$ | $(\to E), 5, 7$ |
| 9. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{cons}}} \overline{\mathsf{just}}(pr) : \mathsf{Mb}\ X(\mathsf{tail}(\mathsf{f}\ \ x\ \ y))$ | $(Eq), 8$ |
| 10. | $\Gamma,\ \Omega \vdash \lambda r.pr : \mathsf{Colist}\ C(y) \to X(\mathsf{tail}(\mathsf{f}\ \ x\ \ y))$ | $(\to I), 9$ |
| 11. | $\Gamma,\ \Omega \vdash \lambda r.pr : \forall y.\mathsf{Colist}\ C(y) \to X(\mathsf{tail}(\mathsf{f}\ \ x\ \ y))$ | $(\forall I)10$ |
| 12. | $\Gamma,\ \Omega \vdash \lambda r.pr : \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle$ | $(Def\ \mathcal{K} \subseteq_f\ \ C), 11$ |

| | | |
|---|---|---|
| 13. | $\Gamma,\ p : \mathsf{Colist}\ C \subseteq X \vdash \lambda q.\lambda r.pr : \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle$ | $(\to I), 12$ |
| 14. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.pr : \mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle$ | $(\to I), 13$ |
| 15. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.pr : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big)$ | $(\forall^2 I)14$ |

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{just}}(pr)$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big)$ | $(Hip)$ |

$$
\begin{array}{lll}
3. & \Gamma \vdash y : \mathsf{Colist}\ C(y) & (Var) \\
4. & \Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ y : \mathsf{Colist}\ C(\mathsf{f}\ \ x\ \ y) & (\nu I), 1, 2, 3 \\
5. & \Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ y : \mathsf{Colist}\ C(\mathsf{cons}\ \ x\ y) & (Def\ \ \mathsf{f}), 4
\end{array}
$$

$$
\begin{array}{lll}
6. & x : C(x) \vdash \lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : \mathsf{Colist}\ C(y) \to \mathsf{Colist}\ C(\mathsf{cons}\ \ x\ y) & (\to I), 5 \\
7. & \vdash \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : C(x) \to \mathsf{Colist}\ C(y) \to \mathsf{Colist}\ C(\mathsf{cons}\ \ x\ y) & (\to I), 6 \\
8. & \vdash \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : \forall y.C(x) \to \mathsf{Colist}\ C(y) \to \mathsf{Colist}\ C(\mathsf{cons}\ \ x\ y) & (\forall I), 7 \\
9. & \vdash \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y : \forall x.\forall y.C(x) \to \mathsf{Colist}\ C(y) \to \mathsf{Colist}\ C(\mathsf{cons}\ \ x\ y) & (\forall I), 8
\end{array}
$$

$$\therefore \vdash \overline{\mathsf{cons}} : \forall x.\forall y.C(x) \to \mathsf{Colist}\ C(y) \to \mathsf{Colist}\ C(\mathsf{cons}\ \ x\ y),\ \text{ donde }\ \overline{\mathsf{cons}} = \lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y.$$

□

- **Semántica operacional de la función** $\overline{\mathsf{cons}}$
  - $\overline{\mathsf{head}}\ (\overline{\mathsf{cons}}\ x\ y) \ = \ x.$

$$
\begin{array}{llll}
\overline{\mathsf{head}}\ (\overline{\mathsf{cons}}\ x\ y) & \to & \overline{\mathsf{head}}\ ((\lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y)\ x\ y) & \to \\
\overline{\mathsf{head}}\ ((\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y)\ [x := x]\ y) & \to & \overline{\mathsf{head}}\ ((\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y)\ y) & \to \\
\overline{\mathsf{head}}\ ((\mathsf{MCoRec}\, s_1\ s_2\ y)\ [y := y]) & \to & \overline{\mathsf{head}}\ (\mathsf{MCoRec}\, s_1\ s_2\ y) & \to \\
(\lambda x.\, \mathsf{out}_{2,1}\, x)\ (\mathsf{MCoRec}\, s_1\ s_2\ y) & \to & (\mathsf{out}_{2,1}\, x)\ [x := \mathsf{MCoRec}\, s_1\ s_2\ y] & \to \\
\mathsf{out}_{2,1}\ (\mathsf{MCoRec}\, s_1\ s_2\ y) & \mapsto_\beta & s_1\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to \\
(\lambda p.\lambda q.\lambda r.x)\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to & (\lambda q.\lambda r.x)\ [p := \lambda x.x]\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to \\
(\lambda q.\lambda r.x)\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to & (\lambda r.x)\ [q := \mathsf{MCoRec}\, s_1\ s_2]\ y & \to \\
(\lambda r.x)\ y & \to & x\ [r := y] & \to \\
x & & &
\end{array}
$$

$$\therefore\ \overline{\mathsf{head}}\ (\overline{\mathsf{cons}}\ x\ y) \ = \ x$$

  - $\overline{\mathsf{tail}}\ (\overline{\mathsf{cons}}\ x\ y) \ = \ \overline{\mathsf{just}}\ y.$

$$
\begin{array}{ll}
\overline{\mathsf{tail}}\ (\overline{\mathsf{cons}}\ x\ y) & \to \\
\overline{\mathsf{tail}}\ ((\lambda x.\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y)\ x\ y) & \to \\
\overline{\mathsf{tail}}\ ((\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y)\ [x := x]\ y) & \to \\
\overline{\mathsf{tail}}\ ((\lambda y.\, \mathsf{MCoRec}\, s_1\ s_2\ y)\ y) & \to \\
\overline{\mathsf{tail}}\ ((\mathsf{MCoRec}\, s_1\ s_2\ y)\ [y := y]) & \to \\
\overline{\mathsf{tail}}\ (\mathsf{MCoRec}\, s_1\ s_2\ y) & \to \\
(\lambda x.\, \mathsf{out}_{2,2}\, x)\ (\mathsf{MCoRec}\, s_1\ s_2\ y) & \to \\
(\mathsf{out}_{2,2}\, x)\ [x := \mathsf{MCoRec}\, s_1\ s_2\ y] & \to \\
\mathsf{out}_{2,2}\ (\mathsf{MCoRec}\, s_1\ s_2\ y) & \mapsto_\beta \\
s_2\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to \\
(\lambda p.\lambda q.\lambda r.\overline{\mathsf{just}}(pr))\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to \\
(\lambda q.\lambda r.\overline{\mathsf{just}}(pr))\ [p := \lambda x.x]\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to \\
\lambda q.\lambda r.\overline{\mathsf{just}}((\lambda x.x)r))\ (\mathsf{MCoRec}\, s_1\ s_2)\ y & \to \\
(\lambda r.\overline{\mathsf{just}}((\lambda x.x)r))\ [q := \mathsf{MCoRec}\, s_1\ s_2]\ y & \to \\
(\lambda r.\overline{\mathsf{just}}((\lambda x.x)r))\ y & \to \\
\overline{\mathsf{just}}\ ((\lambda x.x)r)\ [r := y] & \to \\
\overline{\mathsf{just}}\ ((\lambda x.x)y) & \to \\
\overline{\mathsf{just}}\ (x\ [x := y]) & \to \\
\overline{\mathsf{just}}\ y &
\end{array}
$$

$$\therefore\ \overline{\mathsf{tail}}\ (\overline{\mathsf{cons}}\ x\ y) \ = \ \overline{\mathsf{just}}\ y$$

- **Single**. Sean $\mathsf{single}$ un símbolo de función y $\mathbb{E}_{\mathsf{single}} = \{\mathsf{head}(\mathsf{single}\ \ x) = x,\ \mathsf{tail}(\mathsf{single}\ \ x) = \mathsf{nothing}\}$. Entonces

$$\vdash \overline{\mathsf{single}} : \forall x.C(x) \to \mathsf{Colist}\ C(\mathsf{single}\ \ x)$$

donde $\overline{\mathsf{single}} = \lambda x.\,\mathsf{MCoRec}\ \ s_1\ s_2\ x$, $s_1 = \lambda p.\lambda q.\lambda r.r$ y $s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{nothing}}$.

*Demostración.* Sea $\mathsf{f} = \mathsf{single}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{single}} = \{\mathsf{head}(\mathsf{f}\ \ x) = x,\ \mathsf{tail}(\mathsf{f}\ \ x) = \mathsf{nothing}\}$. Sea $\Gamma := \{x : C(x)\}$, $\Omega := \{p : \mathsf{Colist}\ C \subseteq X,\ q : C \subseteq_{f\ x} X\}$ y $\Delta := \Omega \cup \{r : C(y)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to C \subseteq_{f\ x} X \to C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle\big)$

1. $\Gamma,\ \Delta \vdash r : C(y)$ $(Var)$
2. $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{single}}} r : C(\mathsf{head}(\mathsf{f}\ \ y))$ $(Eq), 1$
3. $\Gamma,\ \Omega \vdash \lambda r.r : C(y) \to C(\mathsf{head}(\mathsf{f}\ \ y))$ $(\to I), 2$
4. $\Gamma,\ \Omega \vdash \lambda r.r : \forall y.\mathsf{Colist}\ C(y) \to C(\mathsf{head}(\mathsf{f}\ \ y))$ $(\forall I), 3$
5. $\Gamma,\ \Omega \vdash \lambda r.r : C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle$ $(Def\ \mathcal{K} \subseteq_f\ \ C), 4$
6. $\Gamma,\ p : \mathsf{Colist}\ C \subseteq X \vdash \lambda q.\lambda r.r : C \subseteq_{f\ x} X \to C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle$ $(\to I), 5$
7. $\Gamma \vdash \lambda p.\lambda q.\lambda r.r : \mathsf{Colist}\ C \subseteq X \to C \subseteq_{f\ x} X \to C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle$ $(\to I), 6$
8. $\Gamma \vdash \lambda p.\lambda q.\lambda r.r : \forall X\big(C \subseteq X \to C \subseteq_{f\ x} X \to C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle\big)$ $(\forall^2 I), 7$

$$\therefore\ s_1 = \lambda p.\lambda q.\lambda r.r$$

- $\Gamma \vdash s_2 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to C \subseteq_{f\ x} X \to C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big)$

1. $\Gamma,\ \Delta \vdash \overline{\mathsf{nothing}} : \mathsf{Mb}\ X(\mathsf{nothing})$ $(Hip)$
2. $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{single}}} \overline{\mathsf{nothing}} : \mathsf{Mb}\ X(\mathsf{tail}(\mathsf{f}\ \ y))$ $(Eq), 1$
3. $\Gamma,\ \Omega \vdash \lambda r.\overline{\mathsf{nothing}} : C(y) \to \mathsf{Mb}\ X(\mathsf{tail}(\mathsf{f}\ \ y))$ $(\to I), 2$
4. $\Gamma,\ \Omega \vdash \lambda r.\overline{\mathsf{nothing}} : \forall y.C(y) \to \mathsf{Mb}\ X(\mathsf{tail}(\mathsf{f}\ \ y))$ $(\forall I), 3$
5. $\Gamma,\ \Omega \vdash \lambda r.\overline{\mathsf{nothing}} : C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle$ $(Def\ \mathcal{K} \subseteq_f\ \ C), 4$

6. $\Gamma,\ p : \mathsf{Colist}\ C \subseteq X \vdash \lambda q.\lambda r.\overline{\mathsf{nothing}} : C \subseteq_{f\ x} X \to C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle$ $(\to I), 5$
7. $\Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{nothing}} : \mathsf{Colist}\ C \subseteq X \to C \subseteq_{f\ x} X \to C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle$ $(\to I), 6$
8. $\Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{nothing}} : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big)$ $(\forall^2 I), 7$

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{nothing}}$$

- 

1. $\Gamma \vdash s_1 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle C, \mathsf{head}\rangle\big)$ $(Hip)$
2. $\Gamma \vdash s_2 : \forall X\big(\mathsf{Colist}\ C \subseteq X \to \mathsf{Colist}\ C \subseteq_{f\ x} X \to \mathsf{Colist}\ C \subseteq_{f\ x} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big)$ $(Hip)$

| | | |
|---|---|---|
| 3. | $\Gamma \vdash x : \mathsf{Colist}\ C(x)$ | $(Var)$ |
| 4. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ x : \mathsf{Colist}\ C(\mathsf{f}\ x)$ | $(\nu I), 1, 2, 3$ |
| 5. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ x : \mathsf{Colist}\ C(\mathsf{single}\ x)$ | $(Def\ \mathsf{f}), 4$ |
| 6. | $\vdash \lambda x.\, \mathsf{MCoRec}\, s_1\ s_2\ x : C(x) \to \mathsf{Colist}\ C(\mathsf{single}\ x)$ | $(\to I), 5$ |
| 7. | $\vdash \lambda x.\, \mathsf{MCoRec}\, s_1\ s_2\ x : \forall x.C(x) \to \mathsf{Colist}\ C(\mathsf{single}\ x)$ | $(\forall I), 6$ |

$\therefore \vdash \overline{\mathsf{single}} : \forall x.C(x) \to \mathsf{Colist}\ C(\mathsf{single}\ x)$, donde $\overline{\mathsf{single}} = \lambda x.\, \mathsf{MCoRec}\, s_1\ s_2\ x$.

$\square$

- **Sfrom**. Sean $\mathsf{sfrom}$ un símbolo de función tal que $\mathbb{E}_{\mathsf{sfrom}} = \{\mathsf{head}(\mathsf{sfrom}\ x\ y) = x,\ \mathsf{tail}(\mathsf{sfrom}\ x\ y) = \mathsf{just}(\mathsf{sfrom}((\mathsf{sum}\ x\ \ \mathsf{suc}\ \mathsf{zero})\ y))\}$. Entonces

$$\vdash \overline{\mathsf{sfrom}} : \forall x.\forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathsf{Colist}\ \mathbb{N}(\mathsf{sfrom}\ x\ y)$$

donde $\overline{\mathsf{sfrom}} = \lambda x.\lambda y.\, \mathsf{MCoIt}\ s_1\ s_2\ x$, $s_1 = \lambda p.\lambda q.q$ y $s_2 = \lambda p.\lambda q.\overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}})))$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f}\ x = \mathsf{sfrom}\ x\ y$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{sfrom}} = \{\mathsf{head}(\mathsf{f}\ x) = x,\ \mathsf{tail}(\mathsf{f}\ x) = \mathsf{just}(\mathsf{f}(\mathsf{sum}\ x\ \ \mathsf{suc}\ \mathsf{zero}))\}$. Sea $\Gamma := \{x : \mathbb{N}(x),\ y : \mathbb{N}(y)\}$, $\Omega := \{p : \mathbb{N} \subseteq_{f\ y} X\}$ y $\Delta := \Omega \cup \{q : \mathbb{N}(r)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_{f\ y} \langle \mathbb{N}, \mathsf{head}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash q : \mathbb{N}(r)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{sfrom}}} q : \mathbb{N}(\mathsf{head}(\mathsf{f}\ r\ y))$ | $(Eq), 1$ |
| 3. | $\Gamma,\ \Omega \vdash \lambda q.q : \mathbb{N}(r) \to \mathbb{N}(\mathsf{head}(\mathsf{f}\ r\ y))$ | $(\to I), 2$ |
| 4. | $\Gamma,\ \Omega \vdash \lambda q.q : \forall r.\mathbb{N}(r) \to \mathbb{N}(\mathsf{head}(\mathsf{f}\ r\ y))$ | $(\forall I), 3$ |
| 5. | $\Gamma,\ \Omega \vdash \lambda q.q : \mathbb{N} \subseteq_{f\ y} \langle \mathbb{N}, \mathsf{head}\rangle$ | $(Def\ \mathcal{K} \subseteq_f\ C), 4$ |
| 6. | $\Gamma \vdash \lambda p.\lambda q.q : \mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_{f\ y} \langle \mathbb{N}, \mathsf{head}\rangle$ | $(\to I), 5$ |
| 7. | $\Gamma \vdash \lambda p.\lambda q.q : \forall X\big(\mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_{f\ y} \langle \mathbb{N}, \mathsf{head}\rangle\big)$ | $(\forall^2 I), 6$ |

$$\therefore\ s_1 = \lambda p.\lambda q.q$$

- $\Gamma \vdash s_2 : \forall X\big(\mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_{f\ y} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash \overline{\mathsf{zero}} : \mathbb{N}(\mathsf{zero})$ | $(Hip)$ |
| 2. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \forall x.\mathbb{N}(x) \to \mathbb{N}(\mathsf{suc}\ x)$ | $(Hip)$ |
| 3. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suc}} : \mathbb{N}(\mathsf{zero}) \to \mathbb{N}(\mathsf{suc}\ \mathsf{zero})$ | $(\forall E), 2, [x := \mathsf{zero}]$ |
| 4. | $\Gamma,\ \Delta \vdash \overline{\mathsf{suczero}} : \mathbb{N}(\mathsf{suc}\ \mathsf{zero})$ | $(\to E), 1, 3$ |
| 5. | $\Gamma,\ \Delta \vdash q : \mathbb{N}(r)$ | $(Var)$ |
| 6. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \forall n.\forall m.\mathbb{N}(n) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}(n\ m))$ | $(Hip)$ |

| | | |
|---|---|---|
| 7. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \forall m.\mathbb{N}(r) \to \mathbb{N}(m) \to \mathbb{N}(\mathsf{sum}(r\ m))$ | $(\forall E), 6, [n := r]$ |
| 8. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}} : \mathbb{N}(r) \to \mathbb{N}(\mathsf{suc}\ \mathsf{zero}) \to \mathbb{N}(\mathsf{sum}(r\ \ \mathsf{suc}\ \mathsf{zero}))$ | $(\forall E), 7, [m := \mathsf{suc}\ \mathsf{zero}]$ |
| 9. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}}q : \mathbb{N}(\mathsf{suc}\ \mathsf{zero}) \to \mathbb{N}(\mathsf{sum}(r\ \ \mathsf{suc}\ \mathsf{zero}))$ | $(\to E), 5, 8$ |
| 10. | $\Gamma,\ \Delta \vdash \overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}) : \mathbb{N}(\mathsf{sum}(r\ \ \mathsf{suc}\ \mathsf{zero}))$ | $(\to E), 4, 9$ |

$$\begin{array}{lll} 11. & \Gamma,\ \Delta \vdash p : \mathbb{N} \subseteq_{f\ y} X & (Var) \\ 12. & \Gamma,\ \Delta \vdash p : \forall x.\mathbb{N}(x) \to X(\mathsf{f}\ x) & (Def\ \mathcal{K} \subseteq_f\ C), 11 \end{array}$$

$$\begin{array}{lll} 13. & \Gamma,\ \Delta \vdash p : \mathbb{N}(\mathsf{sum}(r\ \ \mathsf{suc\ zero})) \to X(\mathsf{f}(\mathsf{sum}(r\ \ \mathsf{suc\ zero}))) & (\forall E), 12, [x := \mathsf{sum}(r\ \ \mathsf{suc\ zero})] \end{array}$$

$$\begin{array}{lll} 14. & \Gamma,\ \Delta \vdash p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}})) : X(\mathsf{f}(\mathsf{sum}(r\ \ \mathsf{suc\ zero}))) & (\to E), 10, 13 \\ 15. & \Gamma,\ \Delta \vdash \overline{\mathsf{just}} : \forall x.X(x) \to \mathsf{Mb}\ X(\mathsf{just}\ x) & (Hip) \end{array}$$

$$\begin{array}{ll} 16. & \Gamma,\ \Delta \vdash \overline{\mathsf{just}} : X(\mathsf{f}(\mathsf{sum}(r\ \ \mathsf{suc\ zero}))) \to \mathsf{Mb}\ X(\mathsf{just}(\mathsf{f}(\mathsf{sum}(r\ \ \mathsf{suc\ zero})))) \\ & \hfill (\forall E), 15, [x := \mathsf{f}(\mathsf{sum}(r\ \ \mathsf{suc\ zero}))] \end{array}$$

$$\begin{array}{lll} 17. & \Gamma,\ \Delta \vdash \overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}))) : \mathsf{Mb}\ X(\mathsf{just}(\mathsf{f}(\mathsf{sum}(r\ \ \mathsf{suc\ zero})))) & (\to E), 14, 16 \\ 18. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{sfrom}}} \overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}))) : \mathsf{Mb}\ X(\mathsf{tail}(\mathsf{f}\ \ r)) & (Eq), 17 \\ 19. & \Gamma,\ \Omega \vdash \lambda q.\overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}))) : \mathbb{N}(r) \to \mathsf{Mb}\ X(\mathsf{tail}(\mathsf{f}\ \ r)) & (\to I), 18 \\ 20. & \Gamma,\ \Omega \vdash \lambda q.\overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}))) : \forall r.\mathbb{N}(r) \to \mathsf{Mb}\ X(\mathsf{tail}(\mathsf{f}\ \ r)) & (\forall I), 19 \\ 21. & \Gamma,\ \Omega \vdash \lambda q.\overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}))) : \mathbb{N} \subseteq_{f\ y} \langle X, \mathsf{tail}\rangle & (Def\ \mathcal{K} \subseteq_f\ C), 20 \end{array}$$

$$\begin{array}{lll} 22. & \Gamma \vdash \lambda p.\lambda q.\overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}))) : \mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_{f\ y} \langle X, \mathsf{tail}\rangle & (\to I), 21 \\ 23. & \Gamma \vdash \lambda p.\lambda q.\overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}}))) : \forall X\big(\mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_{f\ y} \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big) & (\forall^2 I), 22 \end{array}$$

$$\therefore\ s_2 = \lambda p.\lambda q.\overline{\mathsf{just}}(p(\overline{\mathsf{sum}}q(\overline{\mathsf{suczero}})))$$

- 

$$\begin{array}{lll} 1. & \Gamma \vdash s_1 : \forall X\big(\mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_f \langle \mathbb{N}, \mathsf{head}\rangle\big) & (Hip) \\ 2. & \Gamma \vdash s_2 : \forall X\big(\mathbb{N} \subseteq_{f\ y} X \to \mathbb{N} \subseteq_f \langle \mathsf{Mb}\ X, \mathsf{tail}\rangle\big) & (Hip) \\ 3. & \Gamma \vdash x : \mathbb{N}(x) & (Var) \\ 4. & \Gamma \vdash \mathsf{MCoIt}\, s_1\ s_2\ x : \mathsf{Colist}\ \mathbb{N}(\mathsf{f}\ \ x) & (\nu I^-), 1, 2, 3 \\ 5. & \Gamma \vdash \mathsf{MCoIt}\, s_1\ s_2\ x : \mathsf{Colist}\ \mathbb{N}(\mathsf{sfrom}\ \ x\ y) & (Def\ \mathsf{f}), 4 \\ 6. & x : \mathbb{N}(x) \vdash \lambda y.\, \mathsf{MCoIt}\, s_1\ s_2\ x : \mathbb{N}(y) \to \mathsf{Colist}\ \mathbb{N}(\mathsf{sfrom}\ \ x\ y) & (\to I), 5 \\ 7. & \vdash \lambda x.\lambda y.\, \mathsf{MCoIt}\, s_1\ s_2\ x : \mathbb{N}(x) \to \mathbb{N}(y) \to \mathsf{Colist}\ \mathbb{N}(\mathsf{sfrom}\ \ x\ y) & (\to I), 6 \\ 8. & \vdash \lambda x.\lambda y.\, \mathsf{MCoIt}\, s_1\ s_2\ x : \forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathsf{Colist}\ \mathbb{N}(\mathsf{sfrom}\ \ x\ y) & (\forall I), 7 \\ 9. & \vdash \lambda x.\lambda y.\, \mathsf{MCoIt}\, s_1\ s_2\ x : \forall x.\forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathsf{Colist}\ \mathbb{N}(\mathsf{sfrom}\ \ x\ y) & (\forall I), 8 \end{array}$$

$\therefore \vdash \overline{\mathsf{sfrom}} : \forall x.\forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathsf{Colist}\ \mathbb{N}(\mathsf{sfrom}\ \ x\ y)$, donde $\overline{\mathsf{sfrom}} = \lambda x.\lambda y.\, \mathsf{MCoIt}\, s_1\ s_2\ x$.

□

- **From**. Sea from un símbolo de función. La función de los números que pertencen a un intervalo de números naturales se define como

$$\vdash \overline{\mathsf{from}} : \forall x.\forall y.\mathbb{N}(x) \to \mathbb{N}(y) \to \mathsf{Mb\ Colist}\ \mathbb{N}(\mathsf{from}\ \ x\ y)$$

donde

$$\begin{array}{l} \mathsf{from}\ x\ x\ =\ \mathsf{just}(\mathsf{single}\ n) \\ \mathsf{from}\ n\ m\ = \\ if\ n < m\ then\ \mathsf{just}(\mathsf{sfrom}\ n\ m) \\ else\ \mathsf{nothing} \end{array}$$

### 4.2.3. Producto

- **Definición**. Dados $\mathsf{P}$, $\mathsf{Q}$ conjuntos se definen

$$\mathsf{P}^{fst} = \{x \in X | fst(x) \in \mathsf{P}\} \text{ y } \mathsf{Q}^{snd} = \{x \in X | snd(x) \in \mathsf{Q}\}.$$

  El conjunto del producto de dos conjuntos $\mathsf{P}$,$\mathsf{Q}$ se define como:

$$\mathsf{P} \times \mathsf{Q} =_{def}= \nu(\Phi) \quad \text{con} \quad \Phi(X) = \mathsf{P}^{fst} \cap \mathsf{Q}^{snd}$$

- **Especificación funcional**

$$codata\ P\ *\ Q =\ Fst\ P\ |\ Snd\ Q$$

- **Implementación**. Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle \mathsf{P}, \mathsf{fst} \rangle$, $\mathcal{C}_2 = \langle \mathsf{Q}, \mathsf{snd} \rangle$. Se define el producto de dos conjuntos $\mathsf{P}$, $\mathsf{Q}$ como

$$\mathsf{P} \times \mathsf{Q} =_{def} \nu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2)$$

- **Reglas de construcción**

  Sea $t = x\ y$

$$\frac{\Gamma \vdash r : \mathsf{P} \times \mathsf{Q}(t)}{\Gamma \vdash \mathsf{out}_{2,1}\, r : \mathsf{P}(\mathsf{fst}(t))}\ (\nu E)$$

$$\frac{\Gamma \vdash r : \mathsf{P} \times \mathsf{Q}(t)}{\Gamma \vdash \mathsf{out}_{2,2}\, r : \mathsf{Q}(\mathsf{snd}(t))}\ (\nu E)$$

- **Destructores**

  - $\overline{\mathsf{fst}} = \lambda r.\, \mathsf{out}_{2,1}\, r$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{fst}} : \forall t.\, \mathsf{P} \times \mathsf{Q}(t) \to \mathsf{P}(\mathsf{fst}\ \ t)$

$$\begin{array}{lll} 1. & r : \mathsf{P} \times \mathsf{Q}(t) \vdash r : \mathsf{P} \times \mathsf{Q}(t) & (Var) \\ 2. & r : \mathsf{P} \times \mathsf{Q}(t) \vdash \mathsf{out}_{2,1}\, r : \mathsf{P}(\mathsf{fst}\ \ t) & (\nu E), 1 \\ 3. & \vdash \lambda r.\, \mathsf{out}_{2,1}\, r : \mathsf{P} \times \mathsf{Q}(t) \to \mathsf{P}(\mathsf{fst}\ \ t) & (\to I), 2 \\ 4. & \vdash \lambda r.\, \mathsf{out}_{2,1}\, r : \forall t.\, \mathsf{P} \times \mathsf{Q}(t) \to \mathsf{P}(\mathsf{fst}\ \ t) & (\forall I), 3 \end{array}$$

$$\therefore\ \overline{\mathsf{fst}} = \lambda r.\, \mathsf{out}_{2,1}\, r$$

□

- $\overline{\mathsf{snd}} = \lambda r.\, \mathsf{out}_{2,2}\, r$

  *Demostración.* Hay que probar que $\vdash \overline{\mathsf{snd}} : \forall t.\, \mathsf{P} \times \mathsf{Q}(t) \to \mathsf{Q}(\mathsf{snd}\ \ t)$

$$\begin{array}{lll} 1. & r : \mathsf{P} \times \mathsf{Q}(t) \vdash r : \mathsf{P} \times \mathsf{Q}(t) & (Var) \\ 2. & r : \mathsf{P} \times \mathsf{Q}(t) \vdash \mathsf{out}_{2,2}\, r : \mathsf{Q}(\mathsf{snd}\ \ t) & (\nu E), 1 \\ 3. & \vdash \lambda r.\, \mathsf{out}_{2,2}\, r : \mathsf{P} \times \mathsf{Q}(t) \to \mathsf{Q}(\mathsf{snd}\ \ t) & (\to I), 2 \\ 4. & \vdash \lambda r.\, \mathsf{out}_{2,2}\, r : \forall t.\, \mathsf{P} \times \mathsf{Q}(t) \to \mathsf{Q}(\mathsf{snd}\ \ t) & (\forall I), 3 \end{array}$$

$$\therefore\ \overline{\mathsf{snd}} = \lambda r.\, \mathsf{out}_{2,2}\, r$$

□

- **Semántica operacional**

$$\begin{array}{rcl} \mathsf{fst}(\mathsf{MColt}\ \ \vec{s}\, x) & \mapsto_\beta & s_1\ \ \mathsf{MColt}\ \ x \\ \mathsf{snd}(\mathsf{MColt}\ \ \vec{s}\, x) & \mapsto_\beta & s_2\ \ \mathsf{MColt}\ \ x \end{array}$$

Algunos ejemplos de extracción de programas son:

- **Constructor.** Sean $\mathsf{pair}$ un símbolo de función y $\mathbb{E}_{\mathsf{pair}} = \{\mathsf{fst}\ (\mathsf{pair}\ \ x\ y) = x,\ \mathsf{snd}\ (\mathsf{pair}\ \ x\ y) = y\}$. El constructor de un producto $\mathsf{pair}$ se define como

$$\vdash \overline{\mathsf{pair}} : \forall x.\forall y.\, \mathsf{P}(x) \to \mathsf{Q}(y) \to \mathsf{P} \times \mathsf{Q}(\mathsf{pair}\ \ x\ y)$$

donde $\overline{\mathsf{pair}} = \lambda x.\lambda y.\, \mathsf{MColt}\ \ s_1\ s_2\ y$, $s_1 = \lambda q.\lambda r.x$ y $s_2 = \lambda q.\lambda r.r$.

*Demostración.* Sea $\mathsf{f}$ tal que $\mathsf{f}\ \ y = \mathsf{pair}\ \ x\ y$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{pair}} = \{\mathsf{fst}\ (\mathsf{f}\ \ y) = x,\ \mathsf{snd}\ (\mathsf{f}\ \ y) = y\}$. Sea $\Gamma := \{x : \mathsf{P}(x), y : \mathsf{Q}(y)\}$, $\Omega := \{q : \mathsf{Q} \subseteq_f X\}$ y $\Delta := \Omega \cup \{r : \mathsf{Q}(r)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{P}, \mathsf{fst} \rangle\big)$

$$\begin{array}{lll} 1. & \Gamma,\ \Delta \vdash x : \mathsf{P}(x) & (Var) \\ 2. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{pair}}} x : \mathsf{P}(\mathsf{fst}(\mathsf{f}\ \ r)) & (Eq), 1 \\ 3. & \Gamma,\ \Omega \vdash \lambda r.x : \mathsf{Q}(r) \to \mathsf{P}(\mathsf{fst}(\mathsf{f}\ \ r)) & (\to I), 2 \\ 4. & \Gamma,\ \Omega \vdash \lambda r.x : \forall r.\, \mathsf{Q}(r) \to \mathsf{P}(\mathsf{fst}(\mathsf{f}\ \ r)) & (\forall I), 3 \\ 5. & \Gamma,\ \Omega \vdash \lambda r.x : \mathsf{Q} \subseteq_f \langle \mathsf{P}, \mathsf{fst} \rangle & (Def\ \mathcal{K} \subseteq_f \mathcal{C}), 4 \\ 6. & \Gamma \vdash \lambda q.\lambda r.x : \mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{P}, \mathsf{fst} \rangle & (\to I), 5 \\ 7. & \Gamma \vdash \lambda q.\lambda r.x : \forall X\big(\mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{P}, \mathsf{fst} \rangle\big) & (\forall^2 I), 6 \end{array}$$

$$\therefore\ s_1 = \lambda q.\lambda r.x$$

- $\Gamma \vdash s_2 : \forall X\big(\, \mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{Q}, \mathsf{snd}\rangle\big)$

$$
\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash r : \mathsf{Q}(r) & (Var) \\
2. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{pair}}} r : \mathsf{Q}(\mathsf{snd}(\mathsf{f}\ \ r)) & (Eq), 1 \\
3. & \Gamma,\ \Omega \vdash \lambda r.r : \mathsf{Q}(r) \to \mathsf{Q}(\mathsf{snd}(\mathsf{f}\ \ r)) & (\to I), 2 \\
4. & \Gamma,\ \Omega \vdash \lambda r.r : \forall r.\, \mathsf{Q}(r) \to \mathsf{Q}(\mathsf{snd}(\mathsf{f}\ \ r)) & (\forall I), 3 \\
5. & \Gamma,\ \Omega \vdash \lambda r.x : \mathsf{Q} \subseteq_f \langle \mathsf{Q}, \mathsf{snd}\rangle & (Def\ \mathcal{K} \subseteq_f \mathcal{C}), 4 \\
6. & \Gamma \vdash \lambda q.\lambda r.r : \mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{Q}, \mathsf{snd}\rangle & (\to I), 5 \\
7. & \Gamma \vdash \lambda q.\lambda r.r : \forall X\big(\, \mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{Q}, \mathsf{snd}\rangle\big) & (\forall^2 I), 6
\end{array}
$$

$$\therefore\ \ s_2 = \lambda q.\lambda r.r$$

-

$$
\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(\, \mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{P}, \mathsf{fst}\rangle\big) & (Hip) \\
2. & \Gamma \vdash s_2 : \forall X\big(\, \mathsf{Q} \subseteq_f X \to \mathsf{Q} \subseteq_f \langle \mathsf{Q}, \mathsf{snd}\rangle\big) & (Hip) \\
3. & \Gamma \vdash y : \mathsf{Q}(y) & (Var) \\
4. & \Gamma \vdash \mathsf{MColt}\, s_1\ s_2\ y : \mathsf{P} \times \mathsf{Q}(\mathsf{f}\ \ y) & (\nu I^-), 1, 2, 3 \\
5. & \Gamma \vdash \mathsf{MColt}\, s_1\ s_2\ y : \mathsf{P} \times \mathsf{Q}(\mathsf{pair}\ \ x\ y) & (Def\ \ \mathsf{f}), 4 \\
6. & x : \mathsf{P}(x) \vdash \lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y : \mathsf{Q}(y) \to \mathsf{P} \times \mathsf{Q}(\mathsf{pair}\ \ x\ y) & (\to I), 5 \\
7. & \vdash \lambda x.\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y : \mathsf{P}(x) \to \mathsf{Q}(y) \to \mathsf{P} \times \mathsf{Q}(\mathsf{pair}\ \ x\ y) & (\to I), 6 \\
8. & \vdash \lambda x.\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y : \forall y.\, \mathsf{P}(x) \to \mathsf{Q}(y) \to \mathsf{P} \times \mathsf{Q}(\mathsf{pair}\ \ x\ y) & (\forall I), 7 \\
9. & \vdash \lambda x.\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y : \forall x.\forall y.\, \mathsf{P}(x) \to \mathsf{Q}(y) \to \mathsf{P} \times \mathsf{Q}(\mathsf{pair}\ \ x\ y) & (\forall I), 8
\end{array}
$$

$$\therefore \vdash \overline{\mathsf{pair}} : \forall x.\forall y.\, \mathsf{P}(x) \to \mathsf{Q}(y) \to \mathsf{P} \times \mathsf{Q}(\mathsf{pair}\ \ x\ y),\ \ \text{donde} \quad \overline{\mathsf{pair}} = \lambda x.\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y.$$

$\square$

- **Semántica operacional de la función $\overline{\mathsf{pair}}$**

  - $\mathsf{fst}\ (\mathsf{pair}\ \ x\ y) \ \ = \ \ x.$

$$
\begin{array}{llll}
\mathsf{fst}\ (\mathsf{pair}\ \ x\ y) & \to & \mathsf{fst}\ ((\lambda x.\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y)\ x\ y) & \to \\
\mathsf{fst}\ ((\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y)\ [x := x]\ y) & \to & \mathsf{fst}\ ((\lambda y.\, \mathsf{MColt}\, s_1\ s_2\ y)\ y) & \to \\
\mathsf{fst}\ (\mathsf{MColt}\, s_1\ s_2\ y\ [y := y]) & \to & \mathsf{fst}\ (\mathsf{MColt}\, s_1\ s_2\ y) & \to \\
(\lambda r.\, \mathsf{out}_{2,1}\, r)\ (\mathsf{MColt}\, s_1\ s_2\ y) & \to & (\mathsf{out}_{2,1}\, r)\ [r := \mathsf{MColt}\, s_1\ s_2\ y] & \to \\
\mathsf{out}_{2,1}\ (\mathsf{MColt}\, s_1\ s_2\ y) & \mapsto_\beta & s_1\ (\mathsf{MColt}\, s_1\ s_2)\ y & \to \\
(\lambda q.\lambda r.x)\ (\mathsf{MColt}\, s_1\ s_2)\ y & \to & (\lambda r.x)\ [q := \mathsf{MColt}\, s_1\ s_2]\ y & \to \\
(\lambda r.x)\ y & \to & x\ [r := y] & \to \\
x & & &
\end{array}
$$

$$\therefore\ \mathsf{fst}\ (\mathsf{pair}\ \ x\ y) \ \ = \ \ x$$

  - $\mathsf{snd}\ (\mathsf{pair}\ \ x\ y) \ \ = \ \ y.$

$$
\begin{array}{lcll}
\mathsf{snd}\ (\mathsf{pair}\ \ x\ y) & \to & \mathsf{snd}\ ((\lambda x.\lambda y.\,\mathsf{MColt}\, s_1\ s_2\ y)\ x\ y) & \to \\
\mathsf{snd}\ ((\lambda y.\,\mathsf{MColt}\, s_1\ s_2\ y)\ [x := x]\ y) & \to & \mathsf{snd}\ ((\lambda y.\,\mathsf{MColt}\, s_1\ s_2\ y)\ y) & \to \\
\mathsf{snd}\ (\mathsf{MColt}\, s_1\ s_2\ y\ [y := y]) & \to & \mathsf{snd}\ (\mathsf{MColt}\, s_1\ s_2\ y) & \to \\
(\lambda r.\,\mathsf{out}_{2,2}\, r)\ (\mathsf{MColt}\, s_1\ s_2\ y) & \to & (\mathsf{out}_{2,2}\, r)\ [r := \mathsf{MColt}\, s_1\ s_2\ y] & \to \\
\mathsf{out}_{2,2}\ (\mathsf{MColt}\, s_1\ s_2\ y) & \mapsto_\beta & s_2\ (\mathsf{MColt}\, s_1\ s_2)\ y & \to \\
(\lambda q.\lambda r.r)\ (\mathsf{MColt}\, s_1\ s_2)\ y & \to & (\lambda r.r)\ [q := \mathsf{MColt}\, s_1\ s_2]\ y & \to \\
(\lambda r.r)\ y & \to & r\ [r := y] & \to \\
y & & &
\end{array}
$$

$$\therefore\ \mathsf{snd}\ (\mathsf{pair}\ \ x\ y)\ \ =\ \ y$$

### 4.2.4. Árboles binarios completos con etiqueta en A estrictamente infinitos

- **Definición**. Sean $A^{label} = \{x \in X | label(x) \in A\}$, $X^{lsbt} = \{x \in X | lsbt(x) \in X\}$ y $X^{rsbt} = \{x \in X | rsbt(x) \in X\}$. El conjunto de los árboles binarios completos con etiqueta en A estrictamente infinitos se define como:

$$\mathsf{InfbT\ A} =_{def} \nu(\Phi) \quad \text{con} \quad \Phi(X) = A^{label} \cap X^{lsbt} \cap X^{rsbt}$$

- **Especificación funcional**

$$codata\ InfbTree\ a = \ Label\ a\ |\ Lsbt\ (InfbTree\ a)\ |\ Rsbt\ (InfbTree\ a)$$

- **Implementación**.Considerese las siguientes cláusulas tales que $\mathcal{C}_1 = \langle \mathsf{A}, \mathsf{label}\rangle$, $\mathcal{C}_2 = \langle X, \mathsf{lsbt}\rangle$, $\mathcal{C}_3 = \langle X, \mathsf{rsbt}\rangle$. Los árboles binarios completos con etiqueta en A estrictamente infinitos se definen como:

$$\mathsf{InfbT\ A} =_{def} \nu(\Phi) \quad \text{donde} \quad \Phi = \lambda X.(\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3)$$

- **Reglas de construcción**

$$\frac{\Gamma \vdash r : \mathsf{InfbT\ A}(t)}{\Gamma \vdash \mathsf{out}_{3,1}\, r : \mathsf{A}(\mathsf{label}(t))}\ (\nu E)$$

$$\frac{\Gamma \vdash r : \mathsf{InfbT\ A}(t)}{\Gamma \vdash \mathsf{out}_{3,2}\, r : \mathsf{InfbT\ A}(\mathsf{lsbt}(t))}\ (\nu E)$$

$$\frac{\Gamma \vdash r : \mathsf{InfbT\ A}(t)}{\Gamma \vdash \mathsf{out}_{3,3}\, r : \mathsf{InfbT\ A}(\mathsf{rsbt}(t))}\ (\nu E)$$

- **Destructores**

  - $\overline{\mathsf{label}} = \lambda x.\,\mathsf{out}_{3,1}\,x$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{label}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{A}(\mathsf{label}\ \ x)$

$$\begin{array}{lll} 1. & x : \mathsf{InfbT\ A}(x) \vdash x : \mathsf{InfbT\ A}(x) & (Var) \\ 2. & x : \mathsf{InfbT\ A}(x) \vdash \mathsf{out}_{3,1}\,x : \mathsf{A}(\mathsf{label}\ \ x) & (\nu E), 1 \\ 3. & \vdash \lambda x.\,\mathsf{out}_{3,1}\,x : \mathsf{InfbT\ A}(x) \to \mathsf{A}(\mathsf{label}\ \ x) & (\to I), 2 \\ 4. & \vdash \lambda x.\,\mathsf{out}_{3,1}\,x : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{A}(\mathsf{label}\ \ x) & (\forall I), 3 \end{array}$$

$$\therefore\ \overline{\mathsf{label}} = \lambda x.\,\mathsf{out}_{3,1}\,x$$

□

  - $\overline{\mathsf{lsbt}} = \lambda x.\,\mathsf{out}_{3,2}\,x$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{lsbt}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ x)$

$$\begin{array}{lll} 1. & x : \mathsf{InfbT\ A}(x) \vdash x : \mathsf{InfbT\ A}(x) & (Var) \\ 2. & x : \mathsf{InfbT\ A}(x) \vdash \mathsf{out}_{3,2}\,x : \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ x) & (\nu E), 1 \\ 3. & \vdash \lambda x.\,\mathsf{out}_{3,2}\,x : \mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ x) & (\to I), 2 \\ 4. & \vdash \lambda x.\,\mathsf{out}_{3,2}\,x : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ x) & (\forall I), 3 \end{array}$$

$$\therefore\ \overline{\mathsf{lsbt}} = \lambda x.\,\mathsf{out}_{3,2}\,x$$

□

  - $\overline{\mathsf{rsbt}} = \lambda x.\,\mathsf{out}_{3,3}\,x$

    *Demostración.* Hay que probar que $\vdash \overline{\mathsf{rsbt}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ x)$

$$\begin{array}{lll} 1. & x : \mathsf{InfbT\ A}(x) \vdash x : \mathsf{InfbT\ A}(x) & (Var) \\ 2. & x : \mathsf{InfbT\ A}(x) \vdash \mathsf{out}_{3,3}\,x : \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ x) & (\nu E), 1 \\ 3. & \vdash \lambda x.\,\mathsf{out}_{3,3}\,x : \mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ x) & (\to I), 2 \\ 4. & \vdash \lambda x.\,\mathsf{out}_{3,3}\,x : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ x) & (\forall I), 3 \end{array}$$

$$\therefore\ \overline{\mathsf{rsbt}} = \lambda x.\,\mathsf{out}_{3,3}\,x$$

□

- **Semántica operacional**

$$\begin{array}{rcl} \mathsf{label}(\mathsf{MCoRec}\ \vec{s}\,x) & \mapsto_\beta & s_1\ \mathsf{MCoRec}\ x \\ \mathsf{lsbt}(\mathsf{MCoRec}\ \vec{s}\,x) & \mapsto_\beta & s_2\ \mathsf{MCoRec}\ x \\ \mathsf{rsbt}(\mathsf{MCoRec}\ \vec{s}\,x) & \mapsto_\beta & s_3\ \mathsf{MCoRec}\ x \end{array}$$

Algunos ejemplos de extracción de programas son:

- **Constructor**. Sean $\mathsf{mkibt}$ un símbolo de función y

$$\mathbb{E}_{\mathsf{mkibt}} = \{\mathsf{label}\ (\mathsf{mkibt}\ \ x\ y\ z) = x,\ \mathsf{lsbt}\ (\mathsf{mkibt}\ \ x\ y\ z) = y,\ \mathsf{rsbt}\ (\mathsf{mkibt}\ \ x\ y\ z) = z\}.$$

El constructor de los árboles binarios completos con etiqueta en A estrictamente infinitos se define como

$$\vdash \overline{\mathsf{mkibt}} : \forall x.\forall y.\forall z.\, \mathsf{A}(x) \to \mathsf{InfbT\ A}(y) \to \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ \ x\ y\ z)$$

donde $\vdash \overline{\mathsf{mkibt}} = \lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\ \ s_1\ \ s_2\ \ s_3\ \ z,\ s_1 = \lambda p.\lambda q.\lambda r.x,\ s_2 = \lambda p.\lambda q.\lambda r.py,\ s_3 = \lambda p.\lambda q.\lambda r.pr.$

*Demostración.* Sea $\mathsf{f} = \mathsf{mkibt}$ tal que $\mathsf{f}\ \ z = \mathsf{mkibt}\ \ x\ y\ z$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{mkibt}} = \{\mathsf{label}\ (\mathsf{f}\ \ x\ y\ z) = x,\ \mathsf{lsbt}\ (\mathsf{f}\ \ x\ y\ z) = y,\ \mathsf{rsbt}\ (\mathsf{f}\ \ x\ y\ z) = z\}$. Sea $\Gamma := \{x : \mathsf{A}(x), y : \mathsf{InfbT\ A}(y), z : \mathsf{InfbT\ A}(z)\}$, $\Omega := \{p : \mathsf{InfbT\ A} \subseteq X,\ q : \mathsf{InfbT\ A} \subseteq_f X\}$ y $\Delta := \Omega \cup \{r : \mathsf{A}(t)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash x : \mathsf{A}(x)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{mkibt}}} x : \mathsf{A}(\mathsf{label}(\mathsf{f}\ \ x\ y\ t))$ | $(Eq), 1$ |
| 3. | $\Gamma,\ \Omega \vdash \lambda r.x : \mathsf{A}(t) \to \mathsf{A}(\mathsf{label}(\mathsf{f}\ \ x\ y\ t))$ | $(\to I), 2$ |
| 4. | $\Gamma,\ \Omega \vdash \lambda r.x : \forall t.\, \mathsf{A}(t) \to \mathsf{A}(\mathsf{label}(\mathsf{f}\ \ x\ y\ t))$ | $(\forall I), 3$ |
| 5. | $\Gamma,\ \Omega \vdash \lambda r.x : \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle$ | $(Def\ \mathcal{K} \subseteq_f \mathcal{C}), 4$ |

| | | |
|---|---|---|
| 6. | $\Gamma,\ p : \mathsf{InfbT\ A} \subseteq X \vdash \lambda q.\lambda r.x : \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle$ | $(\to I), 5$ |
| 7. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.x : \mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle$ | $(\to I), 6$ |
| 8. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.x : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle\big)$ | $(\forall^2 I), 7$ |

$$\therefore\ s_1 = \lambda p.\lambda q.\lambda r.x$$

- $\Gamma \vdash s_2 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash y : \mathsf{InfbT\ A}(y)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A} \subseteq X$ | $(Var)$ |
| 3. | $\Gamma,\ \Delta \vdash p : \forall x.\mathsf{InfbT\ A}(x) \to X(x)$ | $(Def\ \mathcal{K} \subseteq_f \mathcal{C}), 2$ |
| 4. | $\Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A}(y) \to X(y)$ | $(\forall E), 3, [x := y]$ |
| 5. | $\Gamma,\ \Delta \vdash py : X(y)$ | $(\to E), 1, 4$ |
| 6. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{mkibt}}} py : X(\mathsf{lsbt}(\mathsf{f}\ \ x\ y\ t))$ | $(Eq), 5$ |
| 7. | $\Gamma,\ \Omega \vdash \lambda r.py : \mathsf{InfbT\ A}(t) \to X(\mathsf{lsbt}(\mathsf{f}\ \ x\ y\ t))$ | $(\to I), 6$ |
| 8. | $\Gamma,\ \Omega \vdash \lambda r.py : \forall t.\mathsf{InfbT\ A}(t) \to X(\mathsf{lsbt}(\mathsf{f}\ \ x\ y\ t))$ | $(\forall I), 7$ |
| 9. | $\Gamma,\ \Omega \vdash \lambda r.py : \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle$ | $(Def\ \mathcal{K} \subseteq_f \mathcal{C}), 8$ |

| | | |
|---|---|---|
| 10. | $\Gamma,\ p : \mathsf{InfbT\ A} \subseteq X \vdash \lambda q.\lambda r.py : \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle$ | $(\to I), 9$ |
| 11. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.py : \mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle$ | $(\to I), 10$ |
| 12. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.py : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle\big)$ | $(\forall^2 I), 11$ |

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.py$$

- $\Gamma \vdash s_3 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle\big)$

| | | |
|---|---|---|
| 1. | $\Gamma,\ \Delta \vdash r : \mathsf{InfbT\ A}(t)$ | $(Var)$ |
| 2. | $\Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A} \subseteq X$ | $(Var)$ |
| 3. | $\Gamma,\ \Delta \vdash p : \forall x.\mathsf{InfbT\ A}(x) \to X(x)$ | $(Def\ \mathcal{K} \subseteq_f \mathcal{C}), 2$ |
| 4. | $\Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A}(t) \to X(t)$ | $(\forall E), 3, [x := t]$ |
| 5. | $\Gamma,\ \Delta \vdash pr : X(t)$ | $(\to E), 1, 4$ |
| 6. | $\Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{mkibt}}} pr : X(\mathsf{rsbt}(\mathsf{f}\ x\ y\ t))$ | $(Eq), 5$ |
| 7. | $\Gamma,\ \Omega \vdash \lambda r.pr : \mathsf{InfbT\ A}(t) \to X(\mathsf{rsbt}(\mathsf{f}\ x\ y\ t))$ | $(\to I), 6$ |
| 8. | $\Gamma,\ \Omega \vdash \lambda r.pr : \forall t.\mathsf{InfbT\ A}(t) \to X(\mathsf{rsbt}(\mathsf{f}\ x\ y\ t))$ | $(\forall I), 7$ |
| 9. | $\Gamma,\ \Omega \vdash \lambda r.pr : \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle$ | $(Def\ \mathcal{K} \subseteq_f \mathcal{C}), 8$ |

| | | |
|---|---|---|
| 10. | $\Gamma,\ p : \mathsf{InfbT\ A} \subseteq X \vdash \lambda q.\lambda r.pr : \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle$ | $(\to I), 9$ |
| 11. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.pr : \mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle$ | $(\to I), 10$ |
| 12. | $\Gamma \vdash \lambda p.\lambda q.\lambda r.pr : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle\big)$ | $(\forall^2 I), 11$ |

$$\therefore\ s_3 = \lambda p.\lambda q.\lambda r.pr$$

- 

| | | |
|---|---|---|
| 1. | $\Gamma \vdash s_1 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle\big)$ | $(Hip)$ |
| 2. | $\Gamma \vdash s_2 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle\big)$ | $(Hip)$ |
| 3. | $\Gamma \vdash s_3 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle\big)$ | $(Hip)$ |
| 4. | $\Gamma \vdash z : \mathsf{InfbT\ A}(z)$ | $(Var)$ |
| 5. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \mathsf{InfbT\ A}(\mathsf{f}\ z)$ | $(\nu I), 1, 2, 3, 4$ |
| 6. | $\Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$ | $(Def\ \mathsf{f}), 5$ |

| | | |
|---|---|---|
| 7. | $x : \mathsf{A}(x), y : \mathsf{InfbT\ A}(y) \vdash \lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$ | $(\to I), 6$ |
| 8. | $x : \mathsf{A}(x) \vdash \lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \mathsf{InfbT\ A}(y) \to \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$ | $(\to I), 7$ |

| | | |
|---|---|---|
| 9. | $\vdash \lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \mathsf{A}(x) \to \mathsf{InfbT\ A}(y) \to \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$ | $(\to I), 8$ |
| 10. | $\vdash \lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \forall z.\, \mathsf{A}(x) \to \mathsf{InfbT\ A}(y) \to \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$ | $(\forall I), 9$ |
| 11. | $\vdash \lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \forall y.\forall z.\, \mathsf{A}(x) \to \mathsf{InfbT\ A}(y) \to \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$ | $(\forall I), 10$ |

| | | |
|---|---|---|
| 12. | $\vdash \lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z : \forall x.\forall y.\forall z.\, \mathsf{A}(x) \to \mathsf{InfbT\ A}(y) \to \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$ | $(\forall I), 11$ |

$\therefore\ \vdash \overline{\mathsf{mkibt}} : \forall x.\forall y.\forall z.\, \mathsf{A}(x) \to \mathsf{InfbT\ A}(y) \to \mathsf{InfbT\ A}(z) \to \mathsf{InfbT\ A}(\mathsf{mkibt}\ x\ y\ z)$, donde $\overline{\mathsf{mkibt}} = \lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z$

□

- **Semántica operacional de la función $\overline{\mathsf{mkibt}}$**

- $\overline{\mathsf{label}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) \ = \ x.$

$$
\begin{array}{ll}
\overline{\mathsf{label}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) & \to \\
\overline{\mathsf{label}}\ ((\lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ x\ y\ z) & \to \\
\overline{\mathsf{label}}\ ((\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ [x := x]\ y\ z) & \to \\
\overline{\mathsf{label}}\ ((\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ y\ z) & \to \\
\overline{\mathsf{label}}\ ((\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ [y := y]\ z) & \to \\
\overline{\mathsf{label}}\ ((\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ z) & \to \\
\overline{\mathsf{label}}\ ((\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ [z := z]) & \to \\
\overline{\mathsf{label}}\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z) & \to \\
(\lambda x.\, \mathsf{out}_{3,1}\, x)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z) & \to \\
(\mathsf{out}_{3,1}\, x)\ [x := \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z] & \to \\
\mathsf{out}_{3,1}\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z) & \mapsto_\beta \\
s_1\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda p.\lambda q.\lambda r.x)\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda q.\lambda r.x)\ [p := \lambda x.x]\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda q.\lambda r.x)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda r.x)\ [q := \mathsf{MCoRec}\, s_1\ s_2\ s_3\ ]\ z & \to \\
(\lambda r.x)\ z & \to \\
x\ [r := z] & \to \\
x &
\end{array}
$$

$$\therefore\ \overline{\mathsf{label}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) \ = \ x$$

- $\overline{\mathsf{lsbt}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) \ = \ y.$

$$
\begin{array}{ll}
\overline{\mathsf{lsbt}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) & \to \\
\overline{\mathsf{lsbt}}\ ((\lambda x.\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ x\ y\ z) & \to \\
\overline{\mathsf{lsbt}}\ ((\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ [x := x]\ y\ z) & \to \\
\overline{\mathsf{lsbt}}\ ((\lambda y.\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ y\ z) & \to \\
\overline{\mathsf{lsbt}}\ ((\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ [y := y]\ z) & \to \\
\overline{\mathsf{lsbt}}\ ((\lambda z.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ z) & \to \\
\overline{\mathsf{lsbt}}\ ((\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z)\ [z := z]) & \to \\
\overline{\mathsf{lsbt}}\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z) & \to \\
(\lambda x.\, \mathsf{out}_{3,2}\, x)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z) & \to \\
(\mathsf{out}_{3,2}\, x)\ [x := \mathsf{MCoRec}\, s_1\ s_2\ s_3\ z] & \to \\
\mathsf{out}_{3,2}\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ z) & \mapsto_\beta \\
s_2\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda p.\lambda q.\lambda r.py)\ (\lambda x.x)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda q.\lambda r.py)\ [p := \lambda x.x]\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda q.\lambda r.(\lambda x.x)\ y)\ (\mathsf{MCoRec}\, s_1\ s_2\ s_3\ )\ z & \to \\
(\lambda r.(\lambda x.x)\ y)\ [q := \mathsf{MCoRec}\, s_1\ s_2\ s_3\ ]\ z & \to \\
(\lambda r.(\lambda x.x)\ y)\ z & \to \\
((\lambda x.x)\ y)\ [r := z] & \to \\
(\lambda x.x)\ y & \to \\
x\ [x := y] & \to \\
y &
\end{array}
$$

$$\therefore\ \overline{\mathsf{lsbt}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) \ = \ y$$

- $\circ$ $\overline{\mathsf{rsbt}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) \ = \ z.$

$$\begin{array}{ll}
\overline{\mathsf{rsbt}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) & \rightarrow \\
\overline{\mathsf{rsbt}}\ ((\lambda x.\lambda y.\lambda z.\,\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z)\ x\ y\ z) & \rightarrow \\
\overline{\mathsf{rsbt}}\ ((\lambda y.\lambda z.\,\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z)\ [x := x]\ y\ z) & \rightarrow \\
\overline{\mathsf{rsbt}}\ ((\lambda y.\lambda z.\,\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z)\ y\ z) & \rightarrow \\
\overline{\mathsf{rsbt}}\ ((\lambda z.\,\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z)\ [y := y]\ z) & \rightarrow \\
\overline{\mathsf{rsbt}}\ ((\lambda z.\,\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z)\ z) & \rightarrow \\
\overline{\mathsf{rsbt}}\ ((\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z)\ [z := z]) & \rightarrow \\
\overline{\mathsf{rsbt}}\ (\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z) & \rightarrow \\
(\lambda x.\,\mathsf{out}_{3,3}\,x)\ (\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z) & \rightarrow \\
(\mathsf{out}_{3,3}\,x)\ [x := \mathsf{MCoRec}\,s_1\ s_2\ s_3\ z] & \rightarrow \\
\mathsf{out}_{3,3}\ (\mathsf{MCoRec}\,s_1\ s_2\ s_3\ z) & \mapsto_\beta \\
s_3\ (\lambda x.x)\ (\mathsf{MCoRec}\,s_1\ s_2\ s_3\ )\ z & \rightarrow \\
(\lambda p.\lambda q.\lambda r.pr)\ (\lambda x.x)\ (\mathsf{MCoRec}\,s_1\ s_2\ s_3\ )\ z & \rightarrow \\
(\lambda q.\lambda r.pr)\ [p := \lambda x.x]\ (\mathsf{MCoRec}\,s_1\ s_2\ s_3\ )\ z & \rightarrow \\
(\lambda q.\lambda r.(\lambda x.x)\ r)\ (\mathsf{MCoRec}\,s_1\ s_2\ s_3\ )\ z & \rightarrow \\
(\lambda r.(\lambda x.x)\ r)\ [q := \mathsf{MCoRec}\,s_1\ s_2\ s_3\ ]\ z & \rightarrow \\
(\lambda r.(\lambda x.x)\ r)\ z & \rightarrow \\
((\lambda x.x)\ r)\ [r := z] & \rightarrow \\
(\lambda x.x)\ z & \rightarrow \\
x\ [x := z] & \rightarrow \\
z &
\end{array}$$

$$\therefore\ \overline{\mathsf{rsbt}}\ (\overline{\mathsf{mkibt}}\ x\ y\ z) \ = \ z$$

- **Swapt**. Sean $\mathsf{swapt}$ un símbolo de función y $\mathbb{E}_{\mathsf{swapt}} = \{\mathsf{label}\ (\mathsf{swapt}\ \ t) = \mathsf{label}\ \ t,\ \mathsf{lsbt}\ (\mathsf{swapt}\ \ t) = \mathsf{rsbt}\ \ t,\ \mathsf{rsbt}\ (\mathsf{swapt}\ \ t) = \mathsf{lsbt}\ \ t\}$. La función que intercambia los sub-árboles izquierdo y derecho de un árbol se define como:

$$\vdash \overline{\mathsf{swapt}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{swapt}\ \ x)$$

donde $\overline{\mathsf{swapt}} = \lambda x.\,\mathsf{MCoRec}\ \ s_1\ \ s_2\ \ s_3\ \ x,\ s_1 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{label}}r,\ s_2 = \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{rsbt}}r),\ s_3 = \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{lsbt}}r)$.

*Demostración.* Sea $\mathsf{f} = \mathsf{swapt}$. Redefinimos las ecuaciones $\mathbb{E}_{\mathsf{swapt}} = \{\mathsf{label}(\mathsf{f}\ \ t) = \mathsf{label}\ \ t,\ \mathsf{lsbt}(\mathsf{f}\ \ t) = \mathsf{rsbt}\ \ t,\ \mathsf{rsbt}(\mathsf{f}\ \ t) = \mathsf{lsbt}\ \ t\}$. Sea $\Gamma := \{x : \mathsf{InfbT\ A}(x)\}$, $\Omega := \{p : \mathsf{InfbT\ A} \subseteq X,\ q : \mathsf{InfbT\ A} \subseteq_f X\}$ y $\Delta := \Omega \cup \{r : \mathsf{InfbT\ A}(t)\}$.

- $\Gamma \vdash s_1 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle\big)$

$$
\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash r : \mathsf{InfbT\ A}(t) & (Var)\\
2. & \Gamma,\ \Delta \vdash \overline{\mathsf{label}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{A}(\mathsf{label}\ \ x) & (Hip)\\
3. & \Gamma,\ \Delta \vdash \overline{\mathsf{label}} : \mathsf{InfbT\ A}(t) \to \mathsf{A}(\mathsf{label}\ \ t) & (\forall E), 2, [x := t]\\
4. & \Gamma,\ \Delta \vdash \overline{\mathsf{label}}r : \mathsf{A}(\mathsf{label}\ \ t) & (\forall E), 1, 3\\
5. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{swapt}}} \overline{\mathsf{label}}r : \mathsf{A}(\mathsf{label}(\mathsf{f}\ \ t)) & (Eq), 4\\
6. & \Gamma,\ \Omega \vdash \lambda r.\overline{\mathsf{label}}r : \mathsf{InfbT\ A}(t) \to \mathsf{A}(\mathsf{label}(\mathsf{f}\ \ t)) & (\to I), 5\\
7. & \Gamma,\ \Omega \vdash \lambda r.\overline{\mathsf{label}}r : \forall t.\mathsf{InfbT\ A}(t) \to \mathsf{A}(\mathsf{label}(\mathsf{f}\ \ t)) & (\forall I), 6\\
8. & \Gamma,\ \Omega \vdash \lambda r.\overline{\mathsf{label}}r : \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle & (Def\ \mathcal{K} \subseteq_f \mathcal{C}), 7
\end{array}
$$

$$
\begin{array}{lll}
9. & \Gamma,\ p : \mathsf{InfbT\ A} \subseteq X \vdash \lambda q.\lambda r.\overline{\mathsf{label}}r : \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle & (\to I), 8\\
10. & \Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{label}}r : \mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle & (\to I), 9\\
11. & \Gamma \vdash \lambda p.\lambda q.\lambda r.\overline{\mathsf{label}}r : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle\big) & (\forall^2 I), 10
\end{array}
$$

$$\therefore\ s_1 = \lambda p.\lambda q.\lambda r.\overline{\mathsf{label}}r$$

- $\Gamma \vdash s_2 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle\big)$

$$
\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash r : \mathsf{InfbT\ A}(t) & (Var)\\
2. & \Gamma,\ \Delta \vdash \overline{\mathsf{rsbt}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ x) & (Hip)\\
3. & \Gamma,\ \Delta \vdash \overline{\mathsf{rsbt}} : \mathsf{InfbT\ A}(t) \to \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ t) & (\forall E), 2, [x := t]\\
4. & \Gamma,\ \Delta \vdash \overline{\mathsf{rsbt}}r : \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ t) & (\forall E), 1, 3\\
5. & \Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A} \subseteq X & (Var)\\
6. & \Gamma,\ \Delta \vdash p : \forall x.\mathsf{InfbT\ A}(x) \to X(x) & (Def\ \mathcal{K} \subseteq_f \mathcal{C}), 5\\
7. & \Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A}(\mathsf{rsbt}\ \ t) \to X(\mathsf{rsbt}\ \ t) & (\forall E), 6, [x := \mathsf{rsbt}\ \ t]\\
8. & \Gamma,\ \Delta \vdash p(\overline{\mathsf{rsbt}}r) : X(\mathsf{rsbt}\ \ t) & (\to E), 4, 7\\
9. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{swapt}}} p(\overline{\mathsf{rsbt}}r) : X(\mathsf{lsbt}(\mathsf{f}\ \ t)) & (Eq), 8\\
10. & \Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{rsbt}}r) : \mathsf{InfbT\ A}(t) \to X(\mathsf{lsbt}(\mathsf{f}\ \ t)) & (\to I), 9\\
11. & \Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{rsbt}}r) : \forall t.\mathsf{InfbT\ A}(t) \to X(\mathsf{lsbt}(\mathsf{f}\ \ t)) & (\forall I), 10\\
12. & \Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{rsbt}}r) : \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle & (Def\ \mathcal{K} \subseteq_f \mathcal{C}), 11
\end{array}
$$

$$
\begin{array}{lll}
13. & \Gamma,\ p : \mathsf{InfbT\ A} \subseteq X \vdash \lambda q.\lambda r.p(\overline{\mathsf{rsbt}}r) : \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle & (\to I), 12\\
14. & \Gamma \vdash \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{rsbt}}r) : \mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle & (\to I), 13\\
15. & \Gamma \vdash \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{rsbt}}r) : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle\big) & (\forall^2 I), 14
\end{array}
$$

$$\therefore\ s_2 = \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{rsbt}}r)$$

- $\Gamma \vdash s_3 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle\big)$

$$
\begin{array}{lll}
1. & \Gamma,\ \Delta \vdash r : \mathsf{InfbT\ A}(t) & (Var)\\
2. & \Gamma,\ \Delta \vdash \overline{\mathsf{lsbt}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ x) & (Hip)\\
3. & \Gamma,\ \Delta \vdash \overline{\mathsf{lsbt}} : \mathsf{InfbT\ A}(t) \to \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ t) & (\forall E), 2, [x := t]\\
4. & \Gamma,\ \Delta \vdash \overline{\mathsf{lsbt}}r : \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ t) & (\forall E), 1, 3\\
5. & \Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A} \subseteq X & (Var)\\
6. & \Gamma,\ \Delta \vdash p : \forall x.\mathsf{InfbT\ A}(x) \to X(x) & (Def\ \mathcal{K} \subseteq_f \mathcal{C}), 5\\
7. & \Gamma,\ \Delta \vdash p : \mathsf{InfbT\ A}(\mathsf{lsbt}\ \ t) \to X(\mathsf{lsbt}\ \ t) & (\forall E), 6, [x := \mathsf{lsbt}\ \ t]\\
8. & \Gamma,\ \Delta \vdash p(\overline{\mathsf{lsbt}}r) : X(\mathsf{lsbt}\ \ t) & (\to E), 4, 7\\
9. & \Gamma,\ \Delta \vdash_{\mathbb{E}_{\mathsf{swapt}}} p(\overline{\mathsf{lsbt}}r) : X(\mathsf{rsbt}(\mathsf{f}\ \ t)) & (Eq), 8\\
10. & \Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{lsbt}}r) : \mathsf{InfbT\ A}(t) \to X(\mathsf{rsbt}(\mathsf{f}\ \ t)) & (\to I), 9\\
11. & \Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{lsbt}}r) : \forall t.\mathsf{InfbT\ A}(t) \to X(\mathsf{rsbt}(\mathsf{f}\ \ t)) & (\forall I), 10\\
12. & \Gamma,\ \Omega \vdash \lambda r.p(\overline{\mathsf{lsbt}}r) : \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle & (Def\ \mathcal{K} \subseteq_f \mathcal{C}), 11
\end{array}
$$

$$
\begin{array}{lll}
13. & \Gamma,\ p : \mathsf{InfbT\ A} \subseteq X \vdash \lambda q.\lambda r.p(\overline{\mathsf{lsbt}}r) : \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle & (\to I), 12\\
14. & \Gamma \vdash \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{lsbt}}r) : \mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle & (\to I), 13\\
15. & \Gamma \vdash \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{lsbt}}r) : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle\big) & (\forall^2 I), 14
\end{array}
$$

$$\therefore\ s_3 = \lambda p.\lambda q.\lambda r.p(\overline{\mathsf{lsbt}}r)$$

- 

$$
\begin{array}{lll}
1. & \Gamma \vdash s_1 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle \mathsf{A}, \mathsf{label}\rangle\big) & (Hip)\\
2. & \Gamma \vdash s_2 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{lsbt}\rangle\big) & (Hip)\\
3. & \Gamma \vdash s_3 : \forall X\big(\mathsf{InfbT\ A} \subseteq X \to \mathsf{InfbT\ A} \subseteq_f X \to \mathsf{InfbT\ A} \subseteq_f \langle X, \mathsf{rsbt}\rangle\big) & (Hip)\\
4. & \Gamma \vdash x : \mathsf{InfbT\ A}(x) & (Var)\\
5. & \Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ s_3\ x : \mathsf{InfbT\ A}(\mathsf{f}\ \ x) & (\nu I), 1, 2, 3, 4\\
6. & \Gamma \vdash \mathsf{MCoRec}\, s_1\ s_2\ s_3\ x : \mathsf{InfbT\ A}(\mathsf{swapt}\ \ x) & (Def\ \ \mathsf{f}), 5\\
7. & \vdash \lambda x.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ x : \mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{swapt}\ \ x) & (\to), 6\\
8. & \vdash \lambda x.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ x : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{swapt}\ \ x) & (\forall I), 7
\end{array}
$$

$$\therefore \vdash \overline{\mathsf{swapt}} : \forall x.\mathsf{InfbT\ A}(x) \to \mathsf{InfbT\ A}(\mathsf{swapt}\ \ x),\ \text{ donde }\ \ \overline{\mathsf{swapt}} = \lambda x.\, \mathsf{MCoRec}\, s_1\ s_2\ s_3\ x$$

$\square$

# Conclusiones

En este trabajo hemos presentado a la lógica $\mathsf{AF2}^{M\mu\nu}$, un sistema deductivo que extiende a la lógica de segundo orden $\mathsf{AF2}$ con un mecanismo de definiciones (co)inductivas cuya semántica intensional se basa en la teoría de operadores de punto fijo, en particular en el teorema de Knaster-Tarski, aunque es importante recalcar que no requerimos de un concepto sintáctico de monotonía pues nos basamos en las llamadas monotonizaciones de un operador y en sus principios de (co)inducción, conocidos también como principios de Mendler. Adicionalmente esta lógica utiliza una noción de cláusula la cual permite contar con un mecanismo de definición de tipos de datos análogo al utilizado en programación funcional. La lógica $\mathsf{AF2}^{M\mu\nu}$ permite el razonamiento ecuacional mediante la igualdad de Leibniz y cuenta con un método de codificación de pruebas que permite la extracción de programas ($\lambda$-términos) a partir de derivaciones de ciertas especificaciones de funciones definidas mediante un conjunto de ecuaciones, usualmente recursivas. Todo esto se justifica mediante el llamado paradigma de programación con pruebas desarrollado por Krivine y Parigot.

Consideramos que el desarrollo de diversos ejemplos de extracción de programas mediante esta metodología, presentados en el capítulo 4, es nuestra principal aportación, puesto que, hasta donde sabemos, no existe otra fuente de ejemplos en la literatura, sobre todo en lo que concierne al caso de tipos de datos coinductivos. Nuestra amplia colección de ejemplos, permite sostener que la lógica $\mathsf{AF2}^{M\mu\nu}$ resulta más que adecuada como un prototipo de lenguaje de programación funcional que adicionalmente cuenta con el rigor dado por la estructura lógica. En particular cualquier programa extraído de una derivación en la lógica estará libre de ciclos divergentes gracias a la propiedad de normalización fuerte.

Como posibles líneas de trabajo futuro mencionamos la cuestión de implementar esta metodología en un sistema real, por ejemplo, como método de extracción en un demostrador de teoremas o en un asistente de pruebas. Adicionalmente, si bien es una aportación nuestra el proponer un mecanismo de cláusulas reminiscente de las declaraciones de tipos de datos en lenguajes de programación funcionales, resultaría de interés introducir mecanismos aún más cercanos a dichos lenguajes como el uso de una sintaxis más amigable, similar a aquella esbozada en [13]. Finalmente, dado que la metodología empleada se limita al uso de esquemas particulares de recursión, en este caso la (co)iteración y (co)recursión, resulta importante estudiar otros esquemas, como la recursión simultanea, o la recursión por curso de valores, que permitan agregar mayor expresividad a nuestra lógica.

# Bibliografía


[1] U. Berger. Program Extraction from Normalization Proofs. Lecture Notes in Computer Science, 664, pp. 91-106. Springer 1993.

[2] J.Y. Halpern, R. Harper, N. Immerman, P. G. Kolaitis, M. Y. Vardi and V. Vianu. On the Unusual Effectiveness of Logic in Computer Science. In *The Bulletin of Symbolic Logic*. Vol. 7, No. 2, pp. 213-236. Association for Symbolic Logic, 2001. Disponible vía `http://projecteuclid.org/euclid.bsl/1182353776`.

[3] W. A. Howard. The formulae-as-types notion of construction. En To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism, pp. 479–491. Academic Press. 1980

[4] J.L. Krivine. Lambda-Calculus, Types and Models. Ellis Horwood Series in Computers and their Applications. Ellis Horwood, Masson 1993.

[5] J.L. Krivine, M. Parigot. Programming with Proofs. In *In Journal of Information Processing and Cybernetics EIK (Formerly Elektronische Informationsverarbeitung und Kybernetik)* 26(3) pp. 149-167. 1990.

[6] D. Leivant. Reasoning about Functional Programs and Complexity Classes associated with Type Disciplines. Proceedings of 24th Annual Symposium on Foundations of Computer Science pp. 460-469 IEEE Computer Science Press. 1983.

[7] P. Manoury, M. Parigot, M. Simonot. ProPre A Programming Language with Proofs. In A. Voronkov, editor, *International Conference on Logic Programming and Automated Reasoning LPAR 92*. LNAI 624 Springer Verlag 1992.

[8] P. Manoury, M. Simonot. Automatizing Terminations Proofs of Recursively Defined Functions. In *Theoretical Computer Science* 135, Elsevier 1995.

[9] N.P. Mendler. Recursive Types and Type Constraints in Second- Order Lambda Calculus. In *Proceedings of the 2nd Annual Symposium on Locig in Computer Science,Ithaca N.Y.* pp. 30-36 IEEE Computer Society Press, Washington D.C. 1987.

[10] N.P. Mendler. Inductive Types and Type Constraints in the Second- Order Lambda Calculus. *Annals of Pure and Applied Logic* 51(1-2) pp. 159-172. North-Holland 1991.

[11] F. E. Miranda Perea, Lourdes del Carmen González Huesca. Mendler-style Iso-(Co)inductive Predicates: a strongly normalizing approach. Electronic Proceeedings in Theoretical Computer Science, Vol. 81, pp. 30–46. March. 2012

[12] F. E. Miranda Perea. On Extensions of AF2 with Monotone and Clausular (Co)inductive defns. Dissertation. *Ludwig-Maximilians-Universität München*, 2004.

[13] F. E. Miranda Perea. Two Extensions of System F with (Co)iteration and Primitive (Co)recursion Principles. Theoretical Informatics and Applications (43)4, pp. 703-766, doi:10.1051/ita/2009015, 2009.

[14] F. E. Miranda Perea. *La lógica proposicional de segundo orden*. Memorias de la Sociedad Matemática Mexicana. Vol. 40, Aportaciones Matemáticas 2009.

[15] M. Parigot. On the Representation of Data in Lambda-Calculus. Lecture Notes in Computer Science 440. 1990.

[16] M. Parigot, Recursive programming with proofs. In *Theoretical Computer Science* 94, pp.335-356. Elsevier. 1992.

[17] B. C. Pierce. Types and Programming Languages. The MIT Press, 2002.

[18] 14 C. Raffalli. Data Types, Infinity and Equality in System AF2. Lecture Notes in Computer Science, 832, pp. 280-294. Springer. 1993.

[19] C. Raffalli. L' Arithmètique Fonctionnelle du Second Ordre avec Points Fixes, Thèse de l'Université Paris VII. 1994. Available via `http://www.lama.univ-savoie.fr/ RAFFALLI/`.

[20] M. H. Sørensen, Pawel Urzyczyn. "Lectures on the Curry-Howard Isomorphism". *Elsevier*. Studies in Logic and the Foundations of Mathematics. **Vol.** 149., 2006.

[21] M. Tatsuta, Realizability of Inductive Definitions for Constructive Programming. PhD Thesis, University of Tokyo, 1993.