TESIS DOCTORAL

Especificación Orientada a Objetos de Sistemas de Cálculo Simbólico

César Domínguez Pérez

UNIVERSIDAD DE LA RIOJA

TESIS DOCTORAL

Especificación Orientada a Objetos de Sistemas de Cálculo Simbólico

César Domínguez Pérez

Universidad de La Rioja
Servicio de Publicaciones
2003

Esta tesis doctoral, dirigida por los Doctores D. Laureano Lambán Pardo y D. Julio Rúbio García, fue leida el 6 de mayo de 2003, y obtuvo la calificación de Sobresaliente cum Laude por Unanimidad.

Edita: Universidad de La Rioja
Servicio de Publicaciones

ISBN 84-688-2365-1

UNIVERSIDAD DE LA RIOJA

Departamento de Matemáticas y Computación

Especificación Orientada a Objetos de Sistemas de Cálculo Simbólico

Tesis Doctoral

César Domínguez Pérez

Directores: Dr. D. Laureano Lambán Pardo
Dr. D. Julio Rubio García

Logroño, Febrero de 2003

UNIVERSIDAD DE LA RIOJA

Departamento de Matemáticas y Computación

Tesis Doctoral

Especificación Orientada a Objetos de Sistemas de Cálculo Simbólico

Presentada por

César Domínguez Pérez

para la obtención del título de Doctor en Matemáticas

Directores: Dr. D. Laureano Lambán Pardo
Dr. D. Julio Rubio García

Logroño, Febrero de 2003

La realización del presente trabajo ha sido financiada parcialmente por las ayudas ATUR00/35, ATUR01/10 y ATUR02/11 y los Proyectos de Investigación DGES PB98-1621-C02-01 y MCyT TIC2002-01626.

Agradecimientos

A Marisol

A mis padres y hermanos

Me gustaría agradecer desde estas líneas la confianza y la dedicación que han depositado en mí mis directores de tesis: Julio y Laureano. Gracias por haber estado siempre ahí cuando os he necesitado, por vuestra paciencia, vuestro interés y vuestro trato. Sin vuestras ideas, sugerencias y correcciones esta memoria no hubiera sido posible.

También quiero agradecer a Vico la ayuda que me ha prestado tanto personal como científica. Su tesis, brillante y precisa, ha supuesto un excelente punto de partida sobre el que cimentar esta memoria. Además, ha sido una gran amiga a la que recurrir en busca de buenos consejos y una excelente compañera que me ha brindado toda la ayuda imaginable cuando se la he solicitado.

No puedo olvidarme del resto de los compañeros de grupo: Ángel Luis, Juanjo, Jesús Mari, Mirian y Eloy, por las continuas muestras de ánimo y la ayuda que me han prestado en todo momento. Tampoco de mi compañera de despacho y de ciudad, Ana, por las charlas mantenidas en los continuos viajes entre Arnedo y Logroño, apoyándonos el uno en el otro cuando las cosas no iban del todo bien y compartiendo alegrías cuando al final se solucionaban. Debo agradecer también el ambiente envidiable en el que se ha desarrollado este trabajo en el seno de un Departamento excepcional así como el uso de los medios que ha puesto a mi disposición.

Por último, quisiera agradecer la compañía de mis familiares y amigos durante estos años. A Marisol, por permanecer continuamente a mi lado. A mis padres y hermanos por preocuparse día a día por la evolución de este trabajo. Y a mis amigos de cuadrilla por mostrar interés cuando trataba de explicarles lo que estaba haciendo.

A todos vosotros muchas gracias.

Índice General

Introducción

La industria del desarrollo de software está en crisis. O tal vez sería mejor decir que el software *continúa* en crisis. Desde que la terminología *crisis del software* fue acuñada a mediados de los años 60, no se han encontrado motivos para *decretar* el final de la famosa crisis. Los proyectos siguen alargándose más de lo previsto, los presupuestos resultan siempre escasos y las revistas de Ingeniería del Software siguen registrando la poca fiabilidad de algunos de los desarrollos de software más emblemáticos (misiles descontrolados, sistemas de equipajes caóticos en aeropuertos).

Dos consecuencias claras se desprenden de este panorama. En primer lugar, que el problema es difícil y que va a estar con nosotros durante mucho tiempo. Manejar la complejidad inherente al desarrollo de programas es un problema en el que ciertas limitaciones humanas, todavía no claramente elucidadas, aparecen involucradas. En segundo lugar, esta situación muestra el (relativo) fracaso de los que abogaban, en los años 60, por el uso de *métodos formales* (matemáticos) como panacea para acabar con la temida crisis.

No obstante, el punto de vista que dirige el presente trabajo de investigación es que, en ausencia de un conocimiento exacto de cuáles son los factores que influyen en la dificultad esencial del desarrollo de programas, sería arriesgado tirar por la borda la experiencia acumulada a lo largo de los años en la utilización de métodos formales en el análisis e implementación de software.

Nuestra propuesta consiste en utilizar como laboratorio experimental sistemas complejos que ya llevan años funcionando y cuyo campo de aplicación es el Cálculo Simbólico (en Topología Algebraica, concretamente). Dichos sistemas tienen la ventaja de que, proviniendo de las matemáticas, es de esperar que admitan un mejor acomodo al uso de métodos formales en su análisis. En concreto, nuestro interés se centra en la *especificación* de dichos sistemas.

En la realización de programas informáticos que nos permitan resolver algún problema, el camino que debemos recorrer entre ese problema y la implementación del mismo en un sistema de software es muy amplio. Un paso intermedio que podemos dar es la *especificación* de este problema. La especificación de un problema consiste en la descripción abstracta y formal del mismo. Abstracta en el sentido de que buscamos establecer las propiedades fundamentales presentes en dicho problema que darán lugar a las funciones requeridas al sistema que va a ser diseñado, sin preocuparse por los aspectos específicos de la implementación. Además, está expresada en algún lenguaje formal (normalmente en lenguaje matemático) para evitar las imprecisiones propias del lenguaje natural en el que habitualmente se presentan los problemas. Se plantea así la especificación con el objetivo de servir de apoyo en el desarrollo de la posterior implementación centrándose en "qué tiene que hacerse" (qué datos deben manejarse, qué operaciones deben actuar sobre estos datos), sin preocuparse de la descripción del "cómo tiene que hacerse".

Una situación inversa a la anterior, en la que también interviene la especificación como paso intermedio entre la Computación y las Matemáticas, es la *modelización* de programas ya implementados. Cuando tenemos un sistema de software ya implementado, puede ser conveniente realizar la especificación de los algoritmos y de los tipos de datos que intervienen en el mismo. El objetivo de esta especificación es ahora apoyarse en la estructura de un lenguaje formal para, por un lado, confirmar la veracidad de los resultados que nos ofrece el sistema y, por otro lado, razonar acerca de los procesos que se realizan en el mismo. A través de dicho estudio es posible desarrollar otros programas más eficaces, en cuanto a la obtención de nuevos resultados o en cuanto a la complejidad algorítmica de ese sistema.

Desde la década de los setenta se vienen usando técnicas algebraicas para la especificación de software en general y de los tipos de datos que intervienen en los programas en particular. El lenguaje del Álgebra es adecuado para los objetivos perseguidos en la especificación ya que permite definir con precisión y sin ambigüedades las componentes fundamentales de cualquier problema. La teoría de la Especificación Algebraica se ha desarrollado notablemente desde entonces y se han originado distintas técnicas de especificación. Esta teoría se ha constituido como un área de estudio dentro de las Ciencias de la Computación.

Desde hace varios años (al menos desde 1991) viene funcionando un sistema de software llamado EAT (siglas de *Effective Algebraic Topology*) [81, 82]. Este sistema fue desarrollado por J. Rubio y F. Sergeraert y está destinado al Cálculo Simbólico en Topología Algebraica. En concreto realiza cálculos de grupos de homología de algunos espacios topológicos complejos, como por ejemplo de espacios de lazos iterados. En el mismo se implementaron una serie de algoritmos diseñados por Sergeraert que se enmarcan dentro de la *Teoría de la Homología Efectiva* [88] desarrollada por este mismo autor. Antes de comenzar la construcción de este programa, Sergeraert se dio cuenta que la programación funcional debía jugar un papel fundamental en el mismo a través del concepto de *codificación funcional* [89]. Es probablemente por ello por lo que el lenguaje de programación que se utilizó para la implementación del mismo es Common Lisp.

Actualmente se ha desarrollado una segunda versión del programa que se ha llamado Kenzo[1] [34]. Una de las diferencias entre ambos sistemas, aparte de que Kenzo utiliza algoritmos más eficientes (en tiempo y espacio en memoria) y que amplía el cálculo a grupos de homología de nuevos espacios topológicos, es que Kenzo ha sido implemen-

[1]Este nombre proviene de las siglas de *Constructive Algebraic Topology* [80], es decir, CAT, siendo Kenzo el nombre del gato de Sergeraert.

tado utilizando CLOS (Common Lisp Object System) [93] que es la parte orientada a objetos de Common Lisp. En este sistema se utilizan sustancialmente las ventajas derivadas del uso de la programación orientada a objetos como por ejemplo la herencia y el polimorfismo.

A través de ambos programas se han encontrado grupos de homología que no han sido obtenidos por ningún otro método, ni teórico ni automático, y que por lo tanto no han podido ser confirmados, pero tampoco refutados; algunos ejemplos pueden encontrarse en [80] o [82].

El primer objetivo a la hora de implementar ambos sistemas fue mostrar la viabilidad de los algoritmos que Sergeraert había diseñado, pero el análisis formal del programa, y en particular del estudio de sus estructuras de datos no se llevó a cabo. No obstante, a la vista de los resultados obtenidos por los mismos parece interesante realizar dicho análisis, que aún en el caso de que no pudiese dar lugar a una prueba completa de la corrección del programa, al menos permitiese razonar sobre los procesos internos de cálculo.

Esta memoria es la continuación de una serie de trabajos [58, 57, 56] dirigidos hacia la especificación algebraica de las estructuras de datos de EAT que dieron origen a la tesis doctoral presentada por V. Pascual [72]. Relatamos a continuación brevemente los resultados obtenidos en la misma que suponen el punto de partida de esta memoria.

A la hora de comenzar dichos trabajos se observó que en la literatura no se encontraban ejemplos de especificación algebraica de sistemas tan complejos (por tamaño y conceptualmente) como EAT, por lo que la tarea a realizar no consistía simplemente en aplicar técnicas conocidas sino que era necesario introducir nuevas construcciones y conceptos. En particular se comenzó por el estudio de uno de los aspectos más importantes de EAT como son sus estructuras de datos. En este estudio se encontró que entre las estructuras de datos de EAT pueden distinguirse dos *capas*. Una primera, básica, constituida por datos *estándar* (como números, listas, árboles...) y otra segunda, funcional, dedicada a codificar estructuras algebraicas (como grupos (graduados), anillos, complejos de cadenas...), cuyos elementos son datos de la primera capa. Debemos observar que estas estructuras algebraicas son elementos que se utilizan en Topología Algebraica y además que el sistema no trabaja con una única de estas estructuras, sino que debe manejar en tiempo de ejecución familias de ellas. Una dificultad añadida es que la mayoría de estas estructuras tienen naturaleza infinita. Estas estructuras están codificadas en EAT apoyándose en la programación funcional, en concreto usando herramientas definidas por Sergeraert como la *codificación funcional* [89] y los *objetos localmente efectivos* [79].

En este estudio se comprobó que las técnicas de especificación algebraica clásicas (simplificando: las basadas en la semántica inicial) son válidas para especificar los datos de la primera capa. Sin embargo, estas técnicas no son útiles para realizar la especificación de las estructuras algebraicas que forman la segunda capa. No obstante, no fue necesario recurrir a técnicas de especificación con conjuntos infinitos de géneros y operaciones o a formalismos de orden superior, sino que resultó más conveniente utilizar una semántica final.

Para llevar a cabo la especificación de este tipo de estructuras se definió una operación, que se denominó operación *imp*, que modela el paso de un tipo de datos hasta el tipo de las familias del tipo de partida. El nombre de *imp* deriva de que lo que realmente se está especificando son implementaciones de estas estructuras más que las estructuras en sí mismas, es decir, el tratamiento a nivel de máquina que EAT realiza de ellas. Entonces, se obtuvo que las estructuras de datos de EAT forman parte de objetos finales en adecuadas categorías de implementaciones de tipos abstractos de datos. De este modo se pudo concluir que la representación elegida por EAT para implementar sus estructuras es lo "más general" posible, en el sentido de que a partir de cualquier otra implementación existe una forma de llevar ésta a la implementación elegida por EAT. Este resultado fue encontrado dentro de un marco basado en implementaciones, algo alejado de las corrientes actuales en el campo de la especificación algebraica.

Buscando relaciones con las nuevas corrientes en este campo, se encontró, de un modo bastante inesperado, que los resultados obtenidos estaban muy próximos a métodos de especificación algebraica que se habían desarrollado sobre todo en la última década, relacionados con la programación orientada a objetos, como son las especificaciones ocultas y las coálgebras. Resultó bastante sorprendente porque se tenía en mente que la orientación a objetos en el campo de la homología efectiva comenzaba en Kenzo y no en EAT. El sistema EAT está basado en la programación funcional, mientras que Kenzo está basado en la programación orientada a objetos (aunque también permanece presente en el mismo la programación funcional). De este modo, trabajando en un ambiente puramente algebraico se consiguió obtener un objeto final en una determinada categoría que resultaba ser un caso particular de objetos finales obtenidos en estos formalismos para la especificación algebraica orientada a objetos.

Este trabajo comienza donde termina el estudio realizado por V. Pascual, de modo que se intenta proseguir la tarea realizada por al menos tres caminos.

A través de la operación *imp* se consiguió relacionar distintos marcos de especificación como son la especificación algebraica ecuacional, la especificación oculta y la especificación coalgebraica. Si utilizamos el lenguaje de las instituciones, noción que formaliza la idea de un marco de especificación, la operación *imp* supone relaciones entre instituciones que corresponden con los anteriores marcos de especificación. Podemos plantearnos entonces redefinir esta operación en este ambiente más abstracto.

Otra posible vía de estudio es tratar de abarcar fragmentos cada vez mayores de EAT. El trabajo realizado previamente se ha centrado en especificar estructuras de datos "aisladas"; sin embargo, estas estructuras no "viven" solas en el programa, sino que existen operadores que permiten establecer relaciones entre ellas. El siguiente paso a dar es tratar de modelar estas relaciones.

Por último, podemos iniciar el análisis del sistema Kenzo de modo similar a como se ha hecho con EAT. Para ello debemos comprobar hasta qué punto las técnicas utilizadas para EAT son aplicables a Kenzo. Debemos tener en cuenta que para implementar las estructuras de datos de Kenzo se han aprovechado las ventajas que ofrecen las técnicas orientadas a objetos como son, entre otras, la herencia y el polimorfismo, herramientas que no están presentes en EAT.

En esta memoria se presentan los resultados obtenidos en estas tres direcciones. La estructura de la misma es la siguiente. El Capítulo 1 está dedicado a presentar las definiciones básicas y notaciones que se utilizarán a lo largo de la memoria. En concreto, se estudian brevemente las técnicas de especificación más elementales que se recogen en la especificación algebraica ecuacional. En el Capítulo 2 se traslada la construcción *imp* a relaciones entre instituciones. Estas relaciones dan lugar a un diagrama conmutativo de morfismos de instituciones que muestran distintas formas equivalentes de entender esta construcción, bien situándola en la institución algebraica ecuacional, bien en la institución oculta o bien en una nueva definición de institución coalgebraica que hemos necesitado incluir y que recoge una serie de coálgebras particulares. El Capítulo 3 está dedicado a exponer algunas de las dificultades que surgieron a la hora de establecer el anterior diagrama conmutativo, así como a plantear posibles modificaciones y generalizaciones del mismo. En particular se obtienen dos diagramas alternativos al anterior. En el primero se trata de generalizar los nodos que intervienen en el mismo. En el segundo se plantean alternativas con otros tipos de morfismos de instituciones. En el Capítulo 4 se aborda la tarea de realizar la especificación formal de relaciones entre estructuras de datos de EAT. Nos centramos en el estudio de operadores en los que intervienen dos estructuras, de modo que se construye una estructura a partir de otra. Para realizar este modelado se abandona el ámbito institucional para recuperar el lenguaje algebraico utilizado previamente en la especificación de las estructuras algebraicas por separado. El Capítulo 5 recoge los primeros pasos en la dirección de realizar la especificación de las estructuras de datos de Kenzo. En particular nos interesan las características orientadas a objetos presentes en las mismas como son la herencia y el polimorfismo. En la especificación de ambas características la noción de coerción juega un papel fundamental, aunque con interpretaciones distintas para ambas. La memoria termina con una sección dedicada a conclusiones y a trabajo futuro.

Capítulo 1

Preliminares

1.1 Introducción

El propósito de este capítulo es presentar las definiciones, notaciones y resultados básicos que utilizaremos a lo largo de esta memoria. En particular, dedicamos la primera sección a fijar la terminología referente a la Teoría de Conjuntos y la segunda a exponer los conceptos básicos de la Teoría de Categorías que necesitaremos posteriormente. En la tercera sección desarrollamos la técnica de especificación algebraica más elemental: la especificación algebraica ecuacional.

1.2 Conjuntos

A pesar de que las definiciones que presentamos en esta sección representan conocimientos básicos en matemáticas, preferimos incluirlas para fijar las notaciones que usaremos en la memoria. Son muchos los trabajos publicados que desarrollan la Teoría de Conjuntos. Nosotros hemos utilizado como referencias generales [73, 21], junto con [63] para algunas definiciones particulares.

Consideramos como primitivas las nociones de conjunto y de pertenencia ($\in$) de un elemento a un conjunto. Además, entenderemos que cada conjunto está dotado de una igualdad entre sus elementos que habitualmente llamaremos *igualdad literal* del conjunto. No entraremos en cuestiones de fundamentos de la Teoría de Conjuntos. Sin embargo, sí que necesitaremos hablar de *clases* para recoger la idea de clase de los conjuntos o del *universo* de los conjuntos. No obstante, utilizaremos el concepto de clase únicamente a un nivel intuitivo.

Usaremos los conceptos de conjunto vacío ($\emptyset$), subconjunto ($\subseteq$), conjunto diferencia ($\setminus$), producto cartesiano ($\times$) y unión e intersección ($\cup$, $\cap$, respectivamente) de conjuntos. Además, como es habitual, denotaremos por $\mathbb{N}$ al conjunto de los números naturales (que incluye al 0), por $\mathbb{N}^*$ al conjunto $\mathbb{N} \setminus \{0\}$ y por $\mathbb{Z}$ al conjunto de los números enteros.

Si S es un conjunto, $\{A_s\}_{s\in S}$ denota la familia indexada por S de los conjuntos A_s y se llama S-*conjunto*. Si R y S son dos conjuntos con $R \subseteq S$ y $A = \{A_s\}_{s\in S}$ es un S-conjunto, la restricción de la familia A a R se denota por $A|_R$, es decir $A|_R = \{A_s\}_{s\in R}$. Todas las nociones consideradas sobre conjuntos se generalizan de forma natural a familias indexadas de conjuntos.

1.2.1 Relaciones

Definición 1.2.1. Sean A y B dos conjuntos. Una *relación entre A y B* es un subconjunto $R \subseteq A \times B$. Habitualmente expresamos $(a, b) \in R$ escribiendo $a\ R\ b$.

Si $A = B$ entonces hablamos de relación (binaria) en A.

Debemos hacer notar que es habitual utilizar el término *correspondencia* para nombrar lo que aquí hemos llamado relación.

Definición 1.2.2. Sean R una relación entre A y B y C, D dos conjuntos tal que $C \subseteq A$, $D \subseteq B$. La *restricción* de la relación R a $C \times D$ es la relación entre C y D definida por $R|_{C\times D} = \{(x, y) \in R \mid x \in C,\ y \in D\}$

Relaciones sobre conjuntos especialmente importantes son las relaciones de equivalencia y de orden.

Definición 1.2.3. Sean A un conjunto y R una relación en A.

- Se dice que R es una *relación de equivalencia* si es reflexiva, simétrica y transitiva.
- Se dice que R es una *relación de orden* si es reflexiva, antisimétrica y transitiva.

Definición 1.2.4. Sean A un conjunto y R una relación de equivalencia en A. Dos elementos $a, b \in A$ tales que $a\ R\ b$ se dicen *elementos equivalentes en A módulo R*.

Para cada elemento x de A, *la clase de equivalencia de x módulo R*, que denotaremos por $[x]_R$, es el conjunto de todos los elementos de A equivalentes a x módulo R.

El conjunto de todas las clases de equivalencia módulo R se denomina *conjunto cociente* de A módulo R y se denota por $A/_R$.

1.2.2 Funciones

Un tipo de relaciones especiales son las funciones o aplicaciones.

Definición 1.2.5. Sean A y B conjuntos. Una *función parcial* o *aplicación parcial* de A en B es una relación $f \subseteq A \times B$ que verifica que, para cada $a \in A$ existe, como mucho, un $b \in B$ tal que $(a, b) \in f$.

Escribiremos $f(a) = b$ (o $f(a) := b$) en lugar de $(a, b) \in f$ y diremos que $f(a)$ es *indefinido* si no existe $b \in B$ tal que $(a, b) \in f$. En ambos casos, $f(a)$ se dice el *valor* de la función f en el *argumento* a. Denotaremos por $f\colon A \to B$ a una función parcial f de A en B. El conjunto A se llama *dominio* de f y el conjunto B *codominio.* Se llama *dominio de definición* de f al subconjunto de A dado por $Def(f) = \{a \in A \mid \text{existe } b \in B \text{ tal que } f(a) = b\}$ y *conjunto imagen* de f al subconjunto de B dado por $Im(f) = \{b \in B \mid \text{existe } a \in A \text{ tal que } f(a) = b\}$. Si $Def(f) = A$ entonces f se dice *función total.* De este modo, toda aplicación total es una aplicación parcial.

Habitualmente se suele definir una función total $f\colon A \to B$ a través de su dominio, su codominio y una *regla de correspondencia*, que también se denota por f, que permita asignar a cada elemento a de A el elemento $f(a)$ de B. Para que una función parcial $f\colon A \to B$ quede completamente definida de esta forma debemos dar además el dominio de definición $Def(f)$. De esta forma denotaremos una función parcial por $(f, Def(f), A, B)$ o bien por $(f\colon A \to B, Def(f))$ o incluso por $(f, Def(f))$ si el dominio y el codominio se deducen del contexto.

Podemos hablar simplemente de función o aplicación cuando el contexto permita identificar si se trata de una aplicación total o parcial.

La siguiente definición contiene algunas propiedades que pueden verificar las funciones.

Definición 1.2.6. Sea $f\colon A \to B$ una aplicación parcial.

- f es *inyectiva* si para cada $b \in B$, existe a lo más un elemento $a \in A$ tal que $f(a) = b$.
- f es *sobreyectiva* si para cada $b \in B$, existe al menos un elemento $a \in A$ tal que $f(a) = b$.
- f es *biyectiva* si es total, inyectiva y sobreyectiva. En este caso se dice que A y B son *conjuntos biyectivos* y lo representamos por $A \simeq B$.

Ejemplos de funciones que utilizaremos son las siguientes.

Ejemplo 1.2.7.

- Sean $f\colon A \to B$ y $g\colon B \to C$ dos funciones parciales. La función *composición* de f y g se define como la función parcial $g \circ f\colon A \to C$ con dominio de definición $Def(g \circ f) = \{a \in Def(f) \mid f(a) \in Def(g)\}$ dada por $g \circ f(a) := g(f(a))$, $\forall a \in Def(g \circ f)$.
- Sea $f\colon A \to B$ una función biyectiva. La función *inversa* de f se define como la función $f^{-1}\colon B \to A$ dada por $f^{-1}(f(a)) := a$ para todo $a \in A$. Claramente f^{-1} es total y biyectiva.
- Sea A un conjunto. La función *identidad* sobre A es la función total $id_A\colon A \to A$ que se define por $id_A(a) := a$, para todo $a \in A$.

- Sean A, B dos conjuntos con $A \subseteq B$. La función *inclusión* de A en B, es la función total $i_A \colon A \to B$ que se define por $i_A(a) := a$ para todo $a \in A$.

- Sean $f \colon A \to B$ una función parcial y $C \subseteq A$. La función *restricción* de f a C es la función parcial $f|_C \colon C \to B$ con dominio de definición $Def(f|_C) = Def(f) \cap C$ dada por $f|_C(c) := f(c)$, para todo $c \in Def(f|_C)$.

- Un *predicado* es una función con codominio el conjunto de valores booleanos $\{true, false\}$.

$\square$

Sea S es un conjunto y sean $A = \{A_s\}_{s \in S}$, $B = \{B_s\}_{s \in S}$ dos S-conjuntos. Una función $(f \colon A \to B, Def(f))$ consiste en un S-conjunto, $f = \{f_s \colon A_s \to B_s, Def(f_s)\}_{s \in S}$.

1.3 Categorías

En esta sección introducimos una serie de definiciones básicas de Teoría de Categorías que utilizaremos posteriormente en esta memoria. Es amplia la bibliografía sobre esta materia, nosotros hemos utilizado fundamentalmente [7, 60, 6].

Comenzamos dando una definición de categoría. Este concepto pretende abstraer el fondo que comparten muchas de las nociones con las que habitualmente se trabaja en diversas ramas de las matemáticas, formadas por "conjuntos con estructura" y aplicaciones entre ellos que "conservan la estructura".

Definición 1.3.1. Una *categoría* $\mathcal{C}$ consta de

- una clase $Obj(\mathcal{C})$ cuyos elementos se denominan *objetos* de la categoría $\mathcal{C}$;

- para cada par de objetos $A, B \in Obj(\mathcal{C})$, una clase $Morf_{\mathcal{C}}(A, B)$ cuyos elementos se denominan *morfismos de A en B*;

- para cada terna A, B, C de objetos de $\mathcal{C}$, una aplicación $Morf_{\mathcal{C}}(A, B) \times Morf_{\mathcal{C}}(B, C) \to Morf_{\mathcal{C}}(A, C)$ que se denomina *composición*. La composición de un par (f, g) se denota por $g \circ f$;

- para cada objeto A de $\mathcal{C}$, un morfismo distinguido en $Morf_{\mathcal{C}}(A, A)$ que se denomina *identidad* y se denota id_A,

de forma que se verifican los siguientes axiomas:

1. la composición es asociativa, es decir, dados f morfismo de $Morf_{\mathcal{C}}(A, B)$, g morfismo de $Morf_{\mathcal{C}}(B, C)$ y h morfismo de $Morf_{\mathcal{C}}(C, D)$, se verifica la siguiente igualdad

$$h \circ (g \circ f) = (h \circ g) \circ f$$

2. para cada objeto B de $\mathcal{C}$ la identidad id_B se comporta como elemento neutro para la composición. Es decir, dados f morfismo de $Morf_{\mathcal{C}}(A, B)$ y g morfismo de $Morf_{\mathcal{C}}(B, C)$ se tienen las siguientes igualdades

$$id_B \circ f = f, \quad g \circ id_B = g$$

Escribiremos $f\colon A \to B$ para representar $f \in Morf_{\mathcal{C}}(A, B)$ y llamaremos al objeto A *dominio* y al objeto B *codominio* del morfismo f. Además, escribiremos $Morf(\mathcal{C})$ para representar la clase de todos los morfismos de $\mathcal{C}$. Un morfismo $f\colon A \to A$ se dice *endomorfismo* de A.

Ejemplo 1.3.2.

- El primer ejemplo de categoría lo constituye la categoría de los conjuntos, que se denota por Set. Esta categoría tiene por objetos los conjuntos y por morfismos las aplicaciones totales. La composición y el morfismo identidad para un conjunto son la composición de aplicaciones y la aplicación identidad sobre ese conjunto.

- Otros ejemplos de categorías que utilizaremos vienen dados por estructuras matemáticas junto con las aplicaciones que "conserven" dicha estructura. Un ejemplo de este tipo de categorías es la categoría de grupos, que denotamos por GRP, cuyos objetos son los grupos y cuyos morfismos son los homomorfismos de grupos.

□

Las notaciones empleadas para referirnos a la teoría abstracta de categorías se importan del ejemplo característico de la categoría de conjuntos, pero no ha de sorprendernos la aparición de otros ejemplos de naturaleza bien distinta a los que se adapte mejor otro tipo de notación.

Habitualmente se suelen trabajar con categorías especiales que restringen el alcance de la definición anterior. Definimos a continuación alguna de ellas.

Definición 1.3.3.

- Una categoría $\mathcal{C}$ se dice *localmente pequeña* si para cada par de objetos $A, B \in Obj(\mathcal{C})$, $Morf_{\mathcal{C}}(A, B)$ es un conjunto.

- Una categoría $\mathcal{C}$ se dice *pequeña* si $Morf(\mathcal{C})$ es un conjunto.

- Una categoría $\mathcal{C}$ se dice *discreta* si sus únicos morfismos son las identidades.

En una categoría podemos identificar su clase de objetos con la clase formada por los morfismos identidad. De esto se sigue que en una categoría pequeña, la clase de objetos es un conjunto. Habitualmente se trabaja con categorías localmente pequeñas, de modo que se suele adoptar la definición de este tipo de categorías como la definición general de categoría (ver por ejemplo [7]). En este caso, una categoría pequeña es aquélla cuya clase de objetos es un conjunto. Una categoría discreta queda determinada únicamente por la clase de sus objetos, de modo que puede identificarse con dicha clase.

La siguiente definición contiene la noción de isomorfismo en una categoría.

Definición 1.3.4. Sean $\mathcal{C}$ una categoría y $f\colon A \to B$ un morfismo en $\mathcal{C}$. Se dice que f es un *isomorfismo* en $\mathcal{C}$ entre A y B si existe un morfismo en $\mathcal{C}$, $g\colon B \to C$, tal que $g \circ f = id_A$ y $f \circ g = 1_B$. En este caso los objetos A y B se dicen *isomorfos* y se denota por $A \simeq B$.

Definimos a continuación una serie de construcciones sobre categorías que utilizaremos posteriormente en esta memoria.

Definición 1.3.5. Una *subcategoría* $\mathcal{D}$ de una categoría $\mathcal{C}$ es una categoría formada por algunos de los objetos y por algunos de los morfismos de $\mathcal{C}$, tal que verifique que si A es un objeto de $\mathcal{D}$ entonces el morfismo identidad id_A en $\mathcal{C}$ está en $\mathcal{D}$, y si $f\colon A \to B$ y $g\colon B \to C$ son morfismos de $\mathcal{D}$ entonces la composición $g \circ f$ en $\mathcal{C}$ está en $\mathcal{D}$.

Dos tipos de subcategorías de una categoría que se dan con frecuencia son las subcategorías plenas y las cerradas por isomorfismos. Las primeras se caracterizan porque la subcategoría conserva todos los morfismos de la categoría para cada pareja de objetos. Las segundas por contener todos los objetos isomorfos a alguno de los objetos de la subcategoría.

Definición 1.3.6. Una subcategoría $\mathcal{D}$ de una categoría $\mathcal{C}$ se dice *plena* si para cada pareja de objetos A, B de $\mathcal{D}$ se tiene que $Morf_{\mathcal{D}}(A, B) = Morf_{\mathcal{C}}(A, B)$.

Definición 1.3.7. Sea $\mathcal{D}$ una subcategoría de una categoría $\mathcal{C}$. Se dice que $\mathcal{D}$ es *cerrada por isomorfismo* si para cada pareja de objetos $A, B \in \mathcal{C}$ si $A \in \mathcal{D}$ y $A \simeq B$ entonces $B \in \mathcal{D}$.

A partir de una categoría podemos construir otra si invertimos sus morfismos.

Definición 1.3.8. Sea $\mathcal{C}$ una categoría. La categoría *dual* de $\mathcal{C}$ que denotaremos por $\mathcal{C}^{op}$ es la categoría que tiene los mismos objetos que $\mathcal{C}$ y que para cada morfismo $f\colon A \to B$ en $\mathcal{C}$, un morfismo $f\colon B \to A$ se incluye en $\mathcal{C}^{op}$. Es decir, $Morf_{\mathcal{C}}(A, B) = Morf_{\mathcal{C}^{op}}(B, A)$.

Nos interesan algunas nociones elementales en Teoría de Categorías como son coproducto, objeto inicial y objeto final.

Definición 1.3.9. Sea J un conjunto y sea $\{A_j\}_{j\in J}$ una familia de objetos de una categoría $\mathcal{C}$. El *coproducto* de la familia $\{A_j\}_{j\in J}$ es una par $(P, \{i_j\}_{j\in J})$ donde P es un objeto de $\mathcal{C}$ y, para cada $j \in J$, $i_j\colon A_j \to P$ es un morfismo de $\mathcal{C}$, de modo que, para cualquier otro par $(Q, \{s_j\}_{j\in J})$ con Q objeto de $\mathcal{C}$ y $s_j\colon A_j \to Q$ morfismo de $\mathcal{C}$, se verifica que existe un único morfismo $F\colon P \to Q$ en $\mathcal{C}$, tal que para cada $j \in J$, $F \circ i_j = s_j$, es decir, el siguiente diagrama es conmutativo:

$$\begin{tikzcd} A_j \arrow[r, "i_j"] \arrow[dr, "s_j"'] & P \arrow[d, "F"] \\ & Q \end{tikzcd}$$

Definición 1.3.10. Sea $\mathcal{C}$ una categoría.

- Un objeto I de $\mathcal{C}$ se dice *inicial* en $\mathcal{C}$ si, para cualquier otro objeto A de $\mathcal{C}$, existe un único morfismo en $\mathcal{C}$ de I en A .
- Un objeto F de $\mathcal{C}$ se dice *final* en $\mathcal{C}$ si, para cualquier otro objeto A de $\mathcal{C}$, existe un único morfismo en $\mathcal{C}$ de A en F.

El coproducto de una familia, el objeto inicial y el objeto final pueden no existir en una categoría pero, en caso de existir, son únicos salvo isomorfismo.

1.3.1 Funtores

De la misma forma que relacionamos unos conjuntos con otros mediante aplicaciones, o conjuntos estructurados con otros mediante aplicaciones que conserven su estructura, un funtor es un morfismo entre categorías que preserva la estructura de categoría.

Definición 1.3.11. Sean $\mathcal{C}$ y $\mathcal{D}$ dos categorías. Un *funtor* F de $\mathcal{C}$ a $\mathcal{D}$, que denotaremos por $F\colon \mathcal{C} \to \mathcal{D}$, asigna:

- a cada objeto A de $\mathcal{C}$, un objeto $F(A)$ de $\mathcal{D}$;
- a cada morfismo $f\colon A \to B$ de $\mathcal{C}$, un morfismo $F(f)\colon F(A) \to F(B)$ de $\mathcal{D}$;

y verifica que:

1. conserva la composición, es decir, para cada par de morfismos $f \in Morf_{\mathcal{C}}(A, B)$, $g \in Morf_{\mathcal{C}}(B, C)$, se tiene que $F(g \circ f) = F(g) \circ F(f)$;
2. conserva las identidades, es decir, para cada objeto A de $\mathcal{C}$, $F(id_A) = id_{F(A)}$.

Un funtor F de una categoría en sí misma, $F\colon \mathcal{C} \to \mathcal{C}$, se dice *endofuntor* de $\mathcal{C}$.

Se puede definir de modo natural la composición de funtores, operación que resulta ser asociativa y, para cada categoría, existe el correspondiente funtor identidad. De este modo, las categorías junto con los funtores entre ellas constituyen una categoría: la categoría de categorías, que denotamos por CAT. En esta memoria utilizaremos una categoría más restringida que es la categoría de categorías localmente pequeñas que denotaremos por Cat.

Ejemplo 1.3.12.

1. Sean $\mathcal{C}$ y $\mathcal{D}$ dos categorías y sea A un objeto de D. El *funtor constante sobre* A es el funtor que lleva todos los objetos de $\mathcal{C}$ sobre A y todos los morfismos de $\mathcal{C}$ sobre id_A.

2. Entre una subcategoría $\mathcal{D}$ de una categoría $\mathcal{C}$ y esta última categoría se puede definir un *funtor inclusión* dado por la identidad sobre los objetos y los morfismos de $\mathcal{D}$.

3. Un tipo de funtores que se dan con cierta frecuencia son los funtores "tipo olvido" que relacionan dos categorías de modo que se olvida parte de la estructura de la primera. Por ejemplo, podemos definir el funtor olvido entre las categorías GRP y Set de modo que se prescinde de la estructura de grupo, quedándonos sólo con el conjunto subyacente a un grupo y la aplicación que subyace en un homomorfismo de grupos.

□

1.3.2 Transformaciones naturales

Existe una manera de relacionar funtores entre categorías con la idea intuitiva de relacionar cómo dichos funtores actúan sobre los mismos objetos. Esta idea da origen al concepto de *transformación natural*.

Definición 1.3.13. Sean $\mathcal{C}$ y $\mathcal{D}$ dos categorías y sean $F, G \colon \mathcal{C} \to \mathcal{D}$ dos funtores entre ellas. Una *transformación natural* entre F y G, que denotaremos por $\eta \colon F \Rightarrow G$, es una clase de morfismos $(\eta_A \colon F(A) \to G(A))_{A \in Obj(\mathcal{C})}$ de $\mathcal{D}$, un morfismo por cada objeto de $\mathcal{C}$, tal que para cada morfismo $f \colon A \to B$ de $\mathcal{C}$ se tiene que $G(f) \circ \eta_A = \eta_B \circ F(f)$, es decir, el siguiente diagrama es conmutativo:

$$\begin{tikzcd} F(A) \arrow[r, "\eta_A"] \arrow[d, "F(f)"'] & G(A) \arrow[d, "G(f)"] \\ F(B) \arrow[r, "\eta_B"'] & G(B) \end{tikzcd}$$

La composición de dos transformaciones naturales es una transformación natural, siendo además la composición asociativa. Además, para cada funtor existe la transformación natural identidad. Si trabajamos en la categoría de categorías (localmente pequeñas) Cat, dadas dos categorías $\mathcal{C}$ y $\mathcal{D}$ podemos hablar de la categoría de funtores entre $\mathcal{C}$ y $\mathcal{D}$, cuyos objetos son los funtores entre ellas y cuyos morfismos son las transformaciones naturales entre funtores.

1.4 Especificación algebraica ecuacional

A continuación introducimos algunas nociones básicas sobre especificación algebraica. Estas definiciones son bastante estándar y están ampliamente recogidas en la literatura (por ejemplo, [63] es una buena referencia, bien escrita y con muchos ejemplos y ejercicios propuestos, donde podemos encontrar estas definiciones; otras pueden ser [36] o [87]).

La idea básica dentro de la especificación algebraica es que un sistema, en nuestro caso nos referiremos siempre a un tipo de datos, puede ser modelado mediante un álgebra, es decir, como un dominio de datos junto con un conjunto de funciones sobre este dominio. No obstante, es habitual integrar distintos tipos de datos elementales, que forman parte de otro más complicado. De este modo son necesarios varios dominios de datos y se debe exigir que las funciones concuerden con el prototipo que se les asigne. Es por ello conveniente disponer de un nombre para cada uno de estos dominios de datos y también para cada una de las operaciones. Este conjunto de nombres constituye la sintaxis para un álgebra y recibe el nombre de *signatura*.

1.4.1 Signaturas

Definición 1.4.1 (Signatura). Una *signatura* Σ es una pareja de conjuntos de símbolos $\Sigma = (S, \Omega)$. Los elementos de S se denominan *géneros* y los elementos de Ω *operaciones*. Cada operación $\omega \in \Omega$ tiene asociada una *aridad*, es decir, una secuencia no vacía de elementos de S, $s_1 \dots s_n s$, que fija el perfil de la operación (es habitual escribir $\omega\colon s_1 \dots s_n \to s$). En este caso, ω es el *nombre* de una operación con n *argumentos* (cada uno de ellos del género correspondiente) y *resultado* de género s. Cuando $n = 0$, la operación $\omega\colon\ \to s$ se llama *constante* de género s.

Debemos notar que la definición anterior permite tener diferentes operaciones en una signatura con el mismo nombre, tales nombres de operaciones se dicen que están *sobrecargados* (la igualdad entre operaciones implica la igualdad de sus nombres y la igualdad de sus aridades).

Si tenemos dos signaturas $\Sigma = (S, \Omega)$ y $\Sigma' = (S', \Omega')$, decimos que Σ es una *subsignatura* de Σ' si $\Sigma \subseteq \Sigma'$, lo que significa que $S \subseteq S'$ y $\Omega \subseteq \Omega'$. La unión de signaturas, denotada por $\Sigma \cup \Sigma'$, tiene por géneros $S \cup S'$ y por operaciones $\Omega \cup \Omega'$.

Una aplicación entre signaturas, que haga corresponder sus respectivos géneros y operaciones, tal que respete la aridad de las operaciones recibe el nombre de morfismo de signaturas.

Definición 1.4.2 (Morfismo de signaturas). Un *morfismo de signaturas* $\mu\colon \Sigma \to \Sigma'$, entre dos signaturas $\Sigma = (S, \Omega)$ y $\Sigma' = (S', \Omega')$, es una pareja de aplicaciones conjuntistas $\mu = (\mu_S\colon S \to S', \mu_\Omega\colon \Omega \to \Omega')$ tal que para cada operación $\omega \in \Omega$ de aridad $s_1 \dots s_n s$, su imagen, $\mu_\Omega(\omega)$, tiene aridad $\mu_S(s_1) \dots \mu_S(s_n) \mu_S(s)$.

Denotaremos a ambas aplicaciones que integran un morfismo de signaturas por μ, si el contexto no da lugar a confusión.

Signaturas y morfismos de signaturas definen una categoría que llamamos categoría de signaturas y que denotamos por $SIG_{\mathcal{E}}$. Esta categoría constituye el ámbito sintáctico de la especificación algebraica ecuacional.

Ilustraremos con unos ejemplos estas definiciones.

Ejemplo 1.4.3.

1. Definimos una signatura `BOOL` con un único género $S = \{bool\}$ y cuyas operaciones son $\Omega = \{true\colon bool,\ false\colon bool,\ not\colon bool\ bool,\ and\colon bool\ bool\ bool\}$. Habitualmente utilizaremos un lenguaje que es más apropiado para describir signaturas:

   signatura `BOOL`
   géneros *bool*
   operaciones
   $true\colon \;\to bool$
   $false\colon \;\to bool$
   $not\colon bool \to bool$
   $and\colon bool\ bool \to bool$
   finsig.

   Como vemos, esta signatura puede considerarse como la sintaxis para el “álgebra de booleanos”.

2. Ejemplos característicos de morfismos de signaturas son los *renombrados* de géneros y operaciones. Estos constituyen morfismos de signaturas biyectivos. Por ejemplo, si tomamos las siguientes signaturas `MND` y `MND2`:

   signatura `MND`
   géneros m
   operaciones
   $bin\colon m\ m \to m$
   $e\colon \;\to m$
   finsig,

   signatura `MND2`
   géneros l
   operaciones
   $concat\colon l\ l \to l$
   $unit\colon \;\to l$
   finsig,

   la pareja $\mu = (\mu_S, \mu_\Omega)$ dada por $\mu_S(m) = l,\ \mu_\Omega(bin) = concat,\ \mu_\Omega(e) = unit$ define un morfismo de signaturas, que es un isomorfismo.

3. Consideramos ahora la signatura definida por:

signatura `GRP`
géneros g
operaciones
$prd\colon g\ g \to g$
$unt\colon\ \to g$
$inv\colon g \to g$
finsig

Entre `MND` y `GRP` podemos considerar el morfismo de signaturas $\varphi = (\varphi_S, \varphi_\Omega)$ dado por: $\varphi_S(m) = g$, $\varphi_\Omega(bin) = prd$, $\varphi_\Omega(e) = unt$. El morfismo φ corresponde a otro tipo de morfismo natural entre dos signaturas: la *inclusión* de signaturas.

□

1.4.2 Álgebras

Un álgebra atribuye significado a una signatura. Consiste en interpretar los símbolos de géneros y operaciones asociando un conjunto de datos a cada género y una función a cada operación. Representa de este modo un modelo matemático para la sintaxis definida por la signatura.

Definición 1.4.4 (Álgebra total). Sea $\Sigma = (S, \Omega)$ una signatura. Un *álgebra (total)* A para Σ (o Σ-*álgebra*) es un par de familias $A = \langle (A_s)_{s\in S}, (\omega_A)_{\omega\in\Omega}\rangle$. Para cada $s \in S$, A_s es un conjunto, llamado *soporte* en A de género s, y para cada $\omega \in \Omega$ de perfil $\omega\colon s_1 \ldots s_n \to s$, ω_A es una función total $\omega_A\colon A_{s_1}\times\cdots\times A_{s_n} \to A_s$, llamada *interpretación* en A de la operación ω.

Las funciones que se interpretan de las operaciones son funciones *totales*, posteriormente definiremos otro tipo de álgebras en la que podremos relajar esta condición, permitiendo funciones que no tengan que estar necesariamente definidas en todo el dominio, es decir, funciones *parciales*.

Se pueden definir morfismos entre álgebras, más concretamente aplicaciones entre los conjuntos soportes, de forma que sean compatibles con las funciones de ambas álgebras. Estas aplicaciones se llaman homomorfismos.

Definición 1.4.5 (Homomorfismo). Sean $\Sigma = (S, \Omega)$ una signatura y A y B dos Σ-álgebras. Un Σ-*homomorfismo* $f\colon A \to B$ de A en B es una familia de funciones $(f_s\colon A_s \to B_s)_{s\in S}$ tal que para cada operación $\omega\colon s_1 \ldots s_n \to s \in \Omega$, se verifica la siguiente *condición de homomorfismo*

$$f_s(\omega_A(a_1, \ldots, a_n)) = \omega_B(f_{s_1}(a_1), \ldots, f_{s_n}(a_n))$$

para todo $(a_1, \ldots, a_n) \in A_{s_1} \times \cdots \times A_{s_n}$.

Podemos nombrar a un Σ-homomorfismo, simplemente por homomorfismo si el contexto permite distinguir la signatura sobre la que está definido.

Las Σ-álgebras y los Σ-homomorfismos definen una categoría que llamamos categoría de Σ-álgebras y denotamos $Alg(\Sigma)$.

Ejemplo 1.4.6.

1. Un ejemplo de álgebra A para la signatura `BOOL` es el conjunto $A_{bool} = \{0, 1\}$ y las funciones $true_A = 1$, $false_A = 0$, $not_A(1) = 0$ y $not_A(0) = 1$, y para todo $a, b \in A_{bool}$, $and_A(a, b) = a * b$.

2. Un álgebra B para la signatura `MND` es la formada por el conjunto $B_m = \mathbb{Z}$ y las funciones $e_B = 0$ y para todo $a, b \in B_m$, $bin_B(a, b) = a + b$.

3. Un álgebra C para la signatura `GRP` es la formada por el conjunto $C_g = B_m$ y las funciones $e_C = e_B$, $prd_C = bin_B$ y para todo $a \in C_g$, $inv_C(a) = -a$.

4. En general, si nos fijamos en la signatura `GRP`, una `GRP`-álgebra consiste en un conjunto que tiene un elemento distinguido (correspondiente a la operación $e\colon \to g$) y que dispone de dos operaciones internas, una unaria y otra binaria. En concreto, cualquier grupo resulta ser una `GRP`-álgebra. Además, los homomorfismos de grupos son morfismos de `GRP`-álgebras. Análogamente, cualquier monoide es una `MND`-álgebra.

$\square$

Un tipo de dato puede ser modelado con un álgebra y desde un punto de vista matemático podemos decir que los tipos de datos son álgebras. Sin embargo, para el diseño de sistemas de software, interesa en la especificación no tanto la representación concreta del tipo de datos, sino solamente sus propiedades a nivel abstracto. Es decir, en la especificación no es necesario distinguir tipos de datos concretos que sean iguales bajo renombrado de los conjuntos de datos y operaciones. Es más, podría ser útil considerar esta clase de tipos similares como una unidad en un nivel más abstracto. Esto nos lleva a la noción de tipo abstracto de dato.

Definición 1.4.7 (Tipo abstracto de datos). Un *Tipo Abstracto de Datos* (TAD) para una signatura Σ es una subcategoría plena $\mathcal{C}$ de $Alg(\Sigma)$ que es cerrada bajo isomorfismo.

Un morfismo de signaturas, que define una relación entre dos signaturas, da lugar a una relación entre álgebras y homomorfismos vía la noción de reducto. No obstante, la dirección de la relación en este contexto semántico es el inverso de la que tiene el morfismo de signaturas.

Definición 1.4.8 (Reducto de un álgebra). Sean $\Sigma = (S, \Omega)$ y Σ' dos signaturas, $\mu\colon \Sigma \to \Sigma'$ un morfismo de signaturas y A' una Σ'-álgebra. El *μ-reducto* de A' es la Σ-álgebra $A'|\mu$ que tiene como soporte para cada género $s \in S$, el soporte en A' de $\mu_S(s)$, es decir, $A'|\mu_s = A'_{\mu_S(s)}$, y como interpretación de cada operación $\omega \in \Omega$, la interpretación en A' de $\mu_\Omega(\omega)$, es decir, $\omega_{A'|\mu} = \mu_\Omega(\omega)_{A'}$.

En el caso en el que μ sea un morfismo de signaturas inclusión, podemos escribir $A'|\Sigma$ en lugar de $A'|\mu$, para dejar patente que $A'|\mu$ no es sino la restricción de A' a la subsignatura Σ.

Definición 1.4.9 (Reducto de un homomorfismo). Sean $\Sigma = (S, \Omega)$ y Σ' dos signaturas, $\mu\colon \Sigma \to \Sigma'$ un morfismo de signaturas, A', B' dos Σ'-álgebras y $f'\colon A' \to B'$ un Σ'-homomorfismo. El *μ-reducto* de f' es el Σ-homomorfismo $f'|\mu\colon A'|\mu \to B'|\mu$ tal que, para cada género $s \in S$, la función de $f'|\mu$ correspondiente a s, es la función de f' para $\mu_S(s)$, es decir, $(f'|\mu)_s = f'_{\mu_S(s)}$.

Estas construcciones se extienden a un funtor modelos $Mod_{\mathcal{E}}\colon SIG_{\mathcal{E}} \to Cat^{op}$, donde Cat^{op} denota la dual de la categoría de categorías, que asocia a cada signatura Σ la categoría $Alg(\Sigma)$ de sus Σ-álgebras.

Ejemplo 1.4.10. Dados el morfismo de signatura φ entre las signaturas `MND` y `GRP` y la `GRP`-álgebra C del ejemplo anterior, el álgebra φ-reducto de C es la `MND`-álgebra $F|\varphi$ dada por B. Así, en general, la inclusión de la signatura `MND` en `GRP` induce un funtor tipo olvido entre la categoría de `GRP`-álgebras y la de `MND`-álgebras. $\square$

Una construcción especialmente interesante que utilizaremos en este trabajo es la noción de *álgebra cociente*. De modo informal, un álgebra cociente es un álgebra en la que se ha hecho una identificación sobre los elementos del conjunto soporte. Esta identificación se realiza con la ayuda de una *relación de congruencia* que es una relación de equivalencia sobre cada conjunto soporte del álgebra que es preservada por las operaciones.

Definición 1.4.11 (Relación de congruencia). Sean $\Sigma = (S, \Omega)$ una signatura y A una Σ-álgebra. Una *relación de congruencia* sobre A es una familia $Q = (Q_s)_{s\in S}$ de relaciones de equivalencia, Q_s en A_s, $s \in S$, tal que para cada operación $\omega\colon s_1 \dots s_n \to s \in \Omega$, $n \geq 1$ y para cada $a_i, a'_i \in A_{s_i}$ con $a_i\, Q_{s_i}\, a'_i$, $1 \leq i \leq n$ se verifica que:

$$\omega_A(a_1, \dots, a_n)\; Q_s\; \omega_A(a'_1, \dots, a'_n).$$

Definición 1.4.12 (Álgebra cociente). Sean $\Sigma = (S, \Omega)$ una signatura, A una Σ-álgebra, y $Q = (Q_s)_{s\in S}$ una relación de congruencia en A. El *álgebra cociente* de A por Q es la Σ-álgebra $A/_Q$ que se define por:

- el conjunto $A/_{Q_s} = \{[a]_{Q_s} \mid a \in A_s\}$, para cada género $s \in S$,
- la función $\omega_{A/_Q}([a_1]_{Q_{s_1}}, \dots, [a_n]_{Q_{s_n}}) = [\omega_A(a_1, \dots, a_n)]_{Q_s}$, para cada operación $\omega\colon s_1 \dots s_n \to s \in \Omega$, $n \geq 0$, y cada $[a_i]_{Q_{s_i}} \in A/_{Q_{s_i}}$, $i = 1, \dots, n$.

donde $[a]_{Q_s}$ denota la clase de equivalencia de a con respecto a la relación Q_s.

En general, a partir de cualquier homomorfismo entre dos álgebras que sea compatible con las relaciones de equivalencia que definen cocientes sobre las mismas, podemos definir un *homomorfismo cociente* entre las respectivas álgebras cocientes.

Proposición 1.4.13. *Sean $\Sigma = (S, \Omega)$ una signatura, A, B dos Σ-álgebras, $Q = (Q_s)_{s\in S}$ una relación de congruencia en A y $R = (R_s)_{s\in S}$ una relación de congruencia en B, y sea $h\colon A \to B$ un Σ-homomorfismo de A en B tal que para cada $s \in S$ y cada $a, a' \in A_s$ verifica que si $a\ Q_s\ a'$ entonces $h_s(a)\ R_s\ h_s(a')$. Entonces, la familia de aplicaciones $h/_{Q,R} = (h/_{Q,R_s}\colon A/_{Q_s} \to B/_{R_s})_{s\in S}$, que definimos por $h/_{Q,R_s}([a]_{Q_s}) = [h_s(a)]_{R_s}$, para cada $[a]_{Q_s} \in A/_{Q_s}$, es un Σ-homomorfismo de $A/_Q$ en $B/_R$.*

1.4.3 Ecuaciones

A partir de las constantes y de los símbolos de las operaciones de una signatura, junto con un conjunto de símbolos de *variables* para cada género, podemos construir los *términos* de la signatura. Estos términos permiten construir *ecuaciones* y las ecuaciones serán usadas como axiomas para restringir el comportamiento semántico posterior que se exige a las álgebras de una signatura. Una signatura junto con un conjunto de sus ecuaciones constituyen una *especificación*.

Definición 1.4.14 (Variables). A cada signatura $\Sigma = (S, \Omega)$ se le asocia una familia $V = (V_s)_{s\in S}$ de conjuntos infinitos disjuntos. Un elemento de V_s, $s \in S$, se llama *variable de género s*. Una familia $X = (X_s)_{s\in S}$ con $X_s \subseteq V_s$, para todo $s \in S$ se llama *conjunto de variables* para la signatura Σ. Se asume que las variables de V y los nombres de las operaciones de Ω son distintos.

De manera informal, V constituye un "universo" de variables mientras que X contiene las variables con las que vamos a trabajar. El universo se supone lo suficientemente grande para permitir añadir en cualquier momento "nuevas" variables a X.

Definición 1.4.15 (Términos). Sean $\Sigma = (S, \Omega)$ una signatura y X un conjunto de variables para ella. El conjunto de $\Sigma(X)$-*términos*, que denotamos por $T_{\Sigma(X)}$, es la familia de conjuntos $(T_{\Sigma(X),s})_{s\in S}$ que se define por inducción estructural, para cada $s \in S$, de la siguiente forma:

1. $X_s \subseteq T_{\Sigma(X),s}$;
2. si $\omega\colon\ \to s$ es una constante de Ω, entonces $\omega \in T_{\Sigma(X),s}$;
3. si $\omega\colon s_1 \dots s_n \to s$, $n \geq 1$ es una operación de Ω y si $t_i \in T_{\Sigma(X),s_i}$ para cada $i = 1, \dots, n$, entonces $\omega(t_1, \dots, t_n) \in T_{\Sigma(X),s}$.

Un elemento de $T_{\Sigma(X),s}$ se llama $\Sigma(X)$-*término* de género s, o Σ-término de género s o simplemente término de género s, si no hay lugar a confusión.

Si t es un $\Sigma(X)$-término, $Var(t)$ denota el conjunto de variables que están contenidas en t. Un término t con $Var(t) = \emptyset$, se llama *término sin variables*. El conjunto de los términos sin variables para una signatura Σ se denota por $T_\Sigma = (T_{\Sigma,s})_{s \in S}$, donde $T_{\Sigma,s}$ denota el conjunto de los términos sin variables de género s.

Ejemplo 1.4.16. Sea `MND` la signatura del Ejemplo 1.4.3 y sea $X_m = \{x, y, z\}$. Términos de género m son: x, e, $bin(bin(e, x), y)$. □

Definición 1.4.17 (Subtérmino). Un término se dice *subtérmino* de otro si ocurre como subcadena dentro de éste.

En la siguiente sección atribuiremos significado a un término dentro de un álgebra. Para ello será necesario que el género de cada término esté unívocamente determinado. Una signatura que verifique esta condición se dice *fuertemente tipada*. Claramente una signatura es fuertemente tipada si sus operaciones tienen diferentes nombres. Sin embargo, esta condición es demasiado restrictiva en la práctica, ya que incluso los lenguajes de programación clásicos permiten sobrecargar operaciones (por ejemplo, los operadores aritméticos actúan sobre distintos tipos de números). No obstante, no necesitamos todo tipo de operaciones sobrecargadas, nos restringiremos a aquellas operaciones que verifiquen que si tienen el mismo nombre y además tienen los mismos argumentos entonces obligatoriamente son iguales. El siguiente teorema demuestra que dichas aplicaciones son suficientes para garantizar que cada término en una signatura tiene un único género.

Teorema 1.4.18. *Sea $\Sigma = (S, \Omega)$ una signatura. Si para cualquier pareja de operaciones $\omega\colon s_1 \ldots s_n \to s$ y $\omega'\colon s'_1 \ldots s'_m \to s'$ de Ω con $s \neq s'$ se verifica que*

$$\text{si } \omega = \omega' \text{ y } n = m, \text{ entonces existe } i,\ 1 \leq i \leq n, \text{ con } s_i \neq s'_i,$$

entonces la signatura Σ es fuertemente tipada.

Si tenemos una relación entre signaturas dada a través de un morfismo de signaturas, podemos extender dicha relación a los correspondientes conjuntos de términos (la idea consiste en cambiar los nombres de las operaciones en los términos vía el morfismo de signaturas).

Definición 1.4.19 (Extensión de un morfismo de signaturas a términos). Sean $\Sigma = (S, \Omega)$ y $\Sigma' = (S', \Omega')$ dos signaturas, $\mu\colon \Sigma \to \Sigma'$ un morfismo de signaturas y X un conjunto se variables para Σ. La *extensión de μ a X* es el conjunto de variables para Σ' dado por $\mu_T(X) = (\cup_{\mu_S(s)=s'} X_s)_{s' \in S'}$. La *extensión de μ a un Σ-término $t \in T_{\Sigma(X)}$* es el Σ'-término $\mu_T(t) \in T_{\Sigma'(\mu_T(X))}$ que se define por inducción sobre la forma de t del siguiente modo:

- si $t = x$, una variable con $x \in X_s$, $s \in S$, entonces $\mu_T(x) = x$;
- si $t = \omega$, con $\omega\colon\ \to s$ una constante de Ω, entonces $\mu_T(\omega) = \omega'$, con $\omega' = \mu_\Omega(\omega)$;
- si $t = \omega(t_1, \ldots, t_n)$, con $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 1$, y $t_i \in T_{\Sigma(X),s_i}$, $i = 1, \ldots, n$, entonces $\mu_T(\omega(t_1, \ldots, t_n)) = \omega'(\mu_T(t_1), \ldots, \mu_T(t_n))$.

Prescindiremos del subíndice escribiendo $\mu(t)$ en lugar de $\mu_T(t)$ si el contexto no da lugar a confusión.

Ejemplo 1.4.20. El renombrado entre las signaturas `MND` y `MND2` del Ejemplo 1.4.3 extiende los `MND`-términos de género m: x, e, $bin(bin(e,x),y)$ a los `MND2`-términos de género l: x, $unit$, $concat(concat(unit,x),y)$. □

Definición 1.4.21 (Ecuación). Sea $\Sigma = (S, \Omega)$ una signatura. Una Σ-*ecuación* e es una terna $e = (X, t_1, t_2)$ donde X es un conjunto de variables para Σ y $t_1,\ t_2$ son dos $\Sigma(X)$-términos de mismo género, es decir $t_1,\ t_2 \in T_{\Sigma(X),s}$ para algún género $s \in S$. Podemos llamarla simplemente *ecuación* si no hay ambigüedad en la signatura sobre la que está construida.

Cuando X es exactamente el conjunto de variables que están presentes en t_1 y t_2, por simplicidad, no se suele indicar explícitamente el conjunto de variables X. En este caso se expresa una ecuación por $t_1 = t_2$.

Ejemplo 1.4.22. Podemos definir las siguientes ecuaciones para `MND`: $bin(x, bin(y,z)) = bin(bin(x,y),z)$, $bin(x,e) = x$, $bin(e,x) = x$. □

Un morfismo de signaturas puede extenderse a las ecuaciones para estas signaturas, simplemente trasformando por el morfismo cada uno de los términos que forman la ecuación.

Definición 1.4.23 (Extensión de un morfismo de signaturas a ecuaciones). Sea $\mu\colon \Sigma \to \Sigma'$ un morfismo de signaturas y $e = (X, t_1, t_2)$ una Σ-ecuación. La *extensión de* e *por* μ es la Σ'-ecuación $\mu(e) = (\mu(X), \mu(t_1), \mu(t_2))$.

Esta construcción nos lleva a definir un funtor *sentencias*, $Sen_{\mathcal{E}}\colon SIG_{\mathcal{E}} \to Set$, donde Set denota la categoría de conjuntos, que asocia a cada signatura Σ el conjunto de ecuaciones en Σ.

Definición 1.4.24 (Especificación). Una *especificación* consiste en una pareja $Espec = (\Sigma, E)$, con Σ una signatura y E un conjunto de ecuaciones en Σ.

Ejemplo 1.4.25. La signatura `MND` junto con las ecuaciones del ejemplo anterior constituye la especificación habitual para los monoides. □

1.4.4 Satisfacibilidad

La última de las componentes básicas de un marco de especificación consiste en una relación entre sentencias y modelos que permita establecer un criterio para determinar la "veracidad" de una ecuación en un determinado modelo. Para ello es necesario previamente dotar de interpretación semántica dentro de un modelo a cada uno de los términos que forman la ecuación. Comenzamos por asignar significado a sus variables.

Definición 1.4.26 (Valoración). Sean $\Sigma = (S, \Omega)$ una signatura, X un conjunto de variables para Σ y A una Σ-álgebra. Una *valoración* q de X en A es una aplicación $q\colon X \to A$, es decir, una familia de funciones $q = (q_s\colon X_s \to A_s)_{s\in S}$.

Una vez que hemos asignado un valor en un álgebra a cada una de las variables, podemos asignar valor en dicha álgebra a los términos construidos sobre estas variables.

Definición 1.4.27 (Interpretación de términos). Sean $\Sigma = (S, \Omega)$ una signatura, X un conjunto de variables para Σ, A una Σ-álgebra, $q\colon X \to A$ una valoración y $t \in T_{\Sigma(X)}$ un término (fuertemente tipado). La *interpretación* $\overline{q}(t)$ del término t por la valoración q en el álgebra A se define por inducción sobre la estructura de t del siguiente modo:

- si $t = x$, una variable con $x \in X_s$, $s \in S$, entonces $\overline{q}(x) = q(x)$;
- si $t = \omega$, con $\omega\colon\ \to s$ una constante de Ω, entonces $\overline{q}(t) = \omega_A$;
- si $t = \omega(t_1, \ldots, t_n)$, con $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 1$, $t_i \in T_{\Sigma(X),s_i}$ para cada $i = 1, \ldots, n$, entonces $\overline{q}(t) = \omega_A(\overline{q}(t_1), \ldots, \overline{q}(t_n))$.

Nota 1.4.28. La condición de fuertemente tipado es necesaria para garantizar la consistencia de la definición de forma que se asigne una única interpretación a cada término.

Para la interpretación de los términos sin variables no es necesaria la intervención externa de una valoración, por lo que el valor semántico que se asigna a estos términos viene directamente determinado por el álgebra. Denotaremos este valor por $A(t)$ con t un término sin variables.

Definición 1.4.29 (Relación de satisfacción). Sean Σ una signatura, A una Σ-álgebra y $e = (X, t_1, t_2)$ una Σ-ecuación. Se dice que A *satisface* e y lo denotamos $A \models^{\mathcal{E}}_{\Sigma} e$, si para cada valoración en A de las variables de e, $q\colon X \to A$, se verifica que $\overline{q}(t_1) = \overline{q}(t_2)$.

Diremos que una Σ-álgebra A satisface un conjunto de ecuaciones E, y lo denotamos por $A \models^{\mathcal{E}}_{\Sigma} E$, si A satisface todas las ecuaciones del conjunto E.

Una especificación $Espec = (\Sigma, E)$ determina una subcategoría de $Alg(\Sigma)$, la subcategoría plena generada por aquellas Σ-álgebras que satisfacen E. Esta categoría se denota $Mod_{\Sigma}(E)$.

Teorema 1.4.30. *Para toda especificación $Espec = (\Sigma, E)$, la categoría $Mod_{\Sigma}(E)$ es un TAD.*

Ejemplo 1.4.31. La categoría de modelos de la especificación que hemos definido para `MND` es la categoría de los monoides en el ejemplo anterior. □

El ejemplo de especificación algebraica que hemos desarrollado constituye el formalismo más extendido para la especificación de tipos de datos. De hecho, una buena parte de los sistemas de especificación con los que habitualmente se trabaja son generalizaciones o variantes del marco algebraico ecuacional. Por ejemplo, para poder especificar tipos de datos en los que algunos de los operadores sean parciales, podemos generalizar la noción de álgebra de modo que sus operaciones puedan ser parciales. Otras posibles generalizaciones se obtienen al considerar diferentes tipos de sentencias (lógica de predicados, por ejemplo). Podemos encontrar una introducción a algunas de estas generalizaciones en [63, 4]. En el siguiente capítulo utilizaremos varias de estas generalizaciones. Una de ellas constituye, por ejemplo, la especificación oculta [42] que consiste, entre otras cosas, en distinguir en las signaturas dos tipos de géneros: visibles y ocultos.

Capítulo 2

Instituciones orientadas a objetos

2.1 Introducción

En el capítulo anterior hemos esbozado el modo habitual de trabajar con especificaciones algebraicas. Las nociones básicas son cuatro: signatura, álgebra (o modelo), ecuación (o sentencia) y satisfacibilidad. El concepto de *institución* (introducido por Goguen y Burstall [39]) pretende agrupar, de un modo cohesionado, estas cuatro componentes.

Este concepto ha sido ampliamente utilizado en la literatura, podemos citar entre muchos otros trabajos [46, 96, 95, 98, 71, 19, 26, 70, 8], de modo que se han presentado multitud de ejemplos de instituciones y variantes del concepto original de institución. En nuestro intento de modelar usando la teoría de instituciones las estructuras de datos que aparecen en los sistemas de cálculo simbólico EAT y Kenzo, hemos encontrado que las instituciones que mejor se adecúan (aunque no exactamente en el modo que aparecen en la literatura) son aquellas que han sido propuestas para tratar con el paradigma de programación orientada a objetos. En particular, son centrales para nuestra aproximación las instituciones basadas en especificaciones ocultas [38, 41, 15, 45]. En este capítulo, trataremos con detalle esta tecnología, introduciendo también otra familia de instituciones coalgebraicas: las basadas en los métodos coalgebraicos [83, 54].

2.2 Nociones básicas. Institución algebraica ecuacional $\mathcal{E}$

En el concepto de institución [39] se reúnen las cuatro componentes básicas que constituyen un marco abstracto de especificación: signaturas, sentencias, modelos y relación de satisfacibilidad.

Definición 2.2.1 (Institución). Una *institución* $\mathcal{I} = (SIG, Sen, Mod, \models)$ consta de:

1. una categoría SIG, llamada *categoría de signaturas*,
2. un funtor $Sen\colon SIG \to Set$, llamado *funtor sentencias*,
3. un funtor $Mod\colon SIG \to Cat^{op}$, llamado *funtor modelos*, y
4. una relación $\models_\Sigma \subseteq Obj(Mod(\Sigma)) \times Sen(\Sigma)$ para cada $\Sigma \in Obj(SIG)$, llamada Σ-*satisfacción*,

tal que para cada morfismo $\mu\colon \Sigma \to \Sigma'$ en SIG, se verifica la siguiente *condición de satisfacibilidad*:

$$Mod(\mu)(A') \models_\Sigma e \quad \text{si y sólo si} \quad A' \models_{\Sigma'} Sen(\mu)(e)$$

para cada $A' \in Obj(Mod(\Sigma'))$ y cada $e \in Sen(\Sigma)$.

De modo natural aparece la cuestión de establecer pasos entre instituciones, de forma que se puedan trasladar especificaciones dadas en el marco definido por una institución a otras en un marco distinto. Esto conduce al concepto de morfismo de instituciones [39, 46].

Definición 2.2.2 (Morfismo de instituciones). Sean $\mathcal{I} = (SIG, Sen, Mod, \models)$, $\mathcal{I}' = (SIG', Sen', Mod', \models')$ dos instituciones. Un *morfismo de instituciones* entre $\mathcal{I}$ e $\mathcal{I}'$, $\Phi\colon \mathcal{I} \to \mathcal{I}'$, consta de:

1. un funtor $\Phi\colon SIG \to SIG'$,
2. una transformación natural $\alpha\colon Sen' \circ \Phi \Rightarrow Sen$, y
3. una transformación natural $\beta\colon Mod \Rightarrow Mod' \circ \Phi$

tal que verifican la siguiente *condición de satisfacibilidad*:

$$A \models_\Sigma \alpha_\Sigma(e') \qquad \text{si y sólo si} \qquad \beta_\Sigma(A) \models'_{\Phi(\Sigma)} e'$$

para cualquier signatura Σ de $\mathcal{I}$, cualquier Σ-modelo A de $\mathcal{I}$ y cualquier $\Phi(\Sigma)$-sentencia e' de $\mathcal{I}'$.

Observamos que las componentes sintáctica y semántica en un morfismo de instituciones van en direcciones opuestas. Sin embargo, existen ejemplos en los que parece más natural que ambas transformaciones naturales vayan en la misma dirección. Surge así el concepto de *morfismo de instituciones unidireccional* ("forward") [46, 95].

Definición 2.2.3 (Morfismo de instituciones unidireccional). Sean $\mathcal{I} = (SIG, Sen, Mod, \models)$, $\mathcal{I}' = (SIG', Sen', Mod', \models')$ dos instituciones. Un *morfismo de instituciones unidireccional* entre $\mathcal{I}$ e $\mathcal{I}'$, $\Phi\colon \mathcal{I} \to \mathcal{I}'$, consta de:

1. un funtor $\Phi\colon SIG \to SIG'$,
2. una transformación natural $\alpha\colon Sen \Rightarrow Sen' \circ \Phi$, y
3. una transformación natural $\beta\colon Mod \Rightarrow Mod' \circ \Phi$

tal que verifican la siguiente *condición de satisfacibilidad*:

$$A \models_\Sigma e \qquad \text{si y sólo si} \qquad \beta_\Sigma(A) \models'_{\Phi(\Sigma)} \alpha_\Sigma(e)$$

para cualquier signatura Σ, cualquier Σ-modelo A y cualquier Σ-sentencia e de $\mathcal{I}$.

Nota 2.2.4. En general, denotamos una institución $\mathcal{I}$ por la cuaterna $\mathcal{I} = (SIG, Sen, Mod, \models)$, y si el contexto da lugar a confusión por $\mathcal{I} = (SIG_\mathcal{I}, Sen_\mathcal{I}, Mod_\mathcal{I}, \models^\mathcal{I})$. De la misma forma, denotamos un morfismo de instituciones $\Phi\colon \mathcal{I} \to \mathcal{I}'$ por la terna $\Phi = (\Phi, \alpha, \beta)$, y si hay lugar a confusión por $\Phi^{\mathcal{I}-\mathcal{I}'} = (\Phi^{\mathcal{I}-\mathcal{I}'}, \alpha^{\mathcal{I}-\mathcal{I}'}, \beta^{\mathcal{I}-\mathcal{I}'})$. En todo momento indicaremos si estamos trabajando con un morfismo de instituciones o con un morfismo de instituciones unidireccional por lo que no incluimos en la notación referencia explícita para distinguirlos.

Podemos componer morfismos de instituciones o morfismos de instituciones unidireccionales de la forma esperada, de modo que las instituciones junto con cualquiera de estos tipos de morfismos forman una categoría (ver, por ejemplo, [95]).

Ejemplo 2.2.5. La especificación algebraica ecuacional que hemos desarrollado en el capítulo de preliminares constituye un primer ejemplo de institución (ver [39]) que llamamos *institución algebraica ecuacional* y que denotamos $\mathcal{E} = (SIG_\mathcal{E}, Sen_\mathcal{E}, Mod_\mathcal{E}, \models^\mathcal{E})$. $\square$

2.3 La institución algebraica ecuacional definida sobre un universo de datos $\mathcal{E}^D$

Guiados por lo que ocurre cuando queremos especificar implementaciones vamos a definir una variante de la institución algebraica ecuacional. Cuando trabajamos con implementaciones de estructuras algebraicas, como por ejemplo los grupos, lo razonable es trabajar con implementaciones que compartan un mismo modo de representación en el computador para los elementos de esas estructuras. Esta restricción se traduce, a nivel de modelos, en la necesidad de limitar las familias de álgebras a aquellas en las que todas las álgebras que constituyen la familia compartan los mismos soportes. Así, dada una signatura $\Sigma = (S, \Omega)$ suponemos fijado un dominio de datos $D = \{D_s\}_{s\in S}$ de modo que consideramos los posibles modelos que se pueden obtener para esta signatura con soportes en D. A partir de la institución algebraica ecuacional definimos de esta forma una nueva institución cuyos modelos tengan la característica anterior. Observamos que antes de dar esa definición es necesario fijar un dominio de datos sobre el que se construye dicha institución.

Consideramos fijado un conjunto U que llamamos *universo de géneros*. Entonces, para cada $s \in U$ también fijamos un conjunto D_s (no vacío). La familia $D = \{D_s\}_{s \in U}$ se llama *universo de datos*. A continuación definimos la *institución algebraica ecuacional definida sobre el universo de datos* D, que denotaremos por $\mathcal{E}^D = (SIG_{\mathcal{E}^D}, Sen_{\mathcal{E}^D}, Mod_{\mathcal{E}^D}, \models^{\mathcal{E}^D})$, de la siguiente forma.

Signaturas: La categoría $SIG_{\mathcal{E}^D}$ tiene como objetos las signaturas $\Sigma = (S, \Omega \cup C)$, donde el conjunto S de géneros cumple que $S \subseteq U$, el conjunto C es el conjunto de operaciones constantes $\{d\colon\ \to s \mid \text{para cada } d \in D_s,\ \text{para cada } s \in S\}$ y Ω es un conjunto de operaciones sobre S. Además las constantes de Ω tienen nombres distintos de las constantes de C. Los morfismos de $SIG_{\mathcal{E}^D}$ son los morfismos identidad[1].

Sentencias: Definimos el funtor de sentencias $Sen_{\mathcal{E}^D}\colon SIG_{\mathcal{E}^D} \to Set$ como la restricción a la categoría $SIG_{\mathcal{E}^D}$ del funtor sentencias $Sen_{\mathcal{E}}\colon SIG_{\mathcal{E}} \to Set$ de la institución algebraica ecuacional $\mathcal{E}$.

Modelos: El funtor $Mod_{\mathcal{E}^D}\colon SIG_{\mathcal{E}^D} \to Cat^{op}$ asocia a cada objeto $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, la categoría que denotaremos por $Mod_{\mathcal{E}^D}(\Sigma)$. Esta categoría tiene como objetos las Σ-álgebras A que verifiquen que $A_s = D_s$ para cada género $s \in S$, es decir, el conjunto soporte de A, en cada género $s \in S$, es el conjunto correspondiente a ese género en el universo de datos. Y además $d_A = d$ para cada $d \in C$, es decir, el elemento asociado por A a cada constante $d\colon\ \to s$ de C es exactamente d. Los morfismos de la categoría $Mod_{\mathcal{E}^D}(\Sigma)$ son los Σ-homomorfismos identidad.

Completamos la definición del funtor $Mod_{\mathcal{E}^D}\colon SIG_{\mathcal{E}^D} \to Cat^{op}$ sobre los morfismos identidad en $SIG_{\mathcal{E}^D}$ de la única forma posible, es decir, $Mod_{\mathcal{E}^D}(id_\Sigma) = id_{Mod_{\mathcal{E}^D}(\Sigma)}$, para cada morfismo identidad id_Σ de $SIG_{\mathcal{E}^D}$.

Relación de satisfacción: Para cada $\Sigma \in SIG_{\mathcal{E}^D}$ la relación $\models_\Sigma^{\mathcal{E}^D}$ entre $Obj(Mod_{\mathcal{E}^D}(\Sigma))$ y $Sen_{\mathcal{E}^D}(\Sigma)$ se define como la restricción de $\models_\Sigma^{\mathcal{E}}$ a los conjuntos anteriores.

Es inmediato comprobar que las cuatro componentes anteriores verifican la condición de satisfacibilidad, ya que los únicos morfismos en $SIG_{\mathcal{E}^D}$ son las identidades. Obtenemos de esta forma una institución, lo que recogemos en el siguiente teorema.

Teorema 2.3.1. *La tupla $\mathcal{E}^D = (SIG_{\mathcal{E}^D}, Sen_{\mathcal{E}^D}, Mod_{\mathcal{E}^D}, \models^{\mathcal{E}^D})$ constituye una institución.*

Nota 2.3.2. Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, observamos que el nombre "sintáctico" que hemos elegido para cada constante de C es el mismo que el dato

[1] Definimos la categoría $SIG_{\mathcal{E}^D}$ como una categoría discreta. El motivo de esta restricción en los morfismos de la categoría será explicado más adelante, en el siguiente capítulo.

"semántico" que se le asigna en todas las álgebras de $Mod_{\mathcal{E}^D}$ para esta signatura en $\mathcal{E}^D$. Sin embargo, a las constantes de Ω les podemos asignar distintos valores en las diferentes álgebras de esta categoría. Por esta característica exigimos que las constantes de los conjuntos Ω y C presenten nombres distintos. En caso de permitir una constante en $\Omega \cap C$, esta constante quedaría fijada, por pertenecer a C, en todas las álgebras de la categoría.

2.3.1 Un primer morfismo de instituciones unidireccional

En el sistema EAT [81] conviven dos clases de estructuras de datos. Por un lado, podemos encontrar un *primer tipo* de estructuras de datos que se corresponde con los datos usuales como números enteros, listas (finitas) o árboles de símbolos (para representar combinaciones lineales, polinomios, o estructuras similares). Estas estructuras de datos pueden ser modeladas utilizando las técnicas habituales de especificación algebraica (y en particular utilizando el modelo inicial de semántica [63]). Por otro lado, existen un *segundo tipo* de estructuras de datos correspondiente a estructuras matemáticas como grupos (graduados), anillos, conjuntos simpliciales, en los que sus elementos son datos del *primer tipo*. En [59] se muestra cómo la semántica inicial no resulta adecuada para estas estructuras y en [58] se introduce una operación sobre tipos abstractos de datos que permite formalizar la forma de trabajar con este tipo de estructuras.

Podemos explicar los aspectos sintácticos de esta operación por medio del siguiente ejemplo.

Consideramos la signatura `GRP` con un único género g y tres operaciones:

$prd\colon g\ g \to g$
$inv\colon g \to g$
$unt\colon\ \to g$

Obviamente esta signatura es la base para una especificación algebraica de un grupo, sobre el conjunto de datos abstraído por el género g. Pero si, como es usual en los paquetes de Computación Simbólica, es necesario manejar varios grupos sobre el mismo conjunto de datos, se presenta entonces otro tipo de dato que permanece oculto en la signatura `GRP`: el tipo de los grupos representados sobre g. Si hacemos explícito este tipo invisible (u oculto), obtenemos una nueva signatura que denotamos $\mathtt{GRP}_{imp}$ con dos géneros g e $imp\_\mathtt{GRP}$, y las operaciones:

$imp\_prd\colon imp\_\mathtt{GRP}\ g\ g \to g$
$imp\_inv\colon imp\_\mathtt{GRP}\ g \to g$
$imp\_unt\colon imp\_\mathtt{GRP}\ \to g$

A través de esta signatura pretendemos especificar no grupos sino familias de grupos. En [58] se justifica la elección del prefijo *imp* en esta signatura. Mediante este prefijo se quiere resaltar su relación con el concepto de *implementación*, ya que el objetivo que buscamos con el paso de una signatura a otra es especificar familias de grupos mediante una implementación de cada uno de ellos, que se modelan a través del nuevo género. En este trabajo aunque no entraremos a estudiar cuestiones de implementación, sí que mantendremos la notación que se eligió en [58]. Entonces, como hemos comentado, si pretendemos trabajar con familias de grupos es conveniente que los grupos de cada familia compartan el mismo soporte, es decir, el ambiente adecuado en el que situar la signatura `GRP` es $SIG_{\mathcal{E}^D}$. En la signatura $\mathtt{GRP}_{imp}$, el nuevo género pretendemos que especifique a familias de grupos, de modo que cada elemento de este género nos permita recuperar las operaciones de un grupo (sobre el soporte fijado), por lo tanto el soporte de este género debe estar libre y no podemos incluir esta segunda signatura en $SIG_{\mathcal{E}^D}$, es decir, la situamos por ahora en $SIG_{\mathcal{E}}$.

Esta operación entre signaturas puede ser ampliada a morfismos de signaturas junto con una relación entre las álgebras y las sentencias de las correspondientes signaturas que verifique la condición de satisfacibilidad. Obtenemos entonces una relación entre los ingredientes fundamentales que forman parte de un marco de especificación que englobamos en el concepto de institución. De este modo, transformamos la operación anterior en un morfismo entre instituciones. El tipo de morfismo de instituciones que parece natural utilizar en este caso es un morfismo que asigne a sentencias y modelos para la signatura que represente un grupo `GRP`, sentencias y modelos para la signatura que representa familias de grupos $\mathtt{GRP}_{imp}$, es decir un morfismo de instituciones *unidireccional*. Este uso de los morfismos entre instituciones es novedoso en el sentido en el que nosotros no pretendemos a través del morfismo obtener un paso que nos permita movernos entre dos marcos de especificación [95, 46], sino que utilizamos esta herramienta para explicar una operación entre los mismos.

Definimos a continuación un morfismo de instituciones unidireccional $\Phi = (\Phi, \alpha, \beta)$ entre la institución algebraica ecuacional definida sobre un universo de datos D, $\mathcal{E}^D = (SIG_{\mathcal{E}^D}, Sen_{\mathcal{E}^D}, Mod_{\mathcal{E}^D}, \models^{\mathcal{E}^D})$, y la institución algebraica ecuacional $\mathcal{E} = (SIG_{\mathcal{E}}, Sen_{\mathcal{E}}, Mod_{\mathcal{E}}, \models^{\mathcal{E}})$, a través de la cual concretamos las características de la operación anterior.

En primer lugar definimos un funtor Φ de la categoría de signaturas $SIG_{\mathcal{E}^D}$ a la categoría de signaturas $SIG_{\mathcal{E}}$. Para ello debemos establecer una imagen para cada objeto y cada morfismo de $SIG_{\mathcal{E}^D}$.

Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, definimos $\Phi(\Sigma)$ como la signatura $\Sigma_{imp} = (S_{imp}, \Omega_{imp} \cup C)$. Esta signatura está formada por el conjunto de géneros $S_{imp} = \{imp_{\Sigma}\} \cup S$, con imp_{Σ} un nuevo género que no está presente en U. El conjunto de operaciones Ω_{imp} está formado por una operación $imp\_\omega\colon imp_{\Sigma} s_1 \ldots s_n \to s$, por cada operación $\omega\colon s_1 \ldots s_n \to s \in \Omega$. Es decir, obtenemos las operaciones de Ω_{imp} si agregamos el nuevo género como primer argumento a todas las operaciones de Ω. Las constantes de C se mantienen inalteradas. Obviamente, para completar la definición de Φ, la imagen del morfismo identidad de una signatura Σ será la correspondiente identidad de Σ_{imp}.

En el siguiente lema demostramos que la aplicación que acabamos de definir es un funtor.

Lema 2.3.3. *La aplicación Φ define un funtor de $SIG_{\mathcal{E}^D}$ en $SIG_{\mathcal{E}}$.*

Demostración. Observamos que Φ está bien definida ya que podemos distinguir las operaciones constantes de Ω de las operaciones constantes de C porque tienen nombres distintos. A nivel de morfismos no hay nada que probar ya que $SIG_{\mathcal{E}^D}$ es una categoría discreta y, por definición de Φ, se conservan las identidades. ■

Una vez que hemos definido la primera componente del morfismo de instituciones unidireccional, daremos a continuación la segunda componente que relaciona las sentencias de ambas instituciones. Se trata de definir una transformación natural $\alpha\colon Sen_{\mathcal{E}^D} \Rightarrow Sen_{\mathcal{E}} \circ \Phi$, entre el funtor sentencias $Sen_{\mathcal{E}^D}\colon SIG_{\mathcal{E}^D} \to Set$ de la institución $\mathcal{E}^D$ y el funtor $Sen_{\mathcal{E}} \circ \Phi\colon SIG_{\mathcal{E}^D} \to Set$. Esta transformación natural consiste en una familia de aplicaciones, una aplicación $\alpha_\Sigma\colon Sen_{\mathcal{E}^D}(\Sigma) \to Sen_{\mathcal{E}}(\Sigma_{imp})$ para cada signatura Σ de $SIG_{\mathcal{E}^D}$, que verifique la condición de naturalidad respecto de los morfismos.

Dada una signatura Σ de $SIG_{\mathcal{E}^D}$ definimos la aplicación $\alpha_\Sigma\colon Sen_{\mathcal{E}^D}(\Sigma) \to Sen_{\mathcal{E}}(\Sigma_{imp})$ como la relación que asocia a cada Σ-ecuación e, la Σ_{imp}-ecuación $\alpha_\Sigma(e)$ que se obtiene a partir de e a través del siguiente proceso. Añadimos al conjunto de variables de la ecuación e una *nueva* variable de género imp_Σ, que denotaremos por z_{imp}. A continuación, en cada uno de los términos de la ecuación, sustituimos cada operación de Ω por la operación correspondiente en Ω_{imp}, y no modificamos las variables y las constantes de C. Por último, completamos el término con la variable z_{imp} en la primera componente de cada operación de Ω_{imp} que forme parte de este término.

Utilizamos el siguiente ejemplo para ilustrar esta aplicación.

Ejemplo 2.3.4. Dada la categoría $SIG_{\mathcal{E}^D}$, elegimos un género g del universo de géneros. Para este género tenemos fijado un conjunto de datos D_g. Entonces la signatura `GRP`, que hemos definido anteriormente para tratar de especificar un grupo, pertenece a $SIG_{\mathcal{E}^D}$, si añadimos a la misma los elementos de D_g como constantes. Mediante esta signatura pretendemos especificar los grupos que podemos definir sobre el soporte D_g.

Dadas las variables de género g: x, x_1, x_2, x_3, algunas ecuaciones para esta signatura son:

$$
\begin{aligned}
&prd(prd(x1, x2), x3) = prd(x1, prd(x2, x3))\\
&prd(unt, x) = x\\
&prd(inv(x), x) = unt\\
&prd(x, inv(x)) = unt
\end{aligned}
$$

entonces, las ecuaciones correspondientes por la aplicación $\alpha_{\mathtt{GRP}}$ para la signatura $\mathtt{GRP}_{imp}$ son:

$$
\begin{aligned}
&imp\_prd(z_{imp}, imp\_prd(z_{imp}, x1, x2), x3) = imp\_prd(z_{imp}, x1, imp\_prd(z_{imp}, x2, x3))\\
&imp\_prd(z_{imp}, imp\_unt(z_{imp}), x) = x\\
&imp\_prd(z_{imp}, imp\_inv(z_{imp}, x), x) = imp\_unt(z_{imp})\\
&imp\_prd(z_{imp}, x, imp\_inv(z_{imp}, x)) = imp\_unt(z_{imp})
\end{aligned}
$$

□

Podemos definir la aplicación anterior de un modo más preciso a través de un proceso de inducción sobre la estructura de los términos que constituyen la ecuación. Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, la aplicación $\alpha_\Sigma \colon Sen_{\mathcal{E}^D}(\Sigma) \to Sen_{\mathcal{E}}(\Sigma_{imp})$ asocia a cada Σ-ecuación $e = (X, t, u)$ de $Sen_{\mathcal{E}^D}(\Sigma)$, la Σ_{imp}-ecuación $\alpha_\Sigma(e) := (X \cup \{z_{imp}\}, \theta(t), \theta(u))$, donde $\theta = (\theta_s \colon T_{\Sigma(X),s} \to T_{\Sigma_{imp}(X \cup \{z_{imp}\}),s})_{s \in S}$ es la familia de aplicaciones entre términos que definimos a continuación. Dado $s \in S$, definimos la aplicación θ_s por inducción sobre la estructura de un término $t \in T_{\Sigma(X),s}$ del siguiente modo:

- si $t = x \in X_s$, una variable de género s, $\theta_s(x) = x$;
- si $t = d$, con $d \colon \to s$ una constante de C, entonces $\theta_s(d) = d$;
- si $t = \omega(t_1, \ldots, t_n)$, con $\omega \colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 0$ y $t_i \in T_{\Sigma(X),s_i}$, $i = 1, \ldots, n$, entonces $\theta_s(\omega(t_1, \ldots, t_n)) = imp\_\omega(z_{imp}, \theta_{s_1}(t_1), \ldots, \theta_{s_n}(t_n))$.

Obtenemos entonces el siguiente lema.

Lema 2.3.5. *La familia de aplicaciones $\alpha \colon Sen_{\mathcal{E}^D} \Rightarrow Sen_{\mathcal{E}} \circ \Phi$ constituye una transformación natural.*

Demostración. Como los únicos morfismos de $SIG_{\mathcal{E}^D}$ son las identidades, obtenemos de modo trivial que α verifica la condición de naturalidad. ∎

Para completar la definición de un morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{E}$ nos falta por definir una transformación natural $\beta \colon Mod_{\mathcal{E}^D} \Rightarrow Mod_{\mathcal{E}} \circ \Phi$ entre el funtor modelos $Mod_{\mathcal{E}^D} \colon SIG_{\mathcal{E}^D} \to Cat^{op}$ y el funtor $Mod_{\mathcal{E}} \circ \Phi \colon SIG_{\mathcal{E}^D} \to Cat^{op}$. Para ello necesitamos definir una familia de funtores, un funtor $\beta_\Sigma \colon Mod_{\mathcal{E}^D}(\Sigma) \to Mod_{\mathcal{E}}(\Sigma_{imp})$ para cada signatura Σ de $SIG_{\mathcal{E}^D}$.

Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$ definimos un funtor $\beta_\Sigma \colon Mod_{\mathcal{E}^D}(\Sigma) \to Mod_{\mathcal{E}}(\Sigma_{imp})$ de la siguiente forma. Para cada Σ-álgebra A de $Mod_{\mathcal{E}^D}(\Sigma)$, $\beta_\Sigma(A)$ es la Σ_{imp}-álgebra que está formada por los conjuntos $\beta_\Sigma(A)_s := D_s$ para cada $s \in S$ y $\beta_\Sigma(A)_{imp_\Sigma} := \{*\}$ (es decir, el soporte del género imp_Σ es el conjunto unipuntual $\{*\}$). La interpretación en el álgebra $\beta_\Sigma(A)$ de la operación $imp\_\omega \colon imp_\Sigma s_1 \ldots s_n \to s$, $n \geq 0$, de Ω_{imp}, es la función definida por:

$$imp\_\omega_{\beta_\Sigma(A)}(*, a_1, \ldots, a_n) = \omega_A(a_1, \ldots, a_n),$$

y de cada constante $d \colon \to s$ de C, el elemento: $d_{\beta_\Sigma(A)} = d$.

Para completar la definición del funtor β_Σ debemos establecer la imagen de los morfismos de $Mod_{\mathcal{E}^D}(\Sigma)$. Ya que en $Mod_{\mathcal{E}^D}(\Sigma)$ sólo se han considerado las identidades, establecemos que la imagen por un funtor β_Σ de la identidad de un álgebra A sea la identidad de $\beta_\Sigma(A)$.

Obtenemos entonces el siguiente lema.

Lema 2.3.6. *La familia de funtores $\beta\colon Mod_{\mathcal{E}^D} \Rightarrow Mod_{\mathcal{E}} \circ \Phi$ constituye una transformación natural.*

Demostración. De nuevo la demostración de la condición naturalidad de esta familia de funtores es trivial ya que los únicos morfismos en $SIG_{\mathcal{E}^D}$ son identidades. ∎

Tenemos definidas todas las componentes que forman parte de un morfismo de instituciones unidireccional. Éstas se recogen en el siguiente resultado.

Teorema 2.3.7. *La terna $\Phi = (\Phi, \alpha, \beta)$ definida anteriormente es un morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{E}$.*

Demostración. Completamos la demostración de este resultado si comprobamos que se verifica la condición de satisfacibilidad. Es decir, si para cada signatura Σ de $SIG_{\mathcal{E}^D}$,

$$A \models_{\Sigma}^{\mathcal{E}^D} e \quad \text{si y sólo si} \quad \beta_{\Sigma}(A) \models_{\Phi(\Sigma)}^{\mathcal{E}} \alpha_{\Sigma}(e)$$

para cualquier sentencia $e \in Sen_{\mathcal{E}^D}(\Sigma)$ y cualquier álgebra $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$.

Nos centramos primero en la demostración de la implicación directa.

Sean una signatura $\Sigma = (S, \Omega \cup C) \in Obj(SIG_{\mathcal{E}^D})$, una Σ-ecuación $e = (X, t, u) \in Sen_{\mathcal{E}^D}(\Sigma)$, con $t, u \in T_{\Sigma(X),s}$, $s \in S$, y una Σ-álgebra $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$. Suponemos que $A \models_{\Sigma}^{\mathcal{E}^D} (X, t, u)$, es decir, que para toda valoración $p\colon X \to A$ se verifica que

$$\overline{p}(t) = \overline{p}(u).$$

Tenemos que demostrar si $\beta_{\Sigma}(A) \models_{\Phi(\Sigma)}^{\mathcal{E}} \alpha_{\Sigma}(e)$. Como $\alpha_{\Sigma}(e) = (X \cup \{z_{imp}\}, \theta(t), \theta(u))$, donde z_{imp} es una nueva variable que no está presente en X, entonces debemos probar que para cada valoración $q\colon X \cup \{z_{imp}\} \to \beta_{\Sigma}(A)$, se verifica que

$$\overline{q}(\theta(t)) = \overline{q}(\theta(u)).$$

Dada una valoración arbitraria $q\colon X \cup \{z_{imp}\} \to \beta_{\Sigma}(A)$ por $\beta_{\Sigma}(A)$, consideramos la valoración $q_{|S}\colon X \to A$ por A que se obtiene al restringir q a los géneros de S. Si aplicamos la hipótesis con esta valoración obtenemos que

$$\overline{q_{|S}}(t) = \overline{q_{|S}}(u).$$

Así, es suficiente probar que

$$\overline{q}(\theta(t)) = \overline{q_{|S}}(t)$$

para cualquier término t de género s de S, ya que entonces

$$\overline{q}(\theta(t)) = \overline{q_{|S}}(t) = \overline{q_{|S}}(u) = \overline{q}(\theta(u)).$$

Demostramos la igualdad anterior por inducción sobre la estructura de un término $t \in T_{\Sigma(X),s}$, $s \in S$:

- si $t = x \in X_s$, entonces $\overline{q}(\theta(x)) = \overline{q}(x) = q(x) = q_{|S}(x) = \overline{q_{|S}}(x)$;
- si $t = d$, con $d\colon \to s$ una constante de C, entonces $\overline{q}(\theta(d)) = \overline{q}(d) = d = \overline{q_{|S}}(d)$;
- y si $t = \omega(t_1, \ldots, t_n)$, con $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 0$, y $t_i \in T_{\Sigma(X),s_i}$, $i = 1, \ldots, n$, entonces:

$$
\begin{aligned}
&\overline{q}(\theta(\omega(t_1, \ldots, t_n))) = \\
&\quad = \overline{q}(imp\_\omega(z_{imp}, \theta(t_1), \ldots, \theta(t_n))) \\
&\quad = imp\_\omega_{\beta_\Sigma(A)}(\overline{q}(z_{imp}), \overline{q}(\theta(t_1)), \ldots, \overline{q}(\theta(t_n))) \\
&\quad = imp\_\omega_{\beta_\Sigma(A)}(*, \overline{q}(\theta(t_1)), \ldots, \overline{q}(\theta(t_n))) \\
&\quad = \omega_A(\overline{q}(\theta(t_1)), \ldots, \overline{q}(\theta(t_n))) \text{ por definición de } \beta_\Sigma(A), \\
&\quad = \omega_A(\overline{q_{|S}}(t_1), \ldots, \overline{q_{|S}}(t_n)) \text{ por hipótesis de inducción,} \\
&\quad = \overline{q_{|S}}(\omega(t_1, \ldots, t_n)).
\end{aligned}
$$

La demostración de la implicación inversa es totalmente análoga por inducción sobre los términos, tras la observación de que cada valoración $X \to A$ por A puede ser extendida a una valoración $q\colon X \cup \{z_{imp}\} \to \beta_\Sigma(A)$ por $\beta_\Sigma(A)$, sin más que definir $q(z_{imp}) = *$. ■

2.3.2 Ampliación del morfismo a especificaciones

Como ya se ha comentado en el capítulo de preliminares, uno de los vínculos fundamentales entre las matemáticas y los programas reales es la noción de *tipo abstracto de datos* (y, más en concreto, el concepto de *implementación* de un tipo abstracto de datos [49], [58]). Un tipo abstracto de datos viene dado por una signatura Σ y una categoría $\mathcal{C}$ de Σ-álgebras cerrada por isomorfismo. De este modo, una especificación $Espec = (\Sigma, E)$ define un tipo abstracto de datos: la subcategoría plena de Σ-álgebras que satisfacen las ecuaciones de E.

En el morfismo de instituciones anterior queda implícita una operación entre especificaciones $\phi\colon$ Σ-especificaciones $\to$ Σ_{imp}-especificaciones. La nueva especificación construida pretende representar implementaciones de álgebras definidas por la primera especificación.

Dada una especificación $Espec = (\Sigma, E)$ de $\mathcal{E}^D$, podemos construir otra especificación $\phi(Espec) = (\Sigma_{imp}, E_{imp})$, que denotamos por $Espec_{imp}$, con $\Sigma_{imp} = \Phi(\Sigma)$, y para cada ecuación e de E la ecuación $e_{imp} = \alpha_\Sigma(e)$ se incluye en E_{imp}, donde Φ y α son, respectivamente, el funtor de signaturas y la transformación natural sobre sentencias del morfismo de instituciones unidireccional definido anteriormente.

La relación de satisfacibilidad de este morfismo de instituciones nos permite enunciar el siguiente resultado:

Proposición 2.3.8. *Sea* $Espec = (\Sigma, E)$ *una especificación de* $\mathcal{E}^D$*, con* $\Sigma = (S, \Omega \cup C)$*, y sea* $A = \langle A_{imp_\Sigma},\ (D_s)_{s \in S},\ (imp\_\omega_A \colon A_{imp_\Sigma} \times D_{s_1} \times \cdots \times D_{s_n} \to D_s)_{\omega\colon s_1 \ldots s_n \to s \in \Omega}, (d_A = d)_{d \in C}\rangle$ *una* Σ_{imp}*-álgebra que sea modelo para* $Espec_{imp} = (\Sigma_{imp}, E_{imp})$*. Entonces, cada elemento* $a \in A_{imp_\Sigma}$ *define una* Σ*-álgebra* B_a *de la siguiente forma:* $B_a = \langle (D_s)_{s \in S},\ (imp\_\omega_A(a, \_) \colon D_{s_1} \times \cdots \times D_{s_n} \to D_s)_{\omega\colon s_1 \ldots s_n \to s \in \Omega}, (d_A = d)_{d \in C}\rangle$ *que es modelo para* $Espec = (\Sigma, E)$*.*

A partir de esta proposición obtenemos una interpretación de la construcción *imp*: las álgebras de la signatura Σ_{imp} corresponden con familias de álgebras de la signatura inicial Σ, de modo que el género distinguido en Σ_{imp} se puede considerar como un conjunto de índices para dicha familia. La idea que subyace es que dicho género puede verse como un modelo para *implementaciones* de Σ-álgebras (de ahí el nombre elegido para la construcción) ya que cada dato de este género permite acceder a la información necesaria para recuperar (las operaciones de) una Σ-álgebra. (Remitimos a [58] o a [72] para una explicación más detallada sobre la interpretación de esta operación.)

Ejemplo 2.3.9. Un primer ejemplo que, aunque no representa adecuadamente la complejidad de algunos de los tipos de datos que aparecen en EAT, permite ilustrar la construcción anterior, es la especificación para un grupo $Espec_{\mathtt{GRP}}$. Ésta se compone de la signatura `GRP`, con un conjunto prefijado D_g que serán los elementos de los grupos, junto con ecuaciones para definir un grupo como las dadas en el Ejemplo 2.3.4.

La proposición anterior indica que dado un modelo A de $Espec_{\mathtt{GRP}_{imp}}$, cada elemento a de $A_{imp_{\mathtt{GRP}}}$ (soporte del género $imp_{\mathtt{GRP}}$) nos permite recuperar un modelo para $Espec_{\mathtt{GRP}}$, es decir un grupo sobre el soporte D_g, mediante la terna:

$$(imp\_prd_A(a, -, -), imp\_inv_A(a, -), imp\_unt_A(a)).$$

Las siguientes álgebras concretas muestran la relación existente entre `GRP`-álgebras y $\mathtt{GRP}_{imp}$-álgebras. Al estar prefijado el dominio de datos D, debemos observar que las signaturas `GRP` que aparecen a continuación en los tres ejemplos son distintas, ya que varía su soporte de datos. No obstante, utilizaremos para todas ellas el mismo nombre, así como el mismo género ya que sí que comparten la idea de especificación de un grupo a través de un único género y de tres operaciones.

1. Suponemos que $\mathbb{Z}$ es el dominio en D para el género g. Definimos la $\mathtt{GRP}_{imp}$-álgebra A modelo para $Espec_{\mathtt{GRP}_{imp}}$ que tiene como soporte en A para género $imp_{\mathtt{GRP}}$ a $\mathbb{N}$. La interpretación en A de las operaciones es:

   $imp\_prd_A(n, z1, z2) = z1 + z2$
   $imp\_inv_A(n, z) = -z$
   $imp\_unt_A(n) = 0$

   De esta forma, cada $n \in \mathbb{N}$ determina una `GRP`-álgebra modelo para $Espec_{\mathtt{GRP}}$ que denotamos A_n, siendo $A_n = \langle \mathbb{Z}, \{+_{\mathbb{Z}}, -_{\mathbb{Z}}, 0_{\mathbb{Z}}\}\rangle$. Por tanto, la $\mathtt{GRP}_{imp}$-álgebra A admite ser descrita como una colección indexada por $\mathbb{N}$ de copias de $\mathbb{Z}$.

2. Fijado un $n \in \mathbb{N}$, $n > 1$, consideramos el conjunto $\langle n \rangle = \{0, 1, \ldots, n-1\}$. Suponemos que $\langle n \rangle$ es el dominio en D para el género g. Definimos la $\mathtt{GRP}_{imp}$-álgebra B modelo para $Espec_{\mathtt{GRP}_{imp}}$ si tomamos como soporte para el género $imp_{\mathtt{GRP}}$ el conjunto $B_{imp_{\mathtt{GRP}}} = \{$matrices $n \times n$ | la matriz es la tabla del producto de un grupo sobre $\langle n \rangle\}$. De cada tabla, pueden extraerse los inversos de cada elemento así como el elemento neutro, por lo que la $\mathtt{GRP}_{imp}$-álgebra B representa de forma canónica a una familia de grupos.

   Cualquier grupo finito es isomorfo a uno de los grupos indexados por el conjunto $B_{imp_{\mathtt{GRP}}}$ anterior, por lo que, barriendo todos los $n \in \mathbb{N}$, se cubren todos los grupos finitos.

   Por ejemplo, fijamos $n = 6$ y consideramos el grupo de permutaciones de tres elementos que denotamos por $\mathcal{P}_3$. Vamos a suponer, por concretar, que los tres elementos permutados son $1, 2$ y 3. Si asignamos a cada permutación un ordinal según el orden lexicográfico, es decir, si tomamos la biyección $enum_{\mathcal{P}_3} \colon \mathcal{P}_3 \longrightarrow \langle n \rangle$ dada por: $enum_{\mathcal{P}_3}((1\ 2\ 3)) = 0$, $enum_{\mathcal{P}_3}((1\ 3\ 2)) = 1$, $enum_{\mathcal{P}_3}((2\ 1\ 3)) = 2$, $enum_{\mathcal{P}_3}((2\ 3\ 1)) = 3$, $enum_{\mathcal{P}_3}((3\ 1\ 2)) = 4$ y $enum_{\mathcal{P}_3}((3\ 2\ 1)) = 5$, la siguiente matriz representa la tabla de multiplicar del grupo:

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & 3 & 2 & 5 & 4 \\ 2 & 4 & 0 & 5 & 1 & 3 \\ 3 & 5 & 1 & 4 & 0 & 2 \\ 4 & 2 & 5 & 0 & 3 & 1 \\ 5 & 3 & 4 & 1 & 2 & 0 \end{pmatrix}.$$

3. Consideramos C con el mismo soporte que el álgebra A para el género g y el soporte $\mathbb{N} \setminus \{0, 1\}$ para el género $imp_{\mathtt{GRP}}$, cuyas operaciones vienen dadas por:

$$\begin{array}{l} imp\_prd_C(n, z1, z2) = (z1 + z2)\ mod\ n \\ imp\_inv_C(n, z) = -z\ mod\ n \\ imp\_unt_C(n) = 0 \end{array}$$

   Para cada $n \in \mathbb{N} \setminus \{0, 1\}$, obtenemos la $\mathtt{GRP}$-álgebra $C_n = \langle \mathbb{Z}, \{+_{mod\ n}, -_{mod\ n}, 0_{\mathbb{Z}}\} \rangle$ que tiene el mismo comportamiento que el grupo $\mathbb{Z}/n\mathbb{Z}$ (notar que C_n no es isomorfo a $\mathbb{Z}/n\mathbb{Z}$, de hecho ni siquiera es grupo, es decir no verifica los axiomas de la especificación, pero existe un cociente del mismo que sí lo es).

□

Debemos notar que la idea expresada en la proposición anterior de ver las álgebras de una signatura Σ_{imp} como una familia de Σ-álgebras no es del todo cubierta por el morfismo de instituciones. Esto sí que es cierto desde un punto de vista sintáctico, es decir, la signatura Σ_{imp} procedente de la imagen del morfismo puede ser realmente interpretada como una signatura para las familias de Σ-álgebras. Sin embargo, desde un punto de vista semántico, los modelos en la imagen del morfismo representan implementaciones casi triviales (concretamente implementaciones de familias formadas por una única álgebra).

En la siguiente sección introducimos el marco de la especificación oculta. Este marco nos va permitir redefinir el morfismo de instituciones anterior de modo que utilizaremos como institución de llegada para nuestra construcción una institución oculta. En dicho marco, las imágenes de los modelos por el morfismo podrán ser reunidas (mediante coproducto) en una nueva álgebra, que sí recoge la anterior idea y que corresponde directamente con la implementación de las estructuras de datos del *segundo tipo* elegida en EAT.

2.4 Especificaciones ocultas

2.4.1 Conceptos básicos sobre especificaciones ocultas

Esta sección consiste en un breve resumen de algunos conceptos básicos sobre la teoría de las especificaciones ocultas. Esta teoría tiene su origen en un trabajo de Goguen [38] y ha sido ampliamente desarrollada en estos últimos años por diversos autores. Algunos de los trabajos que ha originando son [41, 15, 40, 22, 45, 27, 43]. Una referencia adecuada, en la que se da una visión general de este tipo de especificaciones es [42].

La idea fundamental de esta teoría es tratar de recoger la diferencia que existe dentro del paradigma orientado a objetos entre los datos que son utilizados como valores y los objetos propiamente dichos. Mientras que los datos son visibles y su representación es accesible y compartida por los elementos que forman el sistema, los objetos tienen un estado que es oculto, es decir, que únicamente puede observarse indirectamente a través del efecto de los métodos al actuar sobre ellos. Para modelar esta situación se distinguen dos tipos de géneros: géneros visibles y géneros ocultos, a través de los cuales se diferencia ambas situaciones.

Definición 2.4.1 (Signatura oculta). Sean $V\Sigma = (VS, V\Omega)$ una signatura y D una $V\Sigma$-álgebra, donde se incluyen en $V\Omega$, como constantes, los elementos de los conjuntos soporte de D que no correspondan con constantes previas de $V\Omega$. Llamaremos a la signatura $V\Sigma$ *signatura visible*. En particular, llamaremos a los elementos de VS *géneros visibles* y a los de $V\Omega$ *operaciones visibles*. La $V\Sigma$-álgebra D se llama *álgebra visible*. Una *signatura oculta* definida sobre $V\Sigma$ y D es una signatura $H\Sigma = (S, \Omega)$ donde $S = HS \sqcup VS$ y $\Omega = H\Omega \sqcup V\Omega$ tal que:

- cada operación $\omega\colon s_1 \dots s_n \to s$ en Ω con $s_i, s \in VS$, $\forall i = 1, \dots, n$, pertenece a $V\Omega$,
- para cada operación $\omega\colon s_1 \dots s_n \to s$ en Ω, en $s_1, \dots, s_n$ existe como mucho un género oculto. Por comodidad, se asume que este género oculto aparece en la primera posición (es decir, es s_1).

Llamaremos a los elementos de HS *géneros ocultos* y a los elementos de $V\Omega$ *operaciones ocultas*. Si no hay lugar a confusión, hablaremos de signatura oculta sin hacer referencia a la signatura y álgebra visibles.

La condición que exige incluir en la signatura visible como constantes todos los datos del álgebra visible se da para asegurar que todos los datos visibles puedan ser generados de modo que tengamos la posibilidad de nombrar a todos ellos. De este modo al álgebra visible se le llama también *álgebra de datos* o *dominio de datos*. Si a partir de la signatura visible se puede asegurar que todos los datos visibles pueden ser generados por las operaciones de la signatura, no será necesario incluir esta condición.

Debemos observar que la únicas operaciones de una signatura oculta $H\Sigma$ que involucran exclusivamente a géneros visibles son las operaciones visibles. Si tenemos en cuenta que el fijar un álgebra de datos viene motivado por la necesidad de disponer de una representación para los datos, el hecho de haber incluido los elementos del álgebra D como constantes en la signatura hace que si consideramos simplemente las operaciones constantes, tengamos un sistema de representantes para los datos. Esto hace que las operaciones visibles (no constantes) sean superfluas desde el punto de vista de la constructibilidad de los datos (el resultado devuelto al aplicar cualquiera de ellas es un elemento de D, ya incluido como constante en la signatura). En este sentido debemos notar que es natural considerar, no el álgebra de términos de la signatura visible, sino un cociente de ésta en el que se haya identificado cada término visible con la constante obtenida por su interpretación en el álgebra D. De este modo podemos considerar como términos visibles solamente las constantes.

Podemos dar una clasificación de las operaciones ocultas dependiendo del papel que jueguen los géneros ocultos en las mismas.

Sea $\omega\colon s_1 \dots s_n \to s$ una operación de una signatura oculta. Se dice que ω:

- es una operación *constructora*, si el género de llegada s es un género oculto,
- es una operación *deconstructora*, si el género de llegada s es un género visible.

A su vez, una operación constructora $\omega\colon s_1 \dots s_n \to s$ es:

- constructora *a partir de datos* cuando $s_1, \dots, s_n$ son géneros visibles,
- constructora *a partir de objetos* (y datos) cuando s_1 es diferente de s (ambos, por supuesto, géneros ocultos), y
- *operación de actualización* cuando $s_1 = s$.

Definición 2.4.2 (Morfismo de signaturas ocultas). Sean $H\Sigma = (S, \Omega)$ y $H\Sigma' = (S', \Omega')$ dos signaturas ocultas sobre la misma signatura visible $V\Sigma = (VS, V\Omega)$ y dominio de datos D. Un *morfismo de signaturas ocultas* $\mu = (\mu_S, \mu_\Omega)\colon H\Sigma \to H\Sigma'$ es un morfismo de signaturas tal que:

1. $\mu_S(v) = v$, para cada $v \in VS$ y $\mu_\Omega(\phi) = \phi$, para cada $\phi \in V\Omega$,
2. $\mu_S(HS) \subseteq HS'$,
3. μ_S es inyectiva,
4. y para cada operación $\omega' \colon s'_1 s'_2 \dots s'_n \to s'$, $n \geq 1$, de $H\Sigma'$ tal que $s'_1 \in \mu_S(HS)$, entonces existe alguna operación ω en $H\Sigma$ tal que $\omega' = \mu_\Omega(\omega)$.

Un *morfismo vertical de signaturas ocultas* entre dos signaturas ocultas (sobre la misma signatura visible y el mismo dominio de datos) es un morfismo de signaturas al que se exigen las condiciones primera y segunda de la definición anterior.

En ocasiones llamaremos a un morfismo de signaturas ocultas *morfismo de signaturas oculto* e incluso *morfismo oculto*.

Las dos primeras condiciones de un morfismo de signaturas ocultas indican que el morfismo de signaturas preserva la visibilidad y la no visibilidad tanto de los géneros como de las operaciones. La cuarta condición expresa que no es posible definir *nuevas* operaciones ocultas en la signatura de llegada con un género oculto entre sus argumentos que sea imagen de un género oculto de la salida.

La condición de inyectividad en los géneros ocultos es necesaria para poder definir correctamente una categoría de signaturas ocultas cuyos morfismos sean los morfismos ocultos. Si no exigimos esta condición, no necesariamente obtenemos un morfismo de signaturas ocultas a partir de la composición de dos morfismos de signaturas ocultas. En el siguiente ejemplo mostramos la necesidad de dicha condición que, por cierto, ha sido pasada por alto en la construcción de otras categorías de signaturas ocultas similares [38, 41, 43].

Ejemplo 2.4.3. Consideramos tres signaturas ocultas, Σ_1, Σ_2 y Σ_3, que definimos por:

signatura Σ_1	**signatura** Σ_2	**signatura** Σ_3
géneros $h_1,\ v$	**géneros** $h_1,\ h_2,\ v$	**géneros** $h_1,\ v$
operaciones	**operaciones**	**operaciones**
$+\colon h_1\ v\ v \to v$	$+\colon h_1\ v\ v \to v$	$+\colon h_1\ v\ v \to v$
finsig,	$*\colon h_2\ v\ v \to v$	$*\colon h_1\ v\ v \to v$
	finsig,	**finsig**.

Estas tres signaturas ocultas están definidas sobre la misma signatura visible $V\Sigma$, que tiene un género visible v, y dominio de datos D_v, que es un conjunto cualquiera que no es necesario hacer explícito para nuestros propósitos. Para que estén bien definidas como signaturas ocultas deben incluir como constantes los elementos de D_v. Los géneros ocultos en las signaturas son h_1 y h_2.

Consideramos el morfismo de signaturas ocultas inclusión μ entre Σ_1 y Σ_2 que se define por la aplicación sobre los géneros dada por $\mu(v) = v$ y $\mu(h_1) = h_1$, y la aplicación sobre las operaciones $\mu(+) = +$. Por otro lado, consideramos un morfismo de signaturas φ entre Σ_2 y Σ_3 que definimos por la aplicación sobre los géneros dada por $\varphi(v) = v$, $\varphi(h_1) = h_1$ y $\varphi(h_2) = h_1$, y la aplicación sobre las operaciones dada por $\varphi(+) = +$ y $\varphi(*) = *$. Este segundo morfismo de signaturas verifica las propiedades de un morfismo de signaturas ocultas a excepción de la condición de inyectividad. Sin embargo, su composición no verifica la cuarta propiedad que se exige a un morfismo de signaturas ocultas. El género oculto h_1 de Σ_3 verifica que $\mu \circ \varphi(h_1) = h_1$, con h_1 género oculto de Σ_1. Además está presente en el soporte de la operación $*$ en Σ_3. Pero esta operación no es imagen por $\mu \circ \varphi$ de ninguna operación de Σ_1. □

Para definir una categoría de signaturas ocultas fijaremos un universo de géneros y además un conjunto de datos para cada uno de estos géneros. En este universo de géneros estarán los elementos entre los que se elegirán los géneros de las distintas signaturas visibles en nuestra categoría y, en los conjuntos de datos asociados a estos géneros, los correspondientes dominios de datos.

Definición 2.4.4 (Categoría de signaturas ocultas sobre un universo de datos D). Consideramos un conjunto U (el *universo de géneros visibles*) fijado. Entonces, para cada $s \in U$ un conjunto D_s (no vacío) es también fijado. La familia $D = \{D_s\}_{s \in U}$ se llama *universo de datos*. Definimos la *categoría de signaturas ocultas sobre el universo de datos D*, y la denotaremos por $SIG_{\mathcal{H}^D}$, a la categoría que tiene por objetos las signaturas ocultas $H\Sigma$ definidas sobre una signatura visible $V\Sigma = (VS, V\Omega)$ y una $V\Sigma$-álgebra de datos $D_{V\Sigma}$, de modo que los géneros de $V\Sigma$ están contenidos en el universo de géneros U y los conjuntos soporte del dominio de datos $D_{V\Sigma}$ son los correspondientes conjuntos del universo de datos D. Los morfismos de la categoría $SIG_{\mathcal{H}^D}$ son los morfismos de signaturas ocultas.

Habitualmente abusaremos de la notación nombrando D al universo de datos prefijado así como al álgebra visible particular asociada a cada signatura. Expresamos de este modo que el álgebra visible asociada a una signatura oculta representa los datos del universo con los que vamos a trabajar dentro de esta signatura.

En el siguiente lema comprobamos que la categoría anterior está bien definida.

Lema 2.4.5. *$SIG_{\mathcal{H}^D}$ es una categoría.*

Demostración. Para cada signatura oculta $H\Sigma$ definida sobre $V\Sigma$ y $D_{V\Sigma}$ el morfismo de signaturas identidad $id_{H\Sigma}$ es de modo evidente un morfismo de signaturas oculto.

Dados dos morfismos de signaturas ocultos $\mu\colon H\Sigma \to H\Sigma'$ y $\mu'\colon H\Sigma' \to H\Sigma''$, tenemos que demostrar que $\mu' \circ \mu$ es un morfismo de signaturas oculto, es decir, que cumple las cuatro propiedades que caracterizan a un morfismo de signaturas oculto:

1. Es claro que la composición $\mu' \circ \mu$ fija géneros y operaciones visibles si μ y μ' lo hacen.

2. $\mu' \circ \mu(HS) = \mu'(\mu(HS)) \subseteq \mu'(HS') \subseteq HS''$.

3. $\mu' \circ \mu_S$ es inyectiva ya que es composición de aplicaciones inyectivas.

4. Dada una operación $\omega'' \colon s''_1 \dots s''_n \to s'' \in H\Omega''$ con $s''_1 = \mu' \circ \mu_S(s_1)$, donde $s_1 \in HS$, tenemos que probar que existe una operación ω en $H\Sigma$ tal que $\omega'' = \mu' \circ \mu_\Omega(\omega)$. Como μ' es un morfismo de signaturas oculto y $\omega'' \colon s''_1 \dots s''_n \to s'' \in H\Omega''$ con $s''_1 = \mu'_S(\mu_S(s_1))$ y $\mu_S(s_1) \in HS'$, entonces existe una operación $\omega' \colon s'_1 \dots s'_n \to s'$ en $H\Sigma'$ tal que $\omega'' = \mu'_\Omega(\omega')$. Tenemos que $\mu'_S(s'_1) = s''_1$, y que $\mu'_S(\mu_S(s_1)) = s''_1$, entonces, como μ'_S inyectiva, obtenemos que $s'_1 = \mu_S(s_1)$. Es decir, tenemos una operación $\omega' \colon s'_1 \dots s'_n \to s'$ en $H\Sigma'$ con $s'_1 = \mu_S(s_1)$, donde $s_1 \in HS$. Entonces, como μ es un morfismo de signaturas oculto, existe una operación $\omega \in H\Omega$ tal que $\omega' = \mu_\Omega(\omega)$. Y como $\omega'' = \mu'_\Omega(\omega')$, obtenemos una operación $\omega \in H\Omega$ tal que $\omega'' = \mu'_\Omega(\mu_\Omega(\omega))$.

Para completar la demostración citar que la composición de morfismos de signaturas ocultos verifica la propiedad asociativa y que el morfismo identidad se comporta como tal (la composición de morfismos de signaturas ocultos no es otra cosa que la composición de las aplicaciones conjuntistas subyacentes). ∎

Los modelos que se asocian a una signatura oculta son álgebras cuya restricción a la signatura visible es exactamente el dominio de datos D asociado a esa signatura. Estos modelos se llaman álgebras ocultas. Un homomorfismo entre álgebras ocultas debe fijar este dominio de datos.

Definición 2.4.6 (Álgebra oculta). Sea $H\Sigma$ una signatura oculta definida sobre una signatura visible $V\Sigma$ y un dominio de datos D. Un *álgebra oculta* A para $H\Sigma$ es una $H\Sigma$-álgebra tal que $A|_{V\Sigma} = D$.

Definición 2.4.7 (Homomorfismo oculto). Sea $H\Sigma$ una signatura oculta definida sobre $V\Sigma$ y D, y sean A, B dos $H\Sigma$-álgebras ocultas. Un *homomorfismo oculto* entre A, B es un $H\Sigma$-homomorfismo $f \colon A \to B$ tal que $f_s = id_{D_s}$ para cada género $s \in V\Sigma$.

Las álgebras ocultas para una signatura oculta $H\Sigma$ sobre $V\Sigma$ y D, junto con los homomorfismos ocultos definen una categoría, que denotaremos $HAlg^D(H\Sigma)$.

A continuación definimos la noción de *satisfacción de comportamiento* ("behavioural satisfaction") que formaliza cuándo un álgebra oculta verifica una ecuación. De manera intuitiva, consiste en comprobar si los dos términos que forman la ecuación "se comportan igual" bajo todos los posibles "experimentos" adecuados a ellos. La idea de experimento se concreta en el concepto de *contexto*. Un contexto consiste en un término de género visible con una única variable.

Definición 2.4.8 (Contexto). Sea $H\Sigma = (S, \Omega)$ una signatura oculta sobre $V\Sigma = (VS, V\Omega)$ y D, y sea $s \in S$ un género. Un *$H\Sigma$-contexto* c para el género s es un $H\Sigma$-término de género visible $v \in VS$ que contiene una única variable z, que aparece una única vez y que es de género s, es decir, $c \in T_{H\Sigma(\{z\}),v}$.

En ocasiones escribimos un contexto c como $c(z)$ para indicar la única variable del término c o bien como $c(z_s)$ con s el género de z. Denotamos por $C_{H\Sigma}[z_s]_v$ al conjunto de todos los $H\Sigma$-contextos para el género s de género visible v con variable z_s, y $C_{H\Sigma}[z_s]$ al conjunto de todos los $H\Sigma$-contextos para el género s de cualquier género visible y con variable en z_s.

Los contextos nos permiten "observar" términos, pensando en la variable como un "hueco" en el que colocar los términos a observar. Un contexto es adecuado para observar aquellos términos cuyo género coincida con el género de la única variable del contexto.

Definición 2.4.9 (Contexto apropiado para un término). Un contexto $c(z)$ es *apropiado* para un término t si el género del término t coincide con el género de la variable z. Denotamos $c(t)$ al resultado de sustituir t por z en el contexto c.

Una *observación* de un término es la sustitución, en un contexto apropiado, de la variable del contexto por el término. Obtenemos así un término de género visible que dará lugar a un dato del dominio de datos. A través de sus observaciones obtenemos el *comportamiento* de un término. Surge de esta forma la satisfacción de comportamiento de una ecuación en la que los dos términos que la forman deben comportarse igual para todos los contextos. Debemos remarcar que para una signatura oculta consideraremos las mismas ecuaciones que le corresponderían como signatura de la especificación algebraica ecuacional.

Definición 2.4.10 (Satisfacción de comportamiento). Sea $H\Sigma = (S, \Omega)$ una signatura oculta sobre $V\Sigma = (VS, V\Omega)$ y D. Una $H\Sigma$-álgebra oculta A *satisface a través de su comportamiento* una $H\Sigma$-ecuación (X, t, u) si y sólo si para cada $H\Sigma$-contexto apropiado c para los términos t y u, se verifica que $A \models^{\mathcal{E}}_{H\Sigma} (X, c(t), c(u))$, donde $\models^{\mathcal{E}}$ es la relación de satisfacibilidad de la especificación algebraica ecuacional. En este caso escribimos $A \models\!\!\!\equiv_{H\Sigma} (X, t, u)$, y eliminaremos la referencia a la signatura oculta si no hay lugar a confusión.

La siguiente nota contiene una consecuencia inmediata de la definición anterior y jugará un papel importante en algunos de los resultados que se obtienen posteriormente.

Nota 2.4.11. Para las ecuaciones visibles, es decir, ecuaciones cuyos términos son de género visible, no existe diferencia entre la satisfacción ecuacional y la satisfacción de comportamiento.

En [42], Goguen y Malcolm presentan un álgebra final para la categoría $HAlg^D(H\Sigma)$ en el caso de ser $H\Sigma$ una signatura oculta que no presente constructores a partir de datos. Esta álgebra, que denotamos $F^{H\Sigma}$, se define de forma explícita a través las siguientes "fórmulas mágicas", según las califican sus propios autores.

A cada género oculto $h \in HS$ se le asocia el soporte

$$F^{H\Sigma}_h = \prod_{v \in VS} [C_{H\Sigma}[z_h]_v \to D_v],$$

es decir, cada $f \in F_h^{H\Sigma}$ es una tupla de funciones indexada por el conjunto de géneros visibles.

Para cada operación deconstructora $\omega\colon h\ v_1 \ldots v_n \to v$, $n \geq 0$, con $h \in HS$ y $v_i, v \in VS$, $i = 1, \ldots, n$, la interpretación de la operación $\omega_{F^{H\Sigma}}\colon F_h^{H\Sigma} \times D_{v_1} \times \cdots \times D_{v_n} \to D_v$ viene dada por

$$\omega_{F^{H\Sigma}}(f, d_1, \ldots, d_n) := f_v(\omega(z_h, d_1, \ldots, d_n))$$

para cada $f \in F_h^{H\Sigma}$ y cada $d_i \in D_i$, $i = 1, \ldots, n$.

Por último, para cada operación constructora $\omega\colon h\ v_1 \ldots v_n \to h'$, con $h, h' \in HS$, $v_i \in VS$, $i = 1, \ldots, n$ se define la función $\omega_{F^{H\Sigma}}\colon F_h^{H\Sigma} \times D_{v_1} \times \cdots \times D_{v_n} \to F_{h'}^{H\Sigma}$ tal que para cada $f \in F_h^{H\Sigma}$, $d_i \in D_i$, $i = 1, \ldots, n$, $\omega_{F^{H\Sigma}}(f, d_1, \ldots, d_n)$ viene dada por la tupla $(k_v\colon C_{H\Sigma}[z_{h'}]_v \to D_v)_{v \in VS}$, tal que para cada $v \in VS$ y cada $c \in C_{H\Sigma}[z_{h'}]_v$ está definida por

$$k_v(c) := f_v(c[\omega(z_h, d_1, \ldots, d_n)]).$$

Recogemos en el siguiente teorema la condición de ser objeto final.

Teorema 2.4.12. *Si $H\Sigma$ es una signatura oculta sin constructores desde datos, entonces la $H\Sigma$-álgebra $F^{H\Sigma}$ es un objeto final en la categoría $HAlg^D(H\Sigma)$.*

No obstante, para poder completar la demostración de que este objeto es final debemos trabajar no con el álgebra de términos de la signatura visible sino con un cociente suyo en el que se identifique cada término visible con la constante obtenida por su interpretación en el álgebra visible. Este hecho, que se ha puesto de manifiesto por ejemplo en [72], no se tuvo en cuenta en [42].

Para terminar esta sección presentamos el siguiente resultado que hemos extraído de [41] que permite establecer una institución para las álgebras ocultas.

Teorema 2.4.13. *Sean $H\Sigma$ y $H\Sigma'$ dos signaturas ocultas definidas sobre la misma signatura y álgebra visibles $V\Sigma$ y D, y sea $\mu\colon H\Sigma \to H\Sigma'$ un morfismo de signaturas oculto entre ellas. Entonces,*

$$\mu(A') \models\!\!\equiv_{H\Sigma} e \text{ si y sólo si } A' \models\!\!\equiv_{H\Sigma'} \mu(e)$$

para toda $H\Sigma'$-álgebra oculta A' y toda $H\Sigma$-ecuación e, donde $\mu(A')$ representa el reducto de A' en $H\Sigma$ y $\mu(e)$ representa la traslación de la ecuación e por el morfismo μ, justo como en la especificación algebraica ecuacional.

2.4.2 Una institución oculta $\mathcal{H}^D$

Definimos a continuación una institución para las álgebras ocultas que es ligeramente más general que la institución oculta de [41] y [15]. En particular, la categoría de signaturas presente en la institución oculta de estos artículos está definida de modo que todas las signaturas de la institución comparten la misma signatura visible y dominio de datos. Nosotros fijamos un universo de datos y permitimos signaturas sobre distintas álgebras visibles, cuyos conjuntos de datos están elegidos entre los conjuntos prefijados en ese universo de datos.

Definición 2.4.14 (Institución ecuacional oculta definida sobre un universo de datos D). Consideramos un conjunto U fijado que llamamos *universo de géneros*. Entonces, para cada $s \in U$ fijamos también un conjunto D_s (no vacío). La familia $D = \{D_s\}_{s \in U}$ se llama *universo de datos*. La *institución ecuacional oculta definida sobre el universo de datos* D, que denotamos por $\mathcal{H}^D = (SIG_{\mathcal{H}^D}, Sen_{\mathcal{H}^D}, Mod_{\mathcal{H}^D}, \models^{\mathcal{H}^D})$, es la institución cuyas componentes son:

1. La categoría $SIG_{\mathcal{H}^D}$ es la categoría de signaturas ocultas que hemos definido en la sección anterior.

2. El funtor $Sen_{\mathcal{H}^D} \colon SIG_{\mathcal{H}^D} \to Set$ asocia a cada signatura oculta $H\Sigma$ definida sobre $V\Sigma$ y D, el conjunto de ecuaciones $Sen_{\mathcal{E}}(H\Sigma)$, y a cada morfismo de signaturas oculto $\mu \colon H\Sigma \to H\Sigma'$ la aplicación $Sen_{\mathcal{E}}(\mu)$. Es decir, el conjunto de sentencias que se asocia a cada signatura oculta $H\Sigma$ es el mismo conjunto de ecuaciones que corresponde a $H\Sigma$ en la institución algebraica ecuacional. De la misma forma, cada morfismo de signaturas oculto da lugar a una aplicación entre los conjuntos de ecuaciones, ya que un morfismo de signaturas ocultas es un caso particular de morfismo de signaturas.

3. El funtor $Mod_{\mathcal{H}^D} \colon SIG_{\mathcal{H}^D} \to Cat^{op}$ asocia a cada signatura oculta $H\Sigma$ definida sobre $V\Sigma$ y D la categoría $HAlg^D(H\Sigma)$, y a cada morfismo de signaturas $\mu \colon H\Sigma \to H\Sigma'$ el funtor $Mod_{\mathcal{H}^D}(\mu) \colon HAlg^D(H\Sigma') \to HAlg^D(H\Sigma)$ que se define a través del reducto.

4. Para cada signatura oculta $H\Sigma$ sobre $V\Sigma$ y D, la relación de satisfacción $\models^{\mathcal{H}^D}_{H\Sigma}$ es la satisfacción de comportamiento $\models\!\!\!\equiv_{H\Sigma}$.

La institución anterior está bien definida a partir del Teorema 2.4.13, ya que los morfismos de $SIG_{\mathcal{H}^D}$ son siempre morfismos de signaturas definidos entre dos signaturas construidas sobre la misma signatura y álgebra visible.

2.4.3 Un morfismo de instituciones unidireccional entre las instituciones $\mathcal{H}^D$ y $\mathcal{E}$

En [15], Burstall y Diaconescu definen un morfismo de instituciones entre una institución oculta y la institución ecuacional. Este morfismo de instituciones se obtiene como corolario de un resultado general que trabaja con la construcción abstracta de una institución oculta asociada a una institución que verifique una serie de condiciones. En esta sección nos apoyaremos en dicho morfismo para definir un morfismo de instituciones unidireccional entre $\mathcal{H}^D$ y $\mathcal{E}$.

Dadas la institución ecuacional oculta sobre un universo de datos D, a la que hemos denotando por $\mathcal{H}^D = (SIG_{\mathcal{H}^D}, Sen_{\mathcal{H}^D}, Mod_{\mathcal{H}^D}, \models^{\mathcal{H}^D})$ y la institución algebraica ecuacional, $\mathcal{E} = (SIG_{\mathcal{E}}, Sen_{\mathcal{E}}, Mod_{\mathcal{E}}, \models^{\mathcal{E}})$, definimos a continuación un morfismo de instituciones unidireccional entre ambas.

Como primera componente del morfismo de instituciones unidireccional, consideramos el funtor olvido entre las categorías de signaturas $i\colon SIG_{\mathcal{H}^D} \to SIG_{\mathcal{E}}$. Este funtor lleva una signatura oculta $H\Sigma$, definida sobre $V\Sigma$ y D, sobre la signatura $H\Sigma$ en la que no se tiene en cuenta la distinción entre géneros ocultos y géneros visibles (ni tampoco entre operaciones ocultas y operaciones visibles). De la misma forma actúa sobre los morfismos de signaturas ocultos, olvidando la distinción entre la parte oculta y la parte visible de los mismos.

Como segunda componente consideramos la transformación natural $id\colon Sen_{\mathcal{H}^D} \Rightarrow Sen_{\mathcal{E}} \circ i$ entre los funtores sentencias $Sen_{\mathcal{H}^D}\colon SIG_{\mathcal{H}^D} \to Set$ y $Sen_{\mathcal{E}} \circ i\colon SIG_{\mathcal{H}^D} \to Set$ dada por las aplicaciones identidad $id_{H\Sigma}\colon Sen_{\mathcal{H}^D}(H\Sigma) \to Sen_{\mathcal{E}}(i(H\Sigma))$, una por cada signatura oculta $H\Sigma$ definida sobre $V\Sigma$ y D de $SIG_{\mathcal{H}^D}$. Observamos que esta trasformación natural está bien definida ya que tenemos que $Sen_{\mathcal{H}^D}(H\Sigma) = Sen_{\mathcal{E}}(i(H\Sigma))$.

Como última componente del morfismo de instituciones unidireccional tenemos que definir una transformación natural $\beta\colon Mod_{\mathcal{H}^D} \Rightarrow Mod_{\mathcal{E}} \circ i$ entre los funtores modelos de ambas instituciones. Esta trasformación natural vendrá dada por una familia de funtores (morfismos en Cat), $\beta_{H\Sigma}\colon Mod_{\mathcal{H}^D}(H\Sigma) \to Mod_{\mathcal{E}}(i(H\Sigma))$, uno por cada signatura oculta $H\Sigma$ definida sobre $V\Sigma$ y D de $SIG_{\mathcal{H}^D}$. La definición de esta familia de funtores se apoya en el concepto de datos equivalentes a través de su comportamiento [41].

Definición 2.4.15 (Equivalencia de comportamiento). Sea $H\Sigma = (S, \Omega)$ una signatura oculta definida sobre $V\Sigma$ y D, y sea A una $H\Sigma$-álgebra oculta. Decimos que dos elementos $a, a' \in A_s$, con $s \in S$, son *equivalentes a través de su comportamiento* si y sólo si $\overline{q_a}(c(z)) = \overline{q_{a'}}(c(z))$ para cualquier $H\Sigma$-contexto $c(z)$ para el género s, donde $q_x\colon \{z\} \to A$, con $x \in A_s$, es la valoración definida por $q_x(z) = x$. Denotaremos que $a, a' \in A_s$ son equivalentes a través de su comportamiento por $a \equiv^A_{H\Sigma,s} a'$. También podemos utilizar $a \equiv_{H\Sigma,s} a'$, o incluso $a \equiv_{H\Sigma} a'$ si no hay lugar a confusión.

Los dos siguientes resultados han sido extraídos de [43]. El primero marca la estrecha relación que existe entre las nociones de equivalencia de comportamiento y satisfacción de comportamiento. En el segundo se establece que la equivalencia de comportamiento es una relación de congruencia.

Lema 2.4.16. *Sean $H\Sigma = (S, \Omega)$ una signatura oculta definida sobre $V\Sigma$ y D, A una $H\Sigma$-álgebra oculta y (X, t, u) una $H\Sigma$-ecuación. Entonces,*

$$A \models_{H\Sigma} (X, t, u) \text{ si y sólo si } \overline{p}(t) \equiv_{H\Sigma} \overline{p}(u), \text{ para toda valoración } p : X \to A.$$

Proposición 2.4.17. *Sean $H\Sigma = (S, \Omega)$ una signatura oculta definida sobre $V\Sigma$ y D, y A una $H\Sigma$-álgebra oculta. Entonces, la familia $\equiv_{H\Sigma} = \{\equiv_{H\Sigma,s}\}_{s\in S}$ es una relación de congruencia en A*

Esta relación de congruencia nos va a permitir definir un álgebra cociente a partir de un álgebra oculta. Utilizaremos esta construcción para definir la familia de funtores que den lugar a una transformación natural entre los correspondientes funtores de modelos de la institución oculta y la institución algebraica ecuacional.

Dada una signatura oculta $H\Sigma = (S, \Omega)$ definida sobre $V\Sigma$ y D, definimos un funtor $\beta_{H\Sigma}\colon Mod_{\mathcal{H}^D}(H\Sigma) \to Mod_{\mathcal{E}}(i(H\Sigma))$. Para cada álgebra $A \in Obj(Mod_{\mathcal{H}^D}(H\Sigma))$, definimos $\beta_{H\Sigma}(A) := A/_{\equiv_{H\Sigma}}$, que es el álgebra cociente de A por la relación de congruencia anterior. Para cada homomorfismo de álgebras $h = (h_s)_{s\in S}\colon A \to B$ de $Mod_{\mathcal{H}^D}(H\Sigma)$, definimos $\beta_{H\Sigma}(h) := h/_{\equiv_{H\Sigma}}$, que es el $H\Sigma$-homomorfismo definido sobre los cocientes a través de h, es decir $h/_{\equiv_{H\Sigma}}$ es la familia de aplicaciones $h/_{\equiv_{H\Sigma,s}}\colon A/_{\equiv_{H\Sigma,s}} \to B/_{\equiv_{H\Sigma,s}}$, una para cada género $s \in S$, dada por: $h/_{\equiv_{H\Sigma,s}}([a]_{\equiv_{H\Sigma,s}}) = [h_s(a)]_{\equiv_{H\Sigma,s}}$, para cada $[a]_{\equiv_{H\Sigma,s}} \in A/_{\equiv_{H\Sigma,s}}$.

El siguiente lema que hemos extraído nuevamente de [43] garantiza que el homomorfismo $h/_{\equiv_{H\Sigma}}$ está bien definido ya que verifica las hipótesis de la Proposición 1.4.13.

Lema 2.4.18. *Sean $H\Sigma = (S, \Omega)$ una signatura oculta, $f\colon A \to B$ un $H\Sigma$-homomorfismo oculto y $a, a' \in A_s$, con $s \in S$. Entonces,*

$$a \equiv_{H\Sigma,s} a' \text{ en } A \text{ si y sólo si } f(a) \equiv_{H\Sigma,s} f(a') \text{ en } B.$$

La demostración de que $\beta\colon Mod_{\mathcal{H}^D} \Rightarrow Mod_{\mathcal{E}} \circ i$ es una transformación natural puede obtenerse como corolario de un resultado más potente en [15]. No obstante, ofrecemos a continuación una demostración directa y mucho más sencilla. Esta demostración se apoya en la siguiente proposición.

Proposición 2.4.19. *Sean $H\Sigma = (S, \Omega)$ y $H\Sigma' = (S', \Omega')$ dos signaturas ocultas definidas sobre $V\Sigma$ y D, $\mu\colon H\Sigma \to H\Sigma'$ un morfismo de signaturas oculto y A' una $H\Sigma'$-álgebra oculta. Entonces,*

$$a \equiv_{H\Sigma,s}^{Mod_{\mathcal{H}^D}(\mu)(A')} b \text{ si y sólo si } a \equiv_{H\Sigma',\mu(s)}^{A'} b$$

para cada $s \in S$ y $a, b \in A'_{\mu(s)}$.

Demostración. Si $s \in S$ es un género visible, entonces $\mu(s)$ es un género visible y obtenemos de forma inmediata la doble implicación ya que es fácil comprobar que la equivalencia de comportamiento coincide con la igualdad en los géneros visibles (ver [43]).

Suponemos que $s \in S$ es un género oculto. Entonces si aplicamos la definición de equivalencia de comportamiento obtenemos que lo que debemos demostrar es

$$\overline{q_a}(c(z)) = \overline{q_b}(c(z)) \text{ si y sólo si } \overline{q'_a}(c'(z')) = \overline{q'_b}(c'(z'))$$

donde el lado izquierdo está cuantificado por todo $H\Sigma$-contexto $c(z)$ para el género s y el derecho por todo $H\Sigma'$-contexto $c'(z')$ para el género $\mu(s)$, donde $q_x\colon \{z\} \to$

$Mod_{\mathcal{H}^D}(\mu)(A')$ y $q'_{x'}\colon \{z'\} \to A'$ son las valoraciones definidas por $q_x(z) = x$ y $q'_{x'}(z') = x'$ respectivamente, con $x \in Mod_{\mathcal{H}^D}(\mu)(A')_s$ y $x' \in A'_{\mu(s)}$.

Nos centramos primero en la implicación inversa.

Sea $c(z)$ un $H\Sigma$-contexto para el género s. Entonces $\mu(c(z))$ es un $H\Sigma'$-contexto para el género $\mu(s)$. Por hipótesis se verifica entonces que

$$\overline{q'_a}(\mu(c(z))) = \overline{q'_b}(\mu(c(z)))$$

con $q'_a\colon \{z\} \to A'$ la valoración definida por $q'_a(z) = a$ y $q'_b\colon \{z\} \to A'$ la valoración definida por $q'_b(z) = b$. Ahora es fácil obtener por inducción sobre la estructura de un término c que

$$\overline{q'_a}(\mu(c(z))) = \overline{q_a}(c(z))$$

con $q_a\colon \{z\} \to Mod_{\mathcal{H}^D}(\mu)(A')$ la valoración definida por $q_a(z) = a$, lo que demuestra esta implicación.

Demostraremos a continuación la implicación directa.

Sea $c'(z')$ un $H\Sigma'$-contexto para el género $\mu(s)$, es decir, c' es un $H\Sigma'$-término visible teniendo una única ocurrencia de una variable z' de género $\mu(s)$. Entonces c' debe contener entre sus subtérminos a un término $\omega'(z', t'_1, \ldots, t'_n)$ con $\omega'\colon s's'_1 \ldots s'_n \to r'$, $n \geq 0$ donde $s' = \mu(s)$, s'_i es género visible y t'_i es un $H\Sigma'$-término de género s_i sin variables, $\forall i = 1, \ldots, n$. Por definición de morfismo de signaturas oculto, existe una operación $\omega\colon ss_1 \ldots s_n \to r \in H\Omega$ tal que $\omega' = \mu(\omega)$.

Suponemos en un primer momento que $c' = \omega'(z', t'_1, \ldots, t'_n)$. Debemos probar que

$$\overline{q'_a}(\omega'(z', t'_1, \ldots, t'_n)) = \overline{q'_b}(\omega'(z', t'_1, \ldots, t'_n)),$$

donde $q'_a\colon \{z'\} \to A'$ es la valoración definida por $q'_a(z') = a$ y $q'_b\colon \{z'\} \to A'$ es la valoración definida por $q'_b(z') = b$.

Tenemos que $\overline{q'_a}(\omega'(z', t'_1, \ldots, t'_n)) = \omega'_{A'}(a, d_{t'_1}, \ldots, d_{t'_n})$, donde $\overline{q'_a}(t'_i) = d_{t'_i}$ es una constante visible, ya que t'_i es un término visible, $\forall i = 1, \ldots, n$, (en el caso $n = 0$, no aparece ningún término visible, y por lo tanto ninguna constante visible, en la expresión anterior). Ahora, como $\omega' = \mu(\omega)$ y $Mod_{\mathcal{H}^D}(\mu)(A')$ es el μ reducto de A', tenemos que $\omega'_{A'}(a, d_{t'_1}, \ldots, d_{t'_n}) = \omega_{Mod_{\mathcal{H}^D}(\mu)(A')}(a, d_{t'_1}, \ldots, d_{t'_n})$. Esta constante es igual a $\overline{q_a}(\omega(z, d_{t'_1}, \ldots, d_{t'_n}))$, donde $q_a\colon \{z\} \to Mod_{\mathcal{H}^D}(\mu)(A')$ es la valoración definida por $q_a(z) = a$.

Si consideramos ahora el $H\Sigma$-contexto $\omega(z, d_{t'_1}, \ldots, d_{t'_n})$ para el género s, tenemos por hipótesis que

$$\overline{q_a}(\omega(z, d_{t'_1}, \ldots, d_{t'_n})) = \overline{q_b}(\omega(z, d_{t'_1}, \ldots, d_{t'_n})),$$

lo que prueba la implicación para este primer caso.

Consideramos en segundo lugar un contexto general $c' = \omega''(\omega'(z', t'_1, \ldots, t'_n))$, que contiene como subtérmino a $\omega'(z', t'_1, \ldots, t'_n)$. Éste es el único subtérmino de c' que contiene una variable. Debemos probar que

$$\overline{q'_a}(\omega''(\omega'(z', t'_1, \ldots, t'_n))) = \overline{q'_b}(\omega''(\omega'(z', t'_1, \ldots, t'_n))),$$

donde $q'_a\colon \{z'\} \to A'$ es la valoración definida por $q'_a(z') = a$ y $q'_b\colon \{z'\} \to A'$ es la valoración definida por $q'_b(z') = b$. Entonces,

$$\begin{aligned}
&\overline{q'_a}(\omega''(\omega'(z', t'_1, \ldots, t'_n))) = \\
&= \omega''_{A'}(\overline{q'_a}(\omega'(z', t'_1, \ldots, t'_n)) \text{ ya que } \omega'' \text{ no tiene variables,} \\
&= \omega''_{A'}(\overline{q'_b}(\omega'(z', t'_1, \ldots, t'_n)) \text{ por la igualdad probada antes,} \\
&= \overline{q'_b}(\omega''(\omega'(z', t'_1, \ldots, t'_n))).
\end{aligned}$$

■

En el siguiente resultado recogemos que cada $\beta_{H\Sigma}$ es un funtor. Es una simple comprobación el demostrar que respeta la identidad y la composición de morfismos de $Mod_{\mathcal{H}^D}(H\Sigma)$.

Lema 2.4.20. *Sea $H\Sigma$ una signatura oculta definida sobre $V\Sigma$ y D. La aplicación $\beta_{H\Sigma}\colon Mod_{\mathcal{H}^D}(H\Sigma) \to Mod_{\mathcal{E}}(i(H\Sigma))$ es un funtor.*

Por fin, demostramos que β define una transformación natural.

Lema 2.4.21. *La familia de funtores $\beta\colon Mod_{\mathcal{H}^D} \Rightarrow Mod_{\mathcal{E}} \circ i$ definen una transformación natural.*

Demostración. Falta por comprobar que la familia de funtores verifica la propiedad de naturalidad, es decir, que para cada morfismo de signaturas oculto $\mu\colon H\Sigma \to H\Sigma'$ de $SIG_{\mathcal{H}^D}$, entre las signaturas ocultas $H\Sigma = (S, \Omega)$ y $H\Sigma'$, es conmutativo el siguiente diagrama:

$$\begin{array}{ccc}
Mod_{\mathcal{H}^D}(H\Sigma) & \xrightarrow{\beta_{H\Sigma}} & Mod_{\mathcal{E}}(i(H\Sigma)) \\
{\scriptstyle Mod_{\mathcal{H}^D}(\mu)}\uparrow & & \uparrow{\scriptstyle Mod_{\mathcal{E}}(i(\mu))} \\
Mod_{\mathcal{H}^D}(H\Sigma') & \xrightarrow[\beta_{H\Sigma'}]{} & Mod_{\mathcal{E}}(i(H\Sigma'))
\end{array}$$

Primero probamos que el diagrama conmuta sobre los objetos de $Mod_{\mathcal{H}^D}(H\Sigma')$. Es decir, si para cada $H\Sigma'$-álgebra oculta A' de $Mod_{\mathcal{H}^D}(H\Sigma')$ se verifica que

$$Mod_{\mathcal{E}}(i(\mu)) \circ \beta_{H\Sigma'}(A') = \beta_{H\Sigma} \circ Mod_{\mathcal{H}^D}(\mu)(A').$$

Por un lado tenemos que, por definición de $\beta_{H\Sigma'}$, $Mod_{\mathcal{E}}(i(\mu))(\beta_{H\Sigma'}(A'))$ es igual a $Mod_{\mathcal{E}}(i(\mu))(A'/_{\equiv_{H\Sigma'}})$. Esta $H\Sigma$-álgebra está definida por:

- para cada género $s \in S$, el conjunto $Mod_{\mathcal{E}}(i(\mu))(A'/_{\equiv_{H\Sigma'}})_s = A'/_{\equiv_{H\Sigma',\mu(s)}}$, por definición de reducto. Ahora, por definición del cociente, este conjunto es igual a $\{[a]_{\equiv_{H\Sigma',\mu(s)}} \mid a \in A'_{\mu(s)}\}$,

- para cada operación $\omega\colon s_1 \dots s_n \rightarrow s \in \Omega$, $n \geq 0$, la función definida por $\omega_{Mod_{\mathcal{E}}(i(\mu))(A'/_{\equiv_{H\Sigma'}})}([a_1]_{\equiv_{H\Sigma',\mu(s_1)}}, \dots, [a_n]_{\equiv_{H\Sigma',\mu(s_n)}}) = \mu(\omega)_{A'/_{\equiv_{H\Sigma'}}}([a_1]_{\equiv_{H\Sigma',\mu(s_1)}}, \dots, [a_n]_{\equiv_{H\Sigma',\mu(s_n)}})$, para cada $[a_i]_{\equiv_{H\Sigma',\mu(s_i)}} \in A'/_{\equiv_{H\Sigma',\mu(s_i)}}$, $i = 1, \dots, n$, por definición de reducto. Nuevamente, por definición del cociente, esta función es igual a $[\mu(\omega)_{A'}(a_1, \dots, a_n)]_{\equiv_{H\Sigma',\mu(s)}}$, para cada $[a_i]_{\equiv_{H\Sigma',\mu(s_i)}} \in A'/_{\equiv_{H\Sigma',\mu(s_i)}}$, $i = 1, \dots, n$.

Por otro lado, por definición de $\beta_{H\Sigma}$, tenemos que $\beta_{H\Sigma}(Mod_{\mathcal{H}^D}(\mu)(A'))$ es igual a $Mod_{\mathcal{H}^D}(\mu)(A')/_{\equiv_{H\Sigma}}$. Esta $H\Sigma$-álgebra está definida por:

- para cada género $s \in S$, el conjunto $Mod_{\mathcal{H}^D}(\mu)(A')/_{\equiv_{H\Sigma,s}} = \{[a]_{\equiv_{H\Sigma,s}} \mid a \in Mod_{\mathcal{H}^D}(\mu)(A')_s\}$, por definición del cociente,
- para cada operación $\omega\colon s_1 \dots s_n \rightarrow s \in \Omega$, $n \geq 0$, la función definida por $\omega_{Mod_{\mathcal{H}^D}(\mu)(A')/_{\equiv_{H\Sigma}}}([a_1]_{\equiv_{H\Sigma,s_1}}, \dots, [a_n]_{\equiv_{H\Sigma,s_n}}) = [\omega_{Mod_{\mathcal{H}^D}(\mu)(A')}(a_1, \dots, a_n)]_{\equiv_{H\Sigma,s}}$, para cada $[a_i]_{\equiv_{H\Sigma,s_i}} \in Mod_{\mathcal{H}^D}(\mu)(A')/_{\equiv_{H\Sigma,s_i}}$, $i = 1, \dots, n$, por definición del cociente. Ahora, por definición de reducto esta función es igual a la función $[\mu(\omega)_{A'}(a_1, \dots, a_n)]_{\equiv_{H\Sigma,s}}$, para cada $[a_i]_{\equiv_{H\Sigma,s_i}} \in Mod_{\mathcal{H}^D}(\mu)(A')/_{\equiv_{H\Sigma,s_i}}$, $i = 1, \dots, n$.

Observamos entonces que para demostrar la igualdad anterior es suficiente probar que

$$\{[a]_{\equiv_{H\Sigma',\mu(s)}} \mid a \in A'_{\mu(s)}\} = \{[a]_{\equiv_{H\Sigma,s}} \mid a \in Mod_{\mathcal{H}^D}(\mu)(A')_s\},$$

para cada género $s \in S$. Esta igualdad se obtiene de forma inmediata a partir de la Proposición 2.4.19.

Una vez que hemos demostrado que el diagrama conmuta para los objetos de $Mod_{\mathcal{H}^D}(H\Sigma')$, abordamos la demostración de que el diagrama también conmuta para sus morfismos. Es decir, tenemos que probar que para cada $H\Sigma'$-homomorfismo oculto $h'\colon A' \rightarrow B'$ de $Mod_{\mathcal{H}^D}(H\Sigma')$ se verifica que

$$Mod_{\mathcal{E}}(i(\mu)) \circ \beta_{H\Sigma'}(h') = \beta_{H\Sigma} \circ Mod_{\mathcal{H}^D}(\mu)(h').$$

Por un lado tenemos que $Mod_{\mathcal{E}}(i(\mu))(\beta_{H\Sigma'}(h'))$ es el $H\Sigma$-homomorfismo dado por la familia de aplicaciones $Mod_{\mathcal{E}}(i(\mu))(h'/_{\equiv_{H\Sigma'}})_s = (h'/_{\equiv_{H\Sigma'}})_{\mu(s)}\colon A'/_{\equiv_{H\Sigma',\mu(s)}} \rightarrow B'/_{\equiv_{H\Sigma',\mu(s)}}$, una para cada género $s \in S$, tal que $(h'/_{\equiv_{H\Sigma'}})_{\mu(s)}([a]_{\equiv_{H\Sigma',\mu(s)}}) = [h'_{\mu(s)}(a)]_{\equiv_{H\Sigma',\mu(s)}}$, para todo $[a]_{\equiv_{H\Sigma',\mu(s)}} \in A'/_{\equiv_{H\Sigma',\mu(s)}}$.

Por otro lado tenemos que $\beta_{H\Sigma}(Mod_{\mathcal{H}^D}(\mu)(h'))$ es el $H\Sigma$-homomorfismo dado por la familia de aplicaciones $Mod_{\mathcal{H}^D}(\mu)(h')/_{\equiv_{H\Sigma,s}}\colon Mod_{\mathcal{H}^D}(\mu)(A')/_{\equiv_{H\Sigma,s}} \rightarrow Mod_{\mathcal{H}^D}(\mu)(B')/_{\equiv_{H\Sigma,s}}$, una para cada género $s \in S$, tal que $Mod_{\mathcal{H}^D}(\mu)(h')/_{\equiv_{H\Sigma,s}}([a]_{\equiv_{H\Sigma,s}}) = [Mod_{\mathcal{H}^D}(\mu)(h')_s(a)]_{\equiv_{H\Sigma,s}}$, para todo $[a]_{\equiv_{H\Sigma,s}} \in Mod_{\mathcal{H}^D}(\mu)(A')/_{\equiv_{H\Sigma,s}}$

Obtenemos que ambos homomorfismos son iguales sin más que aplicar la Proposición 2.4.19 ya que por definición de reducto tenemos que $Mod_{\mathcal{H}^D}(\mu)(h')_s(a) = h'_{\mu(s)}(a)$. ■

Para demostrar que la terna de aplicaciones (i, id, β) que hemos definido anteriormente constituye un morfismo de instituciones unidireccional, únicamente falta por comprobar que se verifica la relación de satisfacibilidad. El siguiente teorema que puede encontrarse en [41] o en [43], expresa que esta relación de satisfacibilidad es compatible con las trasformaciones de sentencias y modelos. Su demostración es inmediata a partir del Lema 2.4.16 y del siguiente corolario que hemos extraído de [63].

Corolario 2.4.22. *Sean Σ una signatura, A una Σ-álgebra, Q una relación de congruencia en A y $t, u \in T_{\Sigma(X)}$ dos términos del mismo género. Entonces, $\overline{q}(t) = \overline{q}(u)$ para toda valoración $q\colon X \to A/_Q$ si y sólo si $[\overline{p}(t)]_Q = [\overline{p}(u)]_Q$ para toda valoración $p\colon X \to A$.*

Teorema 2.4.23. *Sea $H\Sigma$ una signatura oculta definida sobre $V\Sigma$ y D. Entonces,*

$$A \models\!\!\equiv_{H\Sigma} e \text{ si y sólo si } \beta_{H\Sigma}(A) \models_{H\Sigma} e$$

para cualquier una $H\Sigma$-álgebra oculta A y cualquier $H\Sigma$-ecuación e.

Recogemos el morfismo de instituciones unidireccional que hemos definido en el siguiente teorema.

Teorema 2.4.24. *La terna (i, id, β) define un morfismo de instituciones unidireccional entre $\mathcal{H}^D$ y $\mathcal{E}$.*

2.5 Un morfismo de instituciones unidireccional entre la institución $\mathcal{E}^D$ y la institución $\mathcal{H}^D$

Como hemos citado, un primer intento para tratar de modelar las estructuras de datos presentes en EAT condujo a definir una operación entre signaturas, que se ha extendido a modelos y sentencias. Esta operación ha dado lugar a un morfismo de instituciones unidireccional entre una nueva institución, que llamamos institución algebraica ecuacional definida sobre un universo de datos D, a la que hemos denotado por $\mathcal{E}^D$, y la institución algebraica ecuacional $\mathcal{E}$. La necesidad de fijar el universo de datos sobre el que definir los modelos viene originada por lo que se hace en el nivel de las implementaciones. En el sistema EAT, y en general en cualquier sistema que trabaje con estructuras algebraicas, es habitual implementar estas estructuras haciendo que sus elementos respondan a un mismo patrón sintáctico, un mismo modo de representación en el computador. Como comentamos anteriormente, cuando tratamos de modelizar estas estructuras, esta restricción se traduce en la necesidad de limitar las familias de álgebras a aquéllas en las que todas las álgebras de la familia compartan los mismos soportes. Esta condición nos llevaba a fijar un universo de datos D sobre el que definir los modelos.

En el mismo sentido, es claro que el rango de la institución de llegada, en el anterior morfismo de instituciones unidireccional, es también demasiado amplio: únicamente las

Σ_{imp}-álgebras basadas en D son relevantes en nuestro estudio. Este hecho es precisamente la característica fundamental de la *institución oculta* $\mathcal{H}^D$. En esta sección trataremos de redefinir el morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{E}$ a otro entre $\mathcal{E}^D$ y $\mathcal{H}^D$, que refleje nuestro problema de una forma más conveniente.

Consideramos de nuevo fijados un conjunto U que es el universo de géneros y una familia de conjuntos $D = \{D_s\}_{s\in U}$ que es el universo de datos. Estos conjuntos permanecerán fijados a lo largo del capítulo salvo que se indique lo contrario. Definimos a continuación un morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$.

Comenzamos definiendo un funtor entre las categorías de signaturas $\Phi^{\mathcal{E}^D-\mathcal{H}^D}\colon SIG_{\mathcal{E}^D} \to SIG_{\mathcal{H}^D}$. El funtor $\Phi^{\mathcal{E}^D-\mathcal{H}^D}$ asigna a cada signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, la signatura oculta $\Sigma_{imp} = (S_{imp}, \Omega_{imp} \cup C)$, signatura que definimos de la misma forma que $\Phi^{\mathcal{E}^D-\mathcal{E}}(\Sigma)$, sobre la signatura visible $V\Sigma_{imp} = (S, C)$ y el dominio de datos $D = (\{D_s\}_{s\in S}, C)$. Observamos entonces que el único género oculto en la signatura Σ_{imp} es el género destacado imp_Σ y que las únicas operaciones visibles son las constantes de C. Además, una característica fundamental de esta signatura es que todas las operaciones ocultas son deconstructoras.

Como en la institución $\mathcal{E}^D$ sólo hemos considerado morfismos identidad, la definición de Φ sobre morfismos resulta obvia, obteniendo el siguiente lema.

Lema 2.5.1. *La aplicación* $\Phi^{\mathcal{E}^D-\mathcal{H}^D}$ *es un funtor.*

A continuación definimos una transformación natural entre los funtores sentencias $\alpha^{\mathcal{E}^D-\mathcal{H}^D}\colon Sen_{\mathcal{E}^D} \Rightarrow Sen_{\mathcal{H}^D} \circ \Phi^{\mathcal{E}^D-\mathcal{H}^D}$. Esta transformación natural consistirá en una familia de aplicaciones, una aplicación $\alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}\colon Sen_{\mathcal{E}^D}(\Sigma) \to Sen_{\mathcal{H}^D}(\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma))$ para cada signatura Σ de $SIG_{\mathcal{E}^D}$.

Dada una signatura Σ de $SIG_{\mathcal{E}^D}$, observamos que el conjunto de sentencias $Sen_{\mathcal{H}^D}(\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma))$ es, por definición, el mismo conjunto de sentencias que $Sen_{\mathcal{E}}(\Phi^{\mathcal{E}^D-\mathcal{E}}(\Sigma))$. Aprovechando lo anterior, definimos la aplicación $\alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}\colon Sen_{\mathcal{E}^D}(\Sigma) \to Sen_{\mathcal{H}^D}(\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma))$ de la misma forma que la aplicación $\alpha_\Sigma^{\mathcal{E}^D-\mathcal{E}}\colon Sen_{\mathcal{E}^D}(\Sigma) \to Sen_{\mathcal{E}}(\Phi^{\mathcal{E}^D-\mathcal{E}}(\Sigma))$ definida anteriormente. Obtenemos trivialmente que estas aplicaciones constituyen una transformación natural, lo que recogemos en el siguiente lema.

Lema 2.5.2. *La familia de aplicaciones* $\alpha^{\mathcal{E}^D-\mathcal{H}^D}\colon Sen_{\mathcal{E}^D} \Rightarrow Sen_{\mathcal{H}^D} \circ \Phi^{\mathcal{E}^D-\mathcal{H}^D}$ *es una transformación natural.*

Una observación importante que se deduce de la definición de esta trasformación natural es que para cada signatura Σ de $SIG_{\mathcal{E}^D}$ y cada ecuación $e \in Sen_{\mathcal{E}^D}(\Sigma)$, la ecuación $\alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}(e)$ es de género visible. Esto es cierto ya que la signatura Σ_{imp} es *puramente deconstructora*, es decir, todas sus operaciones ocultas son deconstructoras. Entonces las únicas ecuaciones de género oculto en $Sen_{\mathcal{H}^D}(\Sigma_{imp})$ son las igualdades entre variables ocultas. No obstante, una ecuación $\alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}(e)$ no es en ningún caso una igualdad entre variables ocultas, por lo tanto es una ecuación visible.

Por último, definimos la tercera componente del morfismo de instituciones unidireccional que en este caso será una transformación natural $\beta^{\mathcal{E}^D-\mathcal{H}^D}\colon Mod_{\mathcal{E}^D} \Rightarrow Mod_{\mathcal{H}^D} \circ \Phi^{\mathcal{E}^D-\mathcal{H}^D}$. Esta trasformación natural consistirá en una familia de funtores $\beta_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}\colon Mod_{\mathcal{E}^D}(\Sigma) \to Mod_{\mathcal{H}^D}(\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma))$, un funtor para cada signatura Σ de $SIG_{\mathcal{E}^D}$.

Observamos que para cada Σ-álgebra A, podemos considerar $\beta_\Sigma^{\mathcal{E}^D-\mathcal{E}}(A)$ como una Σ_{imp}-álgebra oculta ya que, por definición, su restricción a la signatura visible $V\Sigma_{imp} = (S, C)$ es el álgebra visible $D = (\{D_s\}_{s\in S}, C)$. Además, para cada Σ-homomorfismo identidad $id_A\colon A \to A$, tenemos que $\beta_\Sigma^{\mathcal{E}^D-\mathcal{E}}(id_A)$ es un Σ_{imp}-homomorfismo oculto ya que, obviamente, es la identidad sobre la parte visible.

Así, dada una signatura Σ de $SIG_{\mathcal{E}^D}$, podemos definir $\beta_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}\colon Mod_{\mathcal{E}^D}(\Sigma) \to Mod_{\mathcal{H}^D}(\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma))$ sobre objetos y morfismos, de forma análoga al funtor $\beta_\Sigma^{\mathcal{E}^D-\mathcal{E}}\colon Mod_{\mathcal{E}^D}(\Sigma) \to Mod_{\mathcal{E}}(\Phi^{\mathcal{E}^D-\mathcal{E}}(\Sigma))$.

Nuevamente es trivial deducir que estos funtores constituyen una transformación natural, ya que $\beta^{\mathcal{E}^D-\mathcal{E}}$ es una trasformación natural.

Lema 2.5.3. *La familia de funtores $\beta^{\mathcal{E}^D-\mathcal{H}^D}\colon Mod_{\mathcal{E}^D} \Rightarrow Mod_{\mathcal{H}^D} \circ \Phi^{\mathcal{E}^D-\mathcal{H}^D}$ define una transformación natural.*

Para completar la definición de un morfismo de instituciones unidireccional falta por comprobar que la terna que acabamos de definir es compatible con la condición de satisfacibilidad. Es decir, tenemos que demostrar que para cada signatura Σ de $SIG_{\mathcal{E}^D}$ se verifica

$$A \models_\Sigma^{\mathcal{E}^D} e \quad \text{si y sólo si} \quad \beta_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}(A) \models\!\!\!\equiv^{\mathcal{H}^D}_{\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma)} \alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}(e),$$

para cualquier sentencia $e \in Sen_{\mathcal{E}^D}(\Sigma)$ y cualquier álgebra $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$.

Tenemos que para cualquier sentencia $e \in Sen_{\mathcal{E}^D}(\Sigma)$, la ecuación $\alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}(e)$ es visible. Una nota importante que destacábamos de la relación de satisfacción de comportamiento es que sobre ecuaciones visibles esta relación es, en realidad, la relación de satisfacción ecuacional. Entonces, para probar la condición de satisfacibilidad anterior es suficiente con demostrar que para cada signatura Σ de $SIG_{\mathcal{E}^D}$ se verifica

$$A \models_\Sigma^{\mathcal{E}^D} e \quad \text{si y sólo si} \quad \beta_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}(A) \models^{\mathcal{E}}_{\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma)} \alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}(e),$$

para cualquier sentencia $e \in Sen_{\mathcal{E}^D}(\Sigma)$ y cualquier álgebra $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$.

Por la propia definición de las tres componentes de este morfismo de instituciones unidireccional, obtenemos de modo inmediato que esta condición es cierta, a partir de la condición correspondiente para el morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{E}$.

El siguiente teorema establece el morfismo de instituciones unidireccional que acabamos de definir.

Teorema 2.5.4. *La terna (Φ, α, β) definida anteriormente constituye un morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$.*

2.5.1 Interpretación de la institución oculta en términos de implementaciones de estructuras de datos

Como ya se ha indicado previamente, en EAT conviven dos clases de estructuras de datos. Por un lado se manejan los datos básicos usuales en programación como números, listas, etc. Por otro lado aparecen estructuras de datos, cuyos elementos son esos datos básicos, que corresponden a estructuras matemáticas como grupos, anillos, complejos de cadenas, etc.

La idea que subyace es entender el universo de datos D considerado en las secciones anteriores como el *primer tipo* de estructura de datos: datos predefinidos y presentes en el sistema con el fin de ser utilizados en cualquier momento.

Entonces, una vez fijado el sustrato de datos D sobre el que trabajar, los modelos $Mod_{\mathcal{E}^D}(\Sigma)$ representan estructuras algebraicas definidas sobre tales datos. Éstas corresponden, por lo tanto, al *segundo tipo* de datos de datos presente en EAT. No obstante, en general, no se trata de trabajar con una única estructura matemática, un grupo por ejemplo, sino con familias de grupos (los cuales deben ser susceptibles de ser creados y manejados en tiempo de ejecución). Por lo tanto, para modelar EAT no es suficiente con tener un modelo para un único grupo, sino que debemos tratar de modelar *implementaciones* de grupos (es decir, la representación en la memoria de un computador de la información suficiente sobre un grupo). Entonces, las estructuras de datos codificadas en EAT son en realidad familias de implementaciones de modelos de $Mod_{\mathcal{E}^D}(\Sigma)$. Sin pretender entrar en cuestiones de implementación (en [58] se presenta un estudio detallado de la modelización de las implementaciones de estas estructuras de datos, apoyándose en una noción de implementación de un TAD derivada de la noción de implementación de Hoare [49]) y situándonos en el marco más simple de la especificación algebraica, obtenemos que debemos movernos de categorías de modelos en $\mathcal{E}^D$ hacia una categoría *indexada* (o fibrada) sobre ella (ver [97]). Esto es exactamente lo que ha sido formalmente expresado en la transferencia de la institución $\mathcal{E}^D$ a la institución $\mathcal{H}^D$ en la sección anterior, ya que, como vemos, los modelos en $Mod_{\mathcal{H}^D}(\Sigma_{imp})$ son familias de álgebras de $Mod_{\mathcal{E}^D}(\Sigma)$. Los elementos del conjunto soporte para el género oculto imp_Σ actúan como *índices* para dichas estructuras.

Debemos hacer notar que la idea de ver estas álgebras ocultas como familias de implementaciones de estructuras no es del todo cubierta por el morfismo de instituciones unidireccional que acabamos de definir, tal y como ocurre con el morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{E}$, ya que nuevamente los modelos en la imagen del morfismo representan familias formadas por una única álgebra. Sin embargo, en esta ocasión, estas imágenes pueden ser reunidas (por medio de coproductos) en una nueva álgebra, que corresponde directa y elegantemente con el modo de implementación usado en EAT. Además, esta álgebra resulta tener buenas propiedades desde el punto de vista de las categorías, ya que es el objeto final en la categoría de modelos $Mod_{\mathcal{H}^D}(\Sigma_{imp})$.

En la siguiente sección construiremos dicho objeto final. Además, trataremos de hacer una comparación del mismo con el objeto final general de la teoría oculta y con el modo en el que en EAT han sido implementadas las estructuras de datos de *segundo tipo*.

2.5.2 Objeto final

La existencia de objeto final en las categorías de álgebras ocultas está perfectamente determinada dentro de la teoría de la especificación oculta (ver, por ejemplo, [42]). Concretamente, hemos presentado en el Teorema 2.4.12 dicho objeto final a través de lo que sus autores denominan una "construcción mágica". No obstante, en nuestro caso, trabajamos con una signatura oculta Σ_{imp} con características particulares: presenta un único género oculto, no tiene operaciones visibles (distintas de las constantes) y cada operación es un deconstructor. Si trabajamos con este tipo de signaturas ocultas, obtenemos un objeto final para la categoría $Mod_{\mathcal{H}^D}(\Sigma_{imp})$. Obviamente, este objeto final es isomorfo al presentado por Goguen y Malcolm en [42], pero insistimos, para estas signaturas particulares, lejos de tener ese "aspecto mágico", el álgebra final admite una descripción bastante natural.

Además, podemos obtener este objeto final a partir del morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$ que hemos definido anteriormente. Más adelante daremos una descripción del objeto final como coproducto de las imágenes de los modelos de $\mathcal{E}^D$ por la transformación natural correspondiente en dicho morfismo.

Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, definimos un álgebra $\mathbb{1}^D$, que pertenece a $Obj(Mod_{\mathcal{H}^D}(\Sigma_{imp}))$ de la siguiente manera:

- el conjunto sobre el género oculto imp_Σ es $\mathbb{1}^D_{imp_\Sigma} := Obj(Mod_{\mathcal{E}^D}(\Sigma))$,
- para cada operación $imp\_\omega \colon imp_\Sigma s_1 \dots s_n \to s \in \Omega_{imp}$ la función $imp\_\omega_{\mathbb{1}^D}(A, d_1, \dots, d_n) := \omega_A(d_1, \dots, d_n)$, para cada $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$ y cada $d_i \in D_i,\ i = 1, \dots, n$.

Una aclaración necesaria es que el álgebra $\mathbb{1}^D$ está bien definida. Para ello es suficiente con comprobar que la categoría $Mod_{\mathcal{E}^D}(\Sigma)$ es una categoría pequeña, es decir, el soporte para el género oculto $Obj(Mod_{\mathcal{E}^D}(\Sigma))$ es efectivamente un conjunto. Esto es así ya que al fijar un dominio de datos, las Σ-álgebras que se pueden construir sobre ese dominio de datos forman un conjunto y no una clase estricta.

Teorema 2.5.5. *Sea Σ una signatura en $\mathcal{E}^D$. El objeto $\mathbb{1}^D$ es un álgebra final en $Mod_{\mathcal{H}^D}(\Sigma_{imp})$. Además, este objeto final es el coproducto en $Mod_{\mathcal{H}^D}(\Sigma_{imp})$ de los objetos en la imagen de $\beta_\Sigma^{\mathcal{E}^D - \mathcal{H}^D}(Mod_{\mathcal{E}^D}(\Sigma))$.*

Demostración. Sea $\Sigma = (S, \Omega \cup C)$ una signatura de $SIG_{\mathcal{E}^D}$. Demostraremos primero que $\mathbb{1}^D$ es objeto final en $Mod_{\mathcal{H}^D}(\Sigma_{imp})$.

Para cada álgebra $M \in Obj(Mod_{\mathcal{H}^D}(\Sigma_{imp}))$, tenemos que definir un Σ_{imp}-homomorfismo oculto $f^M \colon M \to \mathbb{1}^D$. Para el género oculto consideramos la aplicación $f^M_{imp_\Sigma} \colon M_{imp_\Sigma} \to \mathbb{1}^D_{imp_\Sigma}$, que a cada elemento $a \in M_{imp_\Sigma}$ le asocia un álgebra $f^M_{imp_\Sigma}(a) := M^a$, dada por: $\omega_{M^a}(d_1, \ldots, d_n) := imp\_\omega_M(a, d_1, \ldots, d_n)$, para cada operación $\omega \colon s_1 \ldots s_n \to s \in \Omega$ y cada tupla de datos $d_i \in D_i$, $i = 1, \ldots, n$.

Tenemos entonces que f^M es un Σ_{imp}-homomorfismo oculto, ya que para cada $imp\_\omega \colon imp_\Sigma s_1 \ldots s_n \to s \in \Omega_{imp}$, y para cada $a \in M_{imp_\Sigma}$, $d_i \in D_i$, $i = 1, \ldots, n$, se verifica que $imp\_\omega_{\mathbb{1}^D}(f^M_{imp_\Sigma}(a), d_1, \ldots, d_n) = imp\_\omega_{\mathbb{1}^D}(M^a, d_1, \ldots, d_n) = \omega_{M^a}(d_1, \ldots, d_n) = imp\_\omega_M(a, d_1, \ldots, d_n)$.

Además, f^M es único. Sea un Σ_{imp}-homomorfismo oculto $g \colon M \to \mathbb{1}^D$ definido por una aplicación $g_{imp_\Sigma} \colon M_{imp_\Sigma} \to \mathbb{1}^D_{imp_\Sigma}$, debemos comprobar que $f^M_{imp_\Sigma} = g_{imp_\Sigma}$. Dado $a \in M_{imp_\Sigma}$, se tiene $g_{imp_\Sigma}(a) = B^a \in Mod_{\mathcal{E}^D}(\Sigma)$. Como g es un Σ_{imp}-homomorfismo, $imp\_\omega_M(a, d_1, \ldots, d_n) = imp\_\omega_{\mathbb{1}^D}(g_{imp_\Sigma}(a), d_1, \ldots, d_n) = imp\_\omega_{\mathbb{1}^D}(B^a, d_1, \ldots, d_n) = \omega_{B^a}(d_1, \ldots, d_n)$, para cada $imp\_\omega \colon imp_\Sigma s_1 \ldots s_n \to s \in \Omega_{imp}$, y cada $d_i \in D_i$, $i = 1, \ldots, n$. Por otro lado, por definición de M^a, tenemos que $imp\_\omega_M(a, d_1, \ldots, d_n) = \omega_{M^a}(d_1, \ldots, d_n)$, para cada $imp\_\omega \colon imp_\Sigma s_1 \ldots s_n \to s \in \Omega_{imp}$, y cada $d_i \in D_i$, $i = 1, \ldots, n$. Obtenemos entonces que $\omega_{B^a}(d_1, \ldots, d_n) = \omega_{M^a}(d_1, \ldots, d_n)$, para cada $\omega \colon s_1 \ldots s_n \to s \in \Omega$, y cada $d_i \in D_i$, $i = 1, \ldots, n$, por lo que $B^a = M^a$. De esta forma, $f^M = g$.

Nos centramos a continuación en la demostración de la segunda parte del teorema, esto es, $\mathbb{1}^D$ es el coproducto en $Mod_{\mathcal{H}^D}(\Sigma_{imp})$ de los objetos en la imagen $\beta^{\mathcal{E}^D - \mathcal{H}^D}_\Sigma(Mod_{\mathcal{E}^D}(\Sigma))$.

Para cada $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$, el soporte para el género distinguido imp_Σ del objeto $\beta_\Sigma(A)$ es el conjunto unipuntual $\{*\}$. Consideramos el único Σ_{imp}-homomorfismo oculto h^A entre $\beta_\Sigma(A)$ y $\mathbb{1}^D$, que como hemos visto, viene dado por la aplicación $h^A_{imp_\Sigma} \colon \beta_\Sigma(A)_{imp_\Sigma} \to \mathbb{1}^D_{imp_\Sigma}$ tal que $h^A_{imp_\Sigma}(*) = A$. Entonces el par $(\mathbb{1}^D, \{h^A\}_{A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))})$ es el coproducto de la familia $\{\beta_\Sigma(A)\}_{A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))}$. En efecto, sea un álgebra $G \in Obj(Mod_{\mathcal{H}^D}(\Sigma_{imp}))$ tal que para cada $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$ exista un Σ_{imp}-homomorfismo oculto g^A entre $\beta_\Sigma(A)$ y G. Entonces, debemos comprobar que podemos definir un único Σ_{imp}-homomorfismo oculto k entre $\mathbb{1}^D$ y G que haga conmutativos todos los diagramas:

$$\begin{tikzcd}
\beta_\Sigma(A) \arrow[r, "h^A"] \arrow[dr, "g^A"'] & \mathbb{1}^D \arrow[d, "k"] \\
& G
\end{tikzcd}$$

El morfismo k estará determinado por la aplicación en el género distinguido $k_{imp_\Sigma} \colon \mathbb{1}^D_{imp_\Sigma} \to G_{imp_\Sigma}$. Consideramos $k_{imp_\Sigma}(A) := g^A_{imp_\Sigma}(*)$, para cada $A \in \mathbb{1}^D_{imp_\Sigma}$. Se trata de un morfismo de Σ_{imp}-álgebras ya que para cada operación $imp\_\omega \colon imp_\Sigma s_1 \ldots s_n \to s \in \Omega_{imp}$, cada $A \in \mathbb{1}^D_{imp_\Sigma}$, y cada tupla $d_i \in D_i$, $i = 1, \ldots, n$, tenemos por un lado que $imp\_\omega_{\mathbb{1}^D}(A, d_1, \ldots, d_n) = \omega_A(d_1, \ldots, d_n)$ por definición de

$1\!\!1^D$ y por otro lado que $imp\_\omega_G(k_{imp_\Sigma}(A), d_1, \ldots, d_n) = imp\_\omega_G(g^A_{imp_\Sigma}(*), d_1, \ldots, d_n) = imp\_\omega_{\beta_\Sigma(A)}(*, d_1, \ldots, d_n) = \omega_A(d_1, \ldots, d_n)$, ya que g^A es un Σ_{imp}-homomorfismo.

Además, los diagramas son conmutativos, ya que para cada $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$ se verifica que $k_{imp_\Sigma}(h^A_{imp_\Sigma}(*)) = k_{imp_\Sigma}(A) = g^A_{imp_\Sigma}(*)$, es decir, $k \circ h^A = g^A$.

Por último, este morfismo es único, ya que si existe otro Σ_{imp}-homomorfismo oculto k' entre $1\!\!1^D$ y G, tal que para cada $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$ verifica que $k' \circ h^A = g^A$, entonces $k_{imp_\Sigma}(A) = g^A_{imp_\Sigma}(*) = k'_{imp_\Sigma}(h^A_{imp_\Sigma}(*)) = k'_{imp_\Sigma}(A)$, por lo que $k = k'$. ■

Dada una de nuestras signaturas Σ_{imp}, podemos hacer en este momento una comparación entre el objeto final general $F^{\Sigma_{imp}}$ dentro de la teoría oculta, que hemos presentado en el Teorema 2.4.12, el objeto final $1\!\!1^D$ que hemos descrito en el teorema anterior, y el modo en el que han sido implementadas las estructuras de datos en EAT [81].

Debemos observar que las signaturas Σ_{imp} no tienen constructores y que por lo tanto, los Σ_{imp}-contextos para el género oculto son muy simples: son expresiones de la forma $imp\_\omega(z, d_1, \ldots, d_n)$, con z la única variable de género oculto y $d_i \in D_i$, $i = 1, \ldots, n$, para cada operación $imp\_\omega\colon imp_\Sigma s_1 \ldots s_n \to s$. Recordamos que se tiene como condición que todos los elementos de D_i son introducidos en la signatura como constantes (una de las razones por las que se introducen dichas constantes en la signatura es para poder obtener esta colección de contextos [42]). Entonces, ya que los conjuntos de contextos $C_{\Sigma_{imp}}[z]_s$ pueden verse como una unión disjunta (indexada por los deconstructores que finalizan en s), cada aplicación de $C_{\Sigma_{imp}}[z]_s$ en D_s puede verse como una tupla de funciones $\omega_s\colon D_{s_1} \times \cdots \times D_{s_n} \to D_s$, una función para cada operación $imp\_\omega\colon imp_\Sigma s_1 \ldots s_n \to s$ de Σ_{imp} que finaliza en s. Al recorrer s todos los géneros visibles obtenemos una serie de funciones que pueden ser reunidas en una única tupla funcional. Así, el objeto final $F^{\Sigma_{imp}}$ puede ser visto como un conjunto de tuplas de funciones y la interpretación de las operaciones imp_ω consiste en la aplicación de una función (que forma parte del primero de sus argumentos) sobre el resto de los argumentos, que son datos. Debemos notar que cada una de estas tuplas representa en realidad un objeto de $Mod_{\mathcal{E}^D}(\Sigma)$ al estar D fijado.

Por ejemplo, en el caso de la signatura $\mathtt{GRP}_{imp}$, existe un único género visible g, y por lo tanto un único conjunto $C_{\mathtt{GRP}_{imp}}[z]_g$. Este conjunto está formado por contextos de la forma $imp\_prd(z, a, b)$ que contribuyen a la definición de una función $f_1\colon D_g \times D_g \to D_g$, contextos de la forma $imp\_inv(z, a)$ que contribuyen a la definición de una función $f_2\colon D_g \to D_g$ y el contexto $imp\_unt(z)$ que contribuye a la definición de una función $f_3\colon \ \to D_g$. Las tuplas formadas por tres funciones con estas aridades constituyen el soporte del género oculto del álgebra $F^{\mathtt{GRP}_{imp}}$ y definen de forma unívoca un álgebra de $Mod_{\mathcal{E}^D}(\mathtt{GRP})$.

Se hace explícito de esta forma el Σ_{imp}-isomorfismo oculto entre los objetos finales $1\!\!1^D$ y $F^{\Sigma_{imp}}$ a través de esta biyección entre sus correspondientes soportes para el género destacado. Observamos entonces que, para nuestro caso particular, el objeto final admite ser descrito como un objeto cuyo soporte en el único género oculto es un conjunto de tuplas funcionales, cada tupla codificando las operaciones de un modelo de $\mathcal{E}^D$. Además, este conjunto soporte es suficiente para actuar como índice de modelos de $\mathcal{E}^D$. Simplemente recuperamos para cada operación de un álgebra de $\mathcal{E}^D$ la correspondiente función en la tupla. Los detalles de dicho objeto están desarrollados en [58]. Por último señalar que este objeto se ajusta completamente con el modo elegido por Rubio y Sergeraert para implementar las estructuras de datos de EAT, utilizando para ello de forma intensiva la programación funcional (ver [81, 58]). Las representaciones en EAT de las estructuras matemáticas consisten básicamente en tuplas de funciones que codifican las operaciones de esas estructuras. De este modo, el teorema de existencia de objeto final prueba de manera formal que la representación que usa EAT para sus estructuras datos es lo "más general" posible desde el punto de vista de la Teoría de Categorías (puede pensarse que recoge como elementos a las álgebras individuales y desde cualquier otra "familia" de álgebras individuales siempre hay un "paso canónico" al objeto final).

Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, recogemos de forma explícita esta representación equivalente del objeto final en $Mod_{\mathcal{H}^D}(\Sigma_{imp})$ mediante la Σ_{imp}-álgebra *canónica* A^{can} dada por:

- $A^{can}_s = D_s$, para cada género $s \in S$,
- $A^{can}_{imp_\Sigma} = \{(f_\omega\colon D_{s_1} \times \cdots \times D_{s_n} \to D_s)_{\omega\colon s_1\ldots s_n \to s \in \Omega} \mid \langle (D_s)_{s\in S}, ((f_\omega)_{\omega\in\Omega}, C)\rangle \in Mod_{\mathcal{E}^D}(\Sigma)\}$,
- $d_{A^{can}} = d$, para cada constante $d \in C$, y
- $imp\_\omega_{A^{can}}((f_\delta)_{\delta\in\Omega}, d_1, \ldots, d_n) := f_\omega(d_1, \ldots, d_n)$, para cada operación $imp\_\omega\colon imp_\Sigma s_1 \ldots s_n \to s \in \Omega_{imp}$ y para cada $(f_\delta)_{\delta\in\Omega} \in A^{can}_{imp_\Sigma}$ y $d_i \in D_i$, $i = 1, \ldots, n$.

Teorema 2.5.6. *La Σ_{imp}-álgebra A^{can} es un objeto final en la categoría $Mod_{\mathcal{H}^D}(\Sigma_{imp})$.*

2.5.3 Ampliación del morfismo a especificaciones

Podemos ampliar el morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$ a especificaciones de forma análoga a como se amplia el morfismo entre $\mathcal{E}^D$ y $\mathcal{E}$, de modo que obtenemos una propiedad equivalente a la Proposición 2.3.8, ahora sobre álgebra ocultas.

Además, en la institución oculta, dada una especificación $Espec_{imp} = (\Sigma_{imp}, E_{imp})$ podemos definir un modelo canónico A'^{can} que es final dentro del TAD definido por dicha especificación. Este modelo canónico se define de la misma forma que el objeto final A^{can} construido en la sección anterior, pero con la restricción de que sólo tomamos tuplas funcionales en el género distinguido que definan, junto con D, un modelo de $Espec = (\Sigma, E)$ (en lugar de simplemente definir un modelo para Σ). Recogemos la correspondiente propiedad de finalidad en el siguiente teorema.

Teorema 2.5.7. *La Σ_{imp}-álgebra A'^{can} es un objeto final en el TAD definido por $Espec_{imp} = (\Sigma_{imp}, E_{imp})$.*

Ejemplo 2.5.8. Continuamos con la especificación de grupo $Espec_{\mathtt{GRP}}$ que hemos introducido en la sección anterior. Fijado un dominio de datos D_g, definimos la $\mathtt{GRP}_{imp}$-álgebra A'^{can} modelo para $Espec_{\mathtt{GRP}_{imp}}$ de la siguiente forma: $A'^{can}_g = D_g$ y $A'^{can}_{imp_{\mathtt{GRP}}} = \{(f_1, f_2, f_3) \mid f_1\colon D_g \times D_g \to D_g,\ f_2\colon D_g \to D_g,\ f_3\colon\ \to D_g$, tal que definan un grupo sobre $D_g\}$.

La interpretación en A'^{can} de las constantes visibles es la habitual y de las tres operaciones no visibles viene definida por:

$$imp\_prd_{A'^{can}}((f_1, f_2, f_3), z1, z2) = f_1(z1, z2)$$
$$imp\_inv_{A'^{can}}((f_1, f_2, f_3), z) = f_2(z)$$
$$imp\_unt_{A'^{can}}((f_1, f_2, f_3)) = f_3$$

Esta álgebra es la más “general” entre las representaciones de grupos sobre D_g, de modo que cubre todos los posibles grupos que pueden ser definidos sobre dicho conjunto. En particular, si consideramos $D_g = \langle n \rangle$ entonces este objeto es isomorfo a la $\mathtt{GRP}_{imp}$-álgebra B del Ejemplo 2.3.9, lo que prueba que esta álgebra es también objeto final en el TAD definido por $Espec_{\mathtt{GRP}_{imp}}$. □

2.6 Un tercer marco de especificación: coálgebras

En las secciones anteriores hemos estudiado una operación entre signaturas, la operación *imp*, dentro de dos marcos diferentes: el de la especificación algebraica ecuacional y el de la especificación oculta. Ahora, nos disponemos a dar una interpretación de la misma operación dentro de un tercer formalismo, como son las coálgebras. Uno de los primeros trabajos en los que aparece el concepto de *coálgebra* en el campo de la especificación algebraica es [74] en 1995. La teoría coalgebraica se ha desarrollado notablemente desde entonces dando lugar a numerosas publicaciones. Artículos que sirven de guías en el uso de las coálgebras son [83] y [54]. Uno de los objetivos de esta teoría consiste en la modelización de los lenguajes orientados a objetos, de modo que ya en el trabajo de Reichel [74] se propone de manera explícita el uso de las coálgebras como semántica para el paradigma orientado a objetos. Artículos que persiguen este objetivo son entre otros [53, 48, 50, 52]. De modo informal, el concepto de coálgebra consiste en un conjunto *oculto* junto con una serie de *observadores* para dicho conjunto. Así, las coálgebras están muy relacionadas con las especificaciones ocultas, como se pone de manifiesto en [22] o en [23], de modo que si se prescinden en las signaturas ocultas de las operaciones constructoras a partir de datos se obtiene que las álgebras para dichas signaturas son en realidad coálgebras particulares.

En el siguiente apartado introducimos las definiciones básicas de esta teoría que pueden encontrarse en cualquiera de los anteriores artículos. No obstante, para poder desarrollar nuestros conceptos dentro de un marco institucional, necesitamos introducir el concepto de *institución coalgebraica asociada a una institución*. La mayor parte de los resultados contenidos en esta sección están publicados en [30].

2.6.1 Coálgebras

Desde un punto de vista categórico, la noción de coálgebra puede verse como el concepto dual de la noción de álgebra. La diferencia fundamental entre un álgebra y una coálgebra es que la primera tienen *constructores*, es decir, operaciones que llegan a un género distinguido y permiten construir elementos del mismo, mientras que la segunda posee *observadores*, es decir, operadores que parten de un género distinguido y permiten observar determinado comportamiento.

Definición 2.6.1 (Coálgebra). Sea $F\colon Set \to Set$ un endofuntor de Set. Una F*-coálgebra* es un par (X, α), donde X es un conjunto y $\alpha\colon X \to F(X)$ es una aplicación.

Si no hay lugar a confusión, hablaremos simplemente de coálgebra, sin hacer referencia al funtor F y podemos denotar una coálgebra por la aplicación $\alpha\colon X \to F(X)$.

Podemos establecer relaciones entre coálgebras a través de la noción de morfismo de coálgebras.

Definición 2.6.2 (Morfismo de coálgebras). Sea $F\colon Set \to Set$ un endofuntor de Set. Un F*-homomorfismo* entre dos F-coálgebras (X, α), (Y, β) es una aplicación $f\colon X \to Y$ tal que $F(f) \circ \alpha = \beta \circ f$.

Para cada endofuntor F de Set, las F-coálgebras junto con los F-homomorfismos entre ellas definen una categoría que denotaremos por $CoAlg(F)$.

2.6.2 Definición de una institución coalgebraica

En esta sección definiremos una institución para coálgebras asociada a una institución dada. Debemos adelantar que nuestra institución no recoge todas las posibles coálgebras definidas sobre cualquier tipo de endofuntor de Set sino que nos restringimos a las coálgebras que se pueden construir sobre un tipo muy particular de endofuntores como son los constantes. Estos endofuntores serán suficientes para nuestros propósitos de modelado.

El punto de partida es una institución $\mathcal{I}$ con la propiedad de que, para cada signatura Σ de $SIG_{\mathcal{I}}$, la categoría $Mod_{\mathcal{I}}(\Sigma)$ es una categoría pequeña, lo que en particular implica que la clase $Obj(Mod_{\mathcal{I}}(\Sigma))$ es un conjunto. Esta condición puede parecer demasiado restrictiva, pero de hecho, no lo es tanto. Por ejemplo, como se ha comentado previamente, cuando tratamos de especificar un sistema de software mediante álgebras sobre una signatura, si exigimos que los conjuntos soporte de cualquier álgebra estén contenidos dentro de una familia de conjuntos de datos fijada (por ejemplo, objetos definibles en un lenguaje de programación particular), entonces obtenemos una institución con la propiedad mencionada.

A partir de esta institución $\mathcal{I}$ que denotamos por $(SIG_{\mathcal{I}}, Sen_{\mathcal{I}}, Mod_{\mathcal{I}}, \models^{\mathcal{I}})$ definimos a continuación una institución, que llamamos *institución coalgebraica asociada a* $\mathcal{I}$, y que denotamos por $CoAlg(\mathcal{I}) = (SIG_{CoAlg(\mathcal{I})}, Sen_{CoAlg(\mathcal{I})}, Mod_{CoAlg(\mathcal{I})}, \models^{CoAlg(\mathcal{I})})$.

La categoría de signaturas $SIG_{CoAlg(\mathcal{I})}$ y el funtor sentencias $Sen_{CoAlg(\mathcal{I})}$ se definen de la misma forma que las respectivas componentes de la institución $\mathcal{I}$.

El funtor modelos $Mod_{CoAlg(\mathcal{I})}\colon SIG_{CoAlg(\mathcal{I})} \to Cat^{op}$ viene dado de la siguiente forma. Para cada signatura Σ de $SIG_{CoAlg(\mathcal{I})}$, definimos $Mod_{CoAlg(\mathcal{I})}(\Sigma) := CoAlg(F_\Sigma)$, donde $F_\Sigma\colon Set \to Set$ es el endofuntor constante sobre el conjunto $Obj(Mod_{\mathcal{I}}(\Sigma))$. Es decir, la categoría de modelos que corresponde a la signatura Σ es la categoría de coálgebras definidas sobre F_Σ.

Para cada morfismo de signaturas $\mu\colon \Sigma \to \Sigma'$ de $SIG_{CoAlg(\mathcal{I})}$, $Mod_{CoAlg(\mathcal{I})}(\mu)$ es el funtor en Cat^{op} entre $CoAlg(F_{\Sigma'})$ y $CoAlg(F_\Sigma)$ que definimos de la siguiente forma: para cada $F_{\Sigma'}$-coálgebra (X', α'), definimos la F_Σ-coálgebra $Mod_{CoAlg(\mathcal{I})}(\mu)((X', \alpha')) := (X', Mod_{\mathcal{I}}(\mu) \circ \alpha')$, donde $Mod_{\mathcal{I}}(\mu)$ representa la aplicación sobre objetos entre los conjuntos $Obj(Mod_{\mathcal{I}}(\Sigma'))$ y $Obj(Mod_{\mathcal{I}}(\Sigma))$. Además, a un $F_{\Sigma'}$-homomorfismo $f\colon X' \to Y'$, entre dos $F_{\Sigma'}$-coálgebras (X', α'), (Y', β'), $Mod_{CoAlg(\mathcal{I})}(\mu)$ le hace corresponder el F_Σ-homomorfismo, entre las F_Σ-coálgebras $(X', Mod_{\mathcal{I}}(\mu) \circ \alpha')$, $(Y', Mod_{\mathcal{I}}(\mu) \circ \beta')$, que está definido por la misma aplicación $f\colon X' \to Y'$. Esta aplicación verifica la condición de homomorfismo $(Mod_{\mathcal{I}}(\mu) \circ \beta') \circ f = F_\Sigma(f) \circ (Mod_{\mathcal{I}}(\mu) \circ \alpha')$, ya que F_Σ un funtor constante y por lo tanto $F_\Sigma(f)$ es una aplicación identidad, y como f es un $F_{\Sigma'}$-homomorfismo entre (X', α') e (Y', β') y $F_{\Sigma'}$ es de nuevo un funtor constante, tenemos que $\alpha' = \beta' \circ f$.

Es inmediato comprobar que $Mod_{CoAlg(\mathcal{I})}(\mu)$ conserva las identidades y la composición de morfismos por lo que realmente define un funtor.

En el siguiente lema expresamos que el funtor modelos está bien definido.

Lema 2.6.3. *La aplicación* $Mod_{CoAlg(\mathcal{I})}\colon SIG_{CoAlg(\mathcal{I})} \to Cat^{op}$ *es un funtor.*

Demostración. Es sencillo demostrar que esta aplicación conserva las identidades y la composición de morfismos a partir de que $Mod_{\mathcal{I}}\colon SIG_{\mathcal{I}} \to Cat^{op}$ es un funtor. ■

Para completar una institución falta por determinar la relación de satisfacción. Para cada signatura Σ de $SIG_{CoAlg(\mathcal{I})}$ definimos una relación $\models_\Sigma^{CoAlg(\mathcal{I})}$ entre $Obj(Mod_{CoAlg(\mathcal{I})}(\Sigma))$ y $Sen_{CoAlg(\mathcal{I})}(\Sigma)$ de la siguiente forma: dada una F_Σ-coálgebra (X, α) de $Mod_{CoAlg(\mathcal{I})}(\Sigma)$ y una ecuación e de $Sen_{CoAlg(\mathcal{I})}(\Sigma)$

$$(X, \alpha) \models_\Sigma^{CoAlg(\mathcal{I})} e \quad \text{si y sólo si} \quad \alpha(x) \models_\Sigma^{\mathcal{I}} e,\ \forall x \in X.$$

Para comprobar que las componentes que hemos definido constituyen una institución es necesario demostrar que cumplen la condición de satisfacibilidad que recogemos en el siguiente teorema.

Teorema 2.6.4. *Sean* $\mu\colon \Sigma \to \Sigma'$ *un morfismo de* $SIG_{CoAlg(\mathcal{I})}$, (X', α') *una* $F_{\Sigma'}$-*coálgebra de* $Mod_{CoAlg(\mathcal{I})}(\Sigma')$ *y* e *una ecuación de* $Sen_{CoAlg(\mathcal{I})}(\Sigma)$. *Entonces,*

$$(X', \alpha') \models_{\Sigma'}^{CoAlg(\mathcal{I})} Sen_{CoAlg(\mathcal{I})}(\mu)(e) \quad \textit{si y sólo si} \quad Mod_{CoAlg(\mathcal{I})}(\mu)((X', \alpha')) \models_\Sigma^{CoAlg(\mathcal{I})} e.$$

Demostración. Si aplicamos la definición de $Sen_{CoAlg(\mathcal{I})}$, de $Mod_{CoAlg(\mathcal{I})}$ y de la relación $\models^{CoAlg(\mathcal{I})}$, obtenemos que es suficiente con demostrar que

$$\alpha'(x') \models^{\mathcal{I}}_{\Sigma'} Sen_{\mathcal{I}}(\mu)(e),\ \forall x' \in X' \quad \text{si y sólo si} \quad Mod_{\mathcal{I}}(\mu) \circ \alpha'(x') \models^{\mathcal{I}}_{\Sigma} e,\ \forall x' \in X'.$$

Esta equivalencia se obtiene directamente a partir de la relación de satisfacibilidad de la institución $\mathcal{I}$. ■

2.6.3 Un morfismo de instituciones unidireccional entre una institución $\mathcal{I}$ y su institución coalgebraica $CoAlg(\mathcal{I})$

La relación que existe entre una institución y su institución coalgebraica asociada $CoAlg(\mathcal{I})$ puede concretarse en un morfismo de instituciones unidireccional entre ambas. Sin embargo, para poder definir este morfismo de instituciones de forma correcta necesitamos exigir una condición más sobre $\mathcal{I}$: cada morfismo en $Mod_{\mathcal{I}}(\Sigma)$ debe ser un endomorfismo. En el siguiente apartado veremos como esta condición está presente en el contexto particular en el que trabajamos (y también que surge de manera natural en las especificaciones ocultas).

Consideramos una institución $\mathcal{I} = (SIG_{\mathcal{I}}, Sen_{\mathcal{I}}, Mod_{\mathcal{I}}, \models^{\mathcal{I}})$ tal que para cada signatura $\Sigma \in Obj(SIG_{\mathcal{I}})$, la categoría $Mod_{\mathcal{I}}(\Sigma)$ es una categoría pequeña y además que todos sus morfismos son siempre endomorfismos, y tomamos la institución coalgebraica $CoAlg(\mathcal{I}) = (SIG_{CoAlg(\mathcal{I})}, Sen_{CoAlg(\mathcal{I})}, Mod_{CoAlg(\mathcal{I})}, \models^{CoAlg(\mathcal{I})})$ asociada a $\mathcal{I}$. En esta sección definiremos un morfismo de instituciones unidireccional de $\mathcal{I}$ en $CoAlg(\mathcal{I})$.

Las componentes del morfismo correspondientes a la relación entre las categorías de signaturas y los funtores sentencias son identidades ya que ambas instituciones comparten estas componentes.

La tercera componente del morfismo de instituciones unidireccional relaciona los modelos de ambas instituciones. Para ello necesitamos definir una transformación natural $\beta\colon Mod_{\mathcal{I}} \Rightarrow Mod_{CoAlg(\mathcal{I})}$ entre los funtores modelos $Mod_{\mathcal{I}}\colon SIG_{\mathcal{I}} \to Cat^{op}$ y $Mod_{CoAlg(\mathcal{I})}\colon SIG_{\mathcal{I}} \to Cat^{op}$, ya que el funtor entre signaturas del morfismo que estamos definiendo es la identidad. Esta transformación natural consiste en un funtor $\beta_{\Sigma}\colon Mod_{\mathcal{I}}(\Sigma) \to Mod_{CoAlg(\mathcal{I})}(\Sigma)$ por cada objeto $\Sigma \in Obj(SIG_{\mathcal{I}})$, tal que verifiquen la propiedad de naturalidad.

Dada una signatura Σ de $SIG_{\mathcal{I}}$ y dado un modelo $A \in Obj(Mod_{\mathcal{I}}(\Sigma))$, definimos $\beta_{\Sigma}(A) := (\{*\}, \alpha_A\colon \{*\} \to F_{\Sigma}(\{*\}))$, donde la aplicación α_A viene dada por $\alpha_A(*) = A$. Es decir, $\beta_{\Sigma}(A)$ es la F_{Σ}-coálgebra que está formada por un conjunto unipuntual y la aplicación que asocia su único elemento al objeto A.

El único homomorfismo existente entre las F_{Σ}-coálgebras $\beta_{\Sigma}(A) = (\{*\}, \alpha_A)$ y $\beta_{\Sigma}(B) = (\{*\}, \alpha_B)$ es la identidad del conjunto $\{*\}$. Así, para que β_{Σ} se extienda a un funtor es necesario que en $Mod_{\mathcal{I}}(\Sigma)$ sólo existan endomorfismos, condición que nosotros hemos exigido. Ahora, es un simple ejercicio comprobar que β_{Σ} respeta las identidades y la composición de morfismos de $Mod_{\mathcal{I}}(\Sigma)$, por lo que realmente obtenemos un funtor $\beta_\Sigma\colon Mod_{\mathcal{I}}(\Sigma) \to Mod_{CoAlg(\mathcal{I})}(\Sigma)$ para cada signatura Σ de $SIG_{\mathcal{I}}$.

Falta por demostrar que esta familia de funtores forman una transformación natural, propiedad que recogemos en el siguiente lema.

Lema 2.6.5. *La familia de funtores $\beta\colon Mod_{\mathcal{I}} \Rightarrow Mod_{CoAlg(\mathcal{I})}$ define una transformación natural.*

Demostración. Debemos demostrar que esta familia de funtores verifican la propiedad de naturalidad, es decir, si para cada morfismo $\mu\colon \Sigma \to \Sigma'$ de $SIG_{\mathcal{I}}$, el siguiente diagrama conmuta sobre objetos y morfismos:

$$\begin{tikzcd}
Mod_{\mathcal{I}}(\Sigma) \arrow[r, "\beta_\Sigma"] & Mod_{CoAlg(\mathcal{I})}(\Sigma) \\
Mod_{\mathcal{I}}(\Sigma') \arrow[u, "Mod_{\mathcal{I}}(\mu)"] \arrow[r, "\beta_{\Sigma'}"'] & Mod_{CoAlg(\mathcal{I})}(\Sigma') \arrow[u, "Mod_{CoAlg(\mathcal{I})}(\mu)"']
\end{tikzcd}$$

Dado $A' \in Obj(Mod_{\mathcal{I}}(\Sigma'))$, si aplicamos sucesivamente la definición $\beta_{\Sigma'}$ y de $Mod_{CoAlg(\mathcal{I})}(\mu)$, tenemos que $Mod_{CoAlg(\mathcal{I})}(\mu)(\beta_{\Sigma'}(A')) = Mod_{CoAlg(\mathcal{I})}(\mu)((\{*\}, \alpha_{A'})) = (\{*\}, Mod_{\mathcal{I}}(\mu)\circ\alpha_{A'})$, que es exactamente la F_Σ-coálgebra definida por $\beta_\Sigma(Mod_{\mathcal{I}}(\mu)(A'))$.

Dado un morfismo $h'\colon A' \to A'$ de $Mod_{\mathcal{I}}(\Sigma')$, si aplicamos de nuevo sucesivamente la definición $\beta_{\Sigma'}$ y de $Mod_{CoAlg(\mathcal{I})}(\mu)$, tenemos que $Mod_{CoAlg(\mathcal{I})}(\mu)(\beta_{\Sigma'}(h')) = Mod_{CoAlg(\mathcal{I})}(\mu)(id_{\{*\}}) = id_{\{*\}}$, que es igual al F_Σ-homomorfismo de coálgebras definido por $\beta_\Sigma(Mod_{\mathcal{I}}(\mu)(h'))$. ■

Para comprobar que esta transformación natural constituye junto con el funtor identidad sobre signaturas y la transformación natural identidad sobre sentencias un morfismo de instituciones unidireccional, únicamente falta por demostrar que éstas verifican la condición de satisfacibilidad. Es decir, debemos demostrar que para cada objeto Σ de $SIG_{\mathcal{I}}$ se verifica que

$$A \models^{\mathcal{I}}_{\Sigma} e \quad \text{si y sólo si} \quad \beta_\Sigma(A) \models^{CoAlg(\mathcal{I})}_{\Sigma} e$$

para cada Σ-modelo A y cada Σ-sentencia e de $\mathcal{I}$.

Pero tenemos que $\beta_\Sigma(A) = (\{*\}, \alpha_A)$, con $\alpha_A(*) = A$, por lo que obtenemos esta condición de satisfacibilidad de forma inmediata.

Reflejamos este morfismo de instituciones en el siguiente teorema.

Teorema 2.6.6. *Sea $\mathcal{I}$ una institución tal que para cada $\Sigma \in Obj(SIG_{\mathcal{I}})$, $Mod_{\mathcal{I}}(\Sigma)$ es una categoría pequeña y cada morfismo en $Mod_{\mathcal{I}}(\Sigma)$ es un endomorfismo. Entonces, existe un morfismo de instituciones unidireccional canónico entre $\mathcal{I}$ y $CoAlg(\mathcal{I})$.*

2.6.4 Un morfismo de instituciones unidireccional entre la institución $\mathcal{E}^D$ y su institución coalgebraica $CoAlg(\mathcal{E}^D)$

Consideramos, como caso particular de institución $\mathcal{I}$, la institución algebraica ecuacional definida sobre un universo de datos D, que hemos denotado por $\mathcal{E}^D$. Esta institución verifica las condiciones exigidas en el morfismo de instituciones unidireccional del apartado anterior. Por un lado, las categorías de modelos $Mod_{\mathcal{E}^D}(\Sigma)$ son pequeñas, ya que estos modelos están construidos sobre un universo de datos D fijado. Por otro lado, en la definición de dicha categoría exigimos que los morfismos sean identidades entre modelos, por lo que en particular son endomorfismos. Entonces, utilizando la técnica anterior, podemos construir la institución coalgebraica asociada a $\mathcal{E}^D$. Además, obtenemos un morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y su institución coalgebraica asociada $CoAlg(\mathcal{E}^D)$.

Este morfismo de instituciones nos permite también modelar las estructuras de datos presentes en EAT. Esto no nos debe resultar extraño, ya que, como podemos deducir, por ejemplo, de [22], la categoría de álgebras ocultas $HAlg^D(\Sigma_{imp})$ es canónicamente equivalente a la categoría de coálgebras que definimos por $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$. De esta forma ambos formalismos son equivalentes en expresividad.

En este caso, otra vez nos movemos desde la categoría de modelos en $\mathcal{E}^D$ a una categoría de familias indexadas ya que cada coálgebra de $CoAlg(\mathcal{E}^D)$ representa una familia de modelos en $\mathcal{E}^D$. No obstante, nuevamente, esta representación no es del todo recogida por el morfismo de instituciones anterior, ya que aunque desde un punto de vista sintáctico, sí que cada funtor F_Σ puede ser interpretado como una signatura para implementaciones de Σ-álgebras, los modelos en la imagen del morfismo representan familias con una única álgebra tal y como ocurre con el morfismo de instituciones que termina en la institución oculta $\mathcal{H}^D$. Sin embargo, podemos agrupar, como en el caso oculto, estas imágenes como coproductos para obtener objetos finales que también corresponden directamente con las estructuras de datos que aparecen en EAT.

2.6.5 Objeto final

La existencia de objeto final en la categoría $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$ puede ser obtenida por diferentes medios: como caso particular de categoría indexada o fibrada [97], a partir de resultados generales sobre coálgebras [5], a partir de resultados conocidos en el área de la especificación coalgebraica [74], a partir de la teoría oculta [42], o en nuestro caso muy particular, simplemente por un argumento elemental en teoría de categorías, porque dicha categoría no es sino la categoría cociente (*slice*) obtenida al fijar un conjunto (en la categoría de conjuntos) [7]. Por tanto, podemos describir este objeto final a través de la aplicación identidad $1\!\!1^D \colon Obj(Mod_{\mathcal{E}^D}(\Sigma)) \to Obj(Mod_{\mathcal{E}^D}(\Sigma))$. Utilizamos para este objeto la misma notación que para el objeto final obtenido en las álgebras ocultas. De hecho el lector que conozca la relación entre álgebras ocultas y coálgebras [22] observará que ésta es la coálgebra que corresponde al álgebra oculta final [42].

El siguiente resultado muestra que, igual que ocurría en el caso de las álgebras ocultas, el morfismo de instituciones permite la reconstrucción del objeto final.

Teorema 2.6.7. *El objeto final* $\mathbb{1}^D$ *es el coproducto de la imagen sobre los modelos del morfismo de instituciones canónico entre* $\mathcal{E}^D$ *y* $CoAlg(\mathcal{E}^D)$*.*

Demostración. Para cada $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$, se tiene la F_Σ-coálgebra $\beta_\Sigma^{\mathcal{E}^D - CoAlg(\mathcal{E}^D)}(A) = (\{*\},\ \alpha_A \colon \{*\} \to Obj(Mod_{\mathcal{E}^D}(\Sigma)))$, donde α_A viene definida por $\alpha_A(*) = A$. Podemos considerar entonces un morfismo i_A entre esta coálgebra y $\mathbb{1}^D$, concretamente $i_A \colon \{*\} \to Obj(Mod_{\mathcal{E}^D}(\Sigma))$, que definimos también por $i_A(*) = A$.

Sea $(X, \alpha \colon X \to Obj(Mod_{\mathcal{E}^D}(\Sigma)))$ otra F_Σ-coálgebra para la que exista un morfismo $h_A \colon \{*\} \to X$ entre las coálgebras $(\{*\},\ \alpha_A)$ y (X, α), para cada $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$. Podemos definir entonces un morfismo $f \colon Obj(Mod_{\mathcal{E}^D}(\Sigma)) \to X$ como $f(A) = h_A(*)$, $\forall A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$. Es claro que éste es el (único) morfismo que verifica $f(i_A(*)) = h_A(*)$, $\forall A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$. ■

El objeto $\mathbb{1}^D$ admite otra representación. Si repetimos el argumento que hemos dado para el objeto final oculto, cada modelo de $Mod_{\mathcal{E}^D}(\Sigma)$, con $\Sigma = (S, \Omega)$ y $\Omega = (\omega_1, \ldots, \omega_k)$, puede ser identificado con una tupla funcional $(f_1, \ldots, f_k)$, que corresponde con las funciones que interpretan las operaciones en ese modelo, ya que el dominio de datos D permanece fijado. Estas tuplas de funciones determinan este objeto final. Recordamos que dicha representación corresponde directamente con el modo en que las estructuras de datos fueron implementadas en EAT.

2.6.6 Un morfismo de instituciones unidireccional entre la institución $CoAlg(\mathcal{E}^D)$ y la institución $\mathcal{H}^D$

Definimos a continuación un morfismo de instituciones unidireccional entre la institución coalgebraica asociada a la institución $\mathcal{E}^D$, $CoAlg(\mathcal{E}^D) = (SIG_{CoAlg(\mathcal{E}^D)}, Sen_{CoAlg(\mathcal{E}^D)}, Mod_{CoAlg(\mathcal{E}^D)}, \models^{CoAlg(\mathcal{E}^D)})$ y la institución oculta $\mathcal{H}^D = (SIG_{\mathcal{H}^D}, Sen_{\mathcal{H}^D}, Mod_{\mathcal{H}^D}, \models^{\mathcal{H}^D})$. Este morfismo está inspirado en la relación conocida entre las coálgebras y las álgebras ocultas [22] y nos permite establecer una conexión entre estos dos marcos de especificación.

Las componentes que relacionan las signaturas $\Phi^{CoAlg(\mathcal{E}^D)-\mathcal{H}^D}$ y las sentencias $\alpha^{CoAlg(\mathcal{E}^D)-\mathcal{H}^D}$ de este morfismo de instituciones unidireccional se definen de la misma forma que las respectivas componentes del morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$.

La tercera y última componente del morfismo de instituciones unidireccional será una transformación natural $\beta \colon Mod_{CoAlg(\mathcal{E}^D)} \Rightarrow Mod_{\mathcal{H}^D} \circ \Phi$. Esta transformación natural consistirá en un funtor $\beta_\Sigma \colon Mod_{CoAlg(\mathcal{E}^D)}(\Sigma) \to Mod_{\mathcal{H}^D}(\Phi(\Sigma))$, por cada signatura Σ de $SIG_{CoAlg(\mathcal{E}^D)}$, verificando la propiedad de naturalidad.

Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{CoAlg(\mathcal{E}^D)}$, para definir un funtor

$\beta_\Sigma\colon Mod_{CoAlg(\mathcal{E}^D)}(\Sigma) \to Mod_{\mathcal{H}^D}(\Phi(\Sigma))$ debemos establecer la imagen de los objetos y morfismos de $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$.

Dada una F_Σ-coálgebra $(\mathcal{X},\gamma)$ de $Obj(Mod_{CoAlg(\mathcal{E}^D)}(\Sigma))$, entonces $\beta_\Sigma((\mathcal{X},\gamma))$ es la Σ_{imp}-álgebra oculta que tiene:

- en el único género oculto imp_Σ, el soporte $\beta_\Sigma((\mathcal{X},\gamma))_{imp_\Sigma} = \mathcal{X}$,
- y para cada operación oculta $imp\_\omega\colon imp_\Sigma s_1 \ldots s_n \to s \in \Omega_{imp}$, $n \geq 0$, la función $imp\_\omega_{\beta_\Sigma((\mathcal{X},\gamma))}(a, d_1, \ldots, d_n) := \omega_{\gamma(a)}(d_1, \ldots, d_n)$, para cada $a \in \mathcal{X}$ y cada $d_i \in D_i$, $\forall i = 1, \ldots, n$.

Dado un homomorfismo $f\colon \mathcal{X} \to \mathcal{X}'$ entre dos F_Σ-coálgebras $(\mathcal{X},\gamma)$ y $(\mathcal{X}',\gamma')$, definimos $\beta_\Sigma(f)$ como el homomorfismo oculto, entre las Σ_{imp}-álgebras $\beta_\Sigma((\mathcal{X},\gamma))$ y $\beta_\Sigma((\mathcal{X}',\gamma'))$, que tiene a f por aplicación en el género oculto.

En el siguiente lema demostramos que el homomorfismo oculto anterior está bien definido.

Lema 2.6.8. *Sea $f\colon \mathcal{X} \to \mathcal{X}'$ un homomorfismo entre dos F_Σ-coálgebras $(\mathcal{X},\gamma)$ y $(\mathcal{X}',\gamma')$, con $\Sigma = (S, \Omega \cup C)$. La aplicación f junto con las aplicaciones identidades id_{D_s}, para cada $s \in S$, definen un Σ_{imp}-homomorfismo oculto entre las Σ_{imp}-álgebras ocultas $\beta_\Sigma((\mathcal{X},\gamma))$ y $\beta_\Sigma((\mathcal{X}',\gamma'))$.*

Demostración. Debemos comprobar que esta familia de aplicaciones verifica la condición de homomorfismo, es decir, que para cada operación $imp\_\omega\colon imp_\Sigma s_1 \ldots, s_n \to s \in \Omega_{imp}$, $n \geq 0$, se verifica que

$$imp\_\omega_{\beta_\Sigma((\mathcal{X},\gamma))}(a, d_1, \ldots, d_n) = imp\_\omega_{\beta_\Sigma((\mathcal{X}',\gamma'))}(f(a), d_1, \ldots, d_n),$$

para todo $a \in \mathcal{X}$ y $d_i \in D_i$, $\forall i = 1, \ldots, n$.

Sea $imp\_\omega\colon imp_\Sigma s_1 \ldots, s_n \to s \in \Omega_{imp}$, $n \geq 0$, y sean $a \in \mathcal{X}$, $d_i \in D_i$, $\forall i = 1, \ldots, n$. Por definición de $\beta_\Sigma((\mathcal{X},\gamma))$ y de $\beta_\Sigma((\mathcal{X}',\gamma'))$ tenemos respectivamente que $imp\_\omega_{\beta_\Sigma((\mathcal{X},\gamma))}(a, d_1, \ldots, d_n) = \omega_{\gamma(a)}(d_1, \ldots, d_n)$ e $imp\_\omega_{\beta_\Sigma((\mathcal{X}',\gamma'))}(f(a), d_1, \ldots, d_n) = \omega_{\gamma' \circ f(a)}(d_1, \ldots, d_n)$.

Ahora bien, como F_Σ es un funtor constante y f es un homomorfismo entre las F_Σ-coálgebras $(\mathcal{X},\gamma)$ y $(\mathcal{X}',\gamma')$, se verifica que $\gamma = \gamma' \circ f$ lo que completa la demostración. ■

Comprobamos a continuación que la familia de aplicaciones que acabamos de definir es una familia de funtores.

Lema 2.6.9. *La aplicación $\beta_\Sigma\colon Mod_{CoAlg(\mathcal{E}^D)}(\Sigma) \to Mod_{\mathcal{H}^D}(\Phi(\Sigma))$ es un funtor para cada signatura Σ de $SIG_{CoAlg(\mathcal{E}^D)}$.*

Demostración. Dada una signatura Σ de $SIG_{CoAlg(\mathcal{E}^D)}$, tenemos que β_Σ respeta identidades. La imagen del homomorfismo identidad de una coálgebra es el homomorfismo identidad entre la correspondiente álgebra oculta, ya que la función correspondiente al género oculto es la identidad.

Similarmente, esta aplicación respeta la composición de dos homomorfismos de coálgebras a través de la composición de la función correspondiente a este género oculto. ∎

Por último, recogemos en el siguiente lema, que la familia de funtores anterior define una transformación natural. La demostración del mismo es inmediata ya que los únicos morfismos de $SIG_{CoAlg(\mathcal{E}^D)}$ son las identidades.

Lema 2.6.10. *La familia de funtores $\beta\colon Mod_{CoAlg(\mathcal{E}^D)} \Rightarrow Mod_{\mathcal{H}^D} \circ \Phi$ constituye una transformación natural.*

Una vez definidas las tres componentes que definen un morfismo de instituciones unidireccional podemos enunciar el siguiente teorema.

Teorema 2.6.11. *La terna (Φ, α, β) definida anteriormente constituye un morfismo de instituciones unidireccional entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$.*

Demostración. Para demostrar que estas operaciones constituyen un morfismo de instituciones unidireccional únicamente falta por comprobar que se verifica la condición de satisfacibilidad, es decir, que para cada signatura Σ de $SIG_{CoAlg(\mathcal{E}^D)}$, se verifica que

$$(\mathcal{X}, \gamma) \models_\Sigma^{CoAlg(\mathcal{E}^D)} e \quad \text{si y sólo si} \quad \beta_\Sigma((\mathcal{X}, \gamma)) \models_{\Phi(\Sigma)}^{\mathcal{H}^D} \alpha_\Sigma(e)$$

para cualquier sentencia $e \in Sen_{CoAlg(\mathcal{E}^D)}(\Sigma)$ y cualquier coálgebra $(\mathcal{X}, \gamma) \in Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$.

Sean $\Sigma = (S, \Omega \cup C)$ una signatura de $SIG_{CoAlg(\mathcal{E}^D)}$, $e = (X, t, u)$ una ecuación de $Sen_{CoAlg(\mathcal{E}^D)}(\Sigma)$, con $t, u \in T_{\Sigma(X),s}$, $s \in S$, y sea $(\mathcal{X}, \gamma)$ una coálgebra de $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$.

Tenemos por un lado que

$$(\mathcal{X}, \gamma) \models_\Sigma^{CoAlg(\mathcal{E}^D)} (X, t, u) \quad \text{si y sólo si} \quad \gamma(a) \models_\Sigma^{\mathcal{E}^D} (X, t, u),\ \forall a \in \mathcal{X},$$

sin más que aplicar la definición de $\models_\Sigma^{CoAlg(\mathcal{E}^D)}$.

Por otro lado tenemos que

$$\beta_\Sigma((\mathcal{X}, \gamma)) \models_{\Phi(\Sigma)}^{\mathcal{H}^D} \alpha_\Sigma((X, t, u)) \quad \text{si y sólo si} \quad \beta_\Sigma((\mathcal{X}, \gamma)) \models_{\Phi(\Sigma)}^{\mathcal{E}^D} (X \cup \{z_{imp}\}, \theta(t), \theta(u)),$$

ya que $\alpha_\Sigma((X, t, u)) = (X \cup \{z_{imp}\}, \theta(t), \theta(u))$ es una ecuación visible, con z_{imp} una nueva variable que no está presente en X.

Entonces, para demostrar el teorema es suficiente con demostrar:

$$\gamma(a) \models^{\mathcal{E}^D}_{\Sigma} (X, t, u),\ \forall a \in \mathcal{X} \quad \text{si y sólo si} \quad \beta_{\Sigma}((\mathcal{X}, \gamma)) \models^{\mathcal{E}^D}_{\Phi(\Sigma)} (X \cup \{z_{imp}\}, \theta(t), \theta(u)).$$

Nos centramos primero en la demostración de la implicación directa.

Suponemos que $\gamma(a) \models^{\mathcal{E}^D}_{\Sigma} (X, t, u),\ \forall a \in \mathcal{X}$. Demostraremos que $\beta_{\Sigma}((\mathcal{X}, \gamma)) \models^{\mathcal{E}^D}_{\Phi(\Sigma)}$ $(X \cup \{z_{imp}\}, \theta(t), \theta(u))$ probando que para cada valoración $q\colon X \cup \{z_{imp}\} \to \beta_{\Sigma}((\mathcal{X}, \gamma))$, se verifica que

$$\overline{q}(\theta(t)) = \overline{q}(\theta(u)).$$

Sea $q\colon X \cup \{z_{imp}\} \to \beta_{\Sigma}((\mathcal{X}, \gamma))$ una valoración, entonces $q(z_{imp}) \in \mathcal{X}$ y por hipótesis $\gamma(q(z_{imp})) \models^{\mathcal{E}^D}_{\Sigma} (X, t, u)$. Además la restricción de q a los géneros visibles en S, $q_{|S} : X \to \beta_{\Sigma}((\mathcal{X}, \gamma))_{|S}$, es una valoración para la Σ-álgebra $\gamma(q(z_{imp}))$. Si utilizamos esta valoración obtenemos que

$$\overline{q_{|S}}(t) = \overline{q_{|S}}(u).$$

Entonces es suficiente probar que para cada uno de los dos términos de la ecuación t y u, se verifica que

$$\overline{q}(\theta(t)) = \overline{q_{|S}}(t).$$

Demostraremos la igualdad anterior por inducción sobre la estructura de un término $t \in T_{\Sigma(X),s}$, $s \in S$, que pueda formar parte de una ecuación $(X, t, u) \in Sen_{CoAlg(\mathcal{E}^D)}(\Sigma)$:

- si $t = x \in X_s$, entonces $\overline{q}(\theta(x)) = \overline{q}(x) = q(x) = q_{|S}(x) = \overline{q_{|S}}(x)$;
- si $t = d$, con $d\colon \to s$ una constante de C, entonces $\overline{q}(\theta(d)) = \overline{q}(d) = d = \overline{q_{|S}}(d)$;
- y si $t = \omega(t_1, \ldots, t_n)$, con $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 0$ y $t_i \in T_{\Sigma(X),s_i}$, $i = 1, \ldots, n$, entonces

$$\begin{aligned}
&\overline{q}(\theta(\omega(t_1, \ldots, t_n))) = \\
&\quad = \overline{q}(imp_\omega(z_{imp}, \theta(t_1), \ldots, \theta(t_n))) \\
&\quad = imp_\omega_{\beta_{\Sigma}((\mathcal{X},\gamma))}(\overline{q}(z_{imp}), \overline{q}(\theta(t_1)), \ldots, \overline{q}(\theta(t_n))) \\
&\quad = imp_\omega_{\beta_{\Sigma}((\mathcal{X},\gamma))}(q(z_{imp}), \overline{q}(\theta(t_1)), \ldots, \overline{q}(\theta(t_n))) \\
&\quad = \omega_{\gamma(q(z_{imp}))}(\overline{q}(\theta(t_1)), \ldots, \overline{q}(\theta(t_n))) \text{ por definición de } \beta_{\Sigma}((\mathcal{X}, \gamma)), \\
&\quad = \omega_{\gamma(q(z_{imp}))}(\overline{q_{|S}}(t_1), \ldots, \overline{q_{|S}}(t_n)) \text{ por hipótesis de inducción,} \\
&\quad = \overline{q_{|S}}(\omega(t_1, \ldots, t_n)).
\end{aligned}$$

Demostramos a continuación la implicación inversa.

Suponemos que $\beta_{\Sigma}((\mathcal{X}, \gamma)) \models^{\mathcal{E}^D}_{\Phi(\Sigma)} (X \cup \{z_{imp}\}, \theta(t), \theta(u))$. Tenemos que demostrar si $\gamma(a) \models^{\mathcal{E}^D}_{\Sigma} (X, t, u),\ \forall a \in \mathcal{X}$. Es decir, probar que para cada $a \in \mathcal{X}$ y cada valoración $p\colon X \to \gamma(a)$ se verifica que $\overline{p}(t) = \overline{p}(u)$.

Sean $a \in \mathcal{X}$ y $p\colon X \to \gamma(a)$ una valoración, entonces la aplicación $q\colon X \cup \{z_{imp}\} \to \beta_\Sigma((\mathcal{X}, \gamma))$, que definimos por $q(z_{imp}) = a$ y $q(x) = p(x)$ para cada $x \in X$, es trivialmente una valoración para $\beta_\Sigma((\mathcal{X}, \gamma))$. Entonces, por hipótesis se cumple que $\overline{q}(\theta(t)) = \overline{q}(\theta(u))$.

Si aplicamos ahora inducción estructural sobre los términos de forma análoga a lo hecho en la implicación anterior, obtenemos nuevamente que, para cualquier término t que pueda formar parte de una ecuación $(X, t, u) \in Sen_{CoAlg(\mathcal{E}^D)}(\Sigma)$, se verifica que $\overline{q}(\theta(t)) = \overline{p}(t)$, lo que concluye la prueba. ■

2.7 Diagrama de morfismos de instituciones unidireccionales

En esta sección vamos a realizar una recopilación de los distintos morfismos de instituciones unidireccionales que hemos desarrollado a lo largo del capítulo. Además estableceremos las relaciones que existen entre los mismos de modo que conseguiremos construir un diagrama conmutativo con dichos morfismos.

Como hemos comentado, nuestro interés en el ámbito de las instituciones surge inicialmente de la búsqueda de formalismos que nos permitieran el modelado formal de las estructuras de datos de EAT. Esto nos ha llevado a interpretar nuestros resultados desde diferentes perspectivas: oculto, coálgebras, categorías, etc. En este capítulo hemos establecido pasos (morfismos) entre las correspondientes instituciones. El primer marco de especificación en el que inicialmente pretendimos explicar estas ideas ha sido el de la especificación algebraica ecuacional, que denotamos por $\mathcal{E}$. En [29] definimos un endomorfismo de $\mathcal{E}$ que recogía las ideas básicas de nuestra construcción sobre tipos de datos. No obstante, como hemos comentado $\mathcal{E}$ no es el marco adecuado debido fundamentalmente a la necesidad de fijar un sustrato de datos sobre los que construir estas estructuras de datos. Hemos considerado de esta forma una nueva institución, la institución algebraica ecuacional definida sobre un universo de datos $\mathcal{E}^D$, que incorpora un dominio de datos prefijado. Entonces se ha modificado el anterior endomorfismo para obtener un morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{E}$.

Como en realidad pretendemos modelar familias de estas estructuras de datos, resulta necesario distinguir entre dos tipos de géneros: los que corresponden a estos datos fijados, y los que representarán las estructuras sobre esos datos. En la literatura se ha desarrollado un marco de especificación que incorpora esta distinción entre géneros como una de sus características fundamentales. Se trata de la teoría de las especificaciones ocultas que queda recogida en la institución oculta $\mathcal{H}^D$. Al renombrar las componentes del morfismo anterior para situarlo dentro de esta institución, se obtiene un morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$ que nos ha permitido dar una buena modelización de nuestro tratamiento de las estructuras.

Relacionada con la teoría oculta se encuentra la teoría de las coálgebras y hemos expresado nuestros resultados en términos de esta teoría. Para ello, a partir de una institución $\mathcal{I}$, hemos definido una institución coalgebraica $CoAlg(\mathcal{I})$ para coálgebras (particulares) adecuada a nuestras necesidades, junto con un morfismo de instituciones unidireccional entre ambas que se obtiene de manera natural. Con esta nueva construcción definimos otro morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $CoAlg(\mathcal{E}^D)$ que permite obtener un modelado equivalente de las estructuras de datos presentes en EAT.

Si ponemos en común los anteriores morfismos de instituciones unidireccionales junto con el morfismo de instituciones que relaciona $\mathcal{H}^D$ con $\mathcal{E}$ y $CoAlg(\mathcal{E}^D)$, obtenemos el siguiente diagrama que resulta ser conmutativo:

$$\begin{array}{ccc} & & CoAlg(\mathcal{E}^D) \\ & \nearrow & \downarrow \\ \mathcal{E}^D & \longrightarrow & \mathcal{H}^D \\ & \searrow & \downarrow \\ & & \mathcal{E} \end{array}$$

Dividimos la demostración de la conmutatividad del diagrama anterior en dos partes. En el primer resultado recogemos que la sección inferior del diagrama es conmutativo.

Teorema 2.7.1. $\Phi^{\mathcal{H}^D-\mathcal{E}} \circ \Phi^{\mathcal{E}^D-\mathcal{H}^D} = \Phi^{\mathcal{E}^D-\mathcal{E}}$.

Demostración. Debemos probar que conmuta tanto los funtores de signaturas como las transformaciones naturales de sentencias y modelos.

Probamos primero que los funtores de signaturas conmutan. Dada una signatura $\Sigma = (S, \Omega \cup C) \in Obj(SIG_{\mathcal{E}^D})$, entonces $\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma) = \Sigma_{imp}$ es una signatura oculta sobre la signatura visible (S, C) y el álgebra de datos D, entonces $\Phi^{\mathcal{H}^D-\mathcal{E}}(\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma)) = \Phi^{\mathcal{H}^D-\mathcal{E}}(\Sigma_{imp}) = \Sigma_{imp}$ es una signatura en la que olvidamos la estructura oculta. Esta signatura es exactamente $\Phi^{\mathcal{E}^D-\mathcal{E}}(\Sigma)$. Para los morfismos de signaturas conmuta trivialmente ya que son identidades.

Las transformaciones naturales para las sentencias conmutan, es decir $\alpha^{\mathcal{H}^D-\mathcal{E}} \circ \alpha^{\mathcal{E}^D-\mathcal{H}^D} = \alpha^{\mathcal{E}^D-\mathcal{E}}$, ya que por definición $\alpha^{\mathcal{H}^D-\mathcal{E}}$ es la identidad y $\alpha^{\mathcal{E}^D-\mathcal{H}^D} = \alpha^{\mathcal{E}^D-\mathcal{E}}$.

Por último probamos que las transformaciones naturales para las sentencias conmutan, es decir, si para cada signatura Σ de $SIG_{\mathcal{E}^D}$, se tiene que

$$\beta^{\mathcal{H}^D-\mathcal{E}}_{\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma)} \circ \beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma} = \beta^{\mathcal{E}^D-\mathcal{E}}_{\Sigma}.$$

Dada un álgebra A de $Mod_{\mathcal{E}^D}(\Sigma)$, entonces $\beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma}(A)$ es la Σ_{imp}-álgebra oculta definida por el conjunto unipuntual, $\{*\}$, sobre el único género oculto. Entonces $\beta^{\mathcal{H}^D-\mathcal{E}}_{\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma)}(\beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma}(A)) = \beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma}(A)$, ya que el cociente que define $\beta^{\mathcal{H}^D-\mathcal{E}}_{\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma)}$ sobre $\beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma}(A)$ es la identidad en la parte visible y el conjunto asociado al género oculto está formado por un único elemento. Entonces obtenemos la conmutatividad de los funtores sobre los objetos ya que hemos definido $\beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma}(A) = \beta^{\mathcal{E}^D-\mathcal{E}}_{\Sigma}(A)$.

Sobre cada homomorfismo identidad id_A de $Mod_{\mathcal{E}^D}(\Sigma)$, obtenemos que los funtores anteriores trivialmente conmutan. ■

Probamos a continuación que conmuta la sección superior del diagrama.

Teorema 2.7.2. $\Phi^{CoAlg(\mathcal{E}^D)-\mathcal{H}^D} \circ \Phi^{\mathcal{E}^D-CoAlg(\mathcal{E}^D)} = \Phi^{\mathcal{E}^D-\mathcal{H}^D}$.

Demostración. Por definición, es inmediato comprobar que conmutan los funtores de signaturas y las transformaciones naturales sobre sentencias, ya que estas componentes entre $\mathcal{E}^D$ y $CoAlg(\mathcal{E}^D)$ están definidas como identidades y las componentes respectivas para las otras integrantes del diagrama están definidas de la misma forma.

Falta por demostrar la conmutatividad de las trasformaciones naturales sobre modelos, es decir, si para cada Σ de $SIG_{\mathcal{E}^D}$, se tiene que

$$\beta^{CoAlg(\mathcal{E}^D)-\mathcal{H}^D}_{\Phi^{\mathcal{E}^D-CoAlg(\mathcal{E}^D)}(\Sigma)} \circ \beta^{\mathcal{E}^D-CoAlg(\mathcal{E}^D)}_{\Sigma} = \beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma}.$$

Dada un álgebra A de $Mod_{\mathcal{E}^D}(\Sigma)$, entonces $\beta^{\mathcal{E}^D-CoAlg(\mathcal{E}^D)}_{\Sigma}(A)$ es la F_Σ-coálgebra $(\{*\}, \gamma_A \colon \{*\} \to Obj(Mod_{\mathcal{E}^D}(\Sigma))$ definida por $\gamma_A(*) = A$. Entonces $\beta^{CoAlg(\mathcal{E}^D)-\mathcal{H}^D}_{\Sigma}((\{*\}, \gamma_A))$ es la Σ_{imp}-álgebra que tiene el conjunto $\{*\}$ como soporte para el género oculto y con soportes para los géneros visibles los respectivos soportes de A. Las funciones para cada operación de esta álgebra están definidas de la manera correspondiente a partir de las funciones de A. Esta álgebra es exactamente la Σ_{imp}-álgebra definida por $\beta^{\mathcal{E}^D-\mathcal{H}^D}_{\Sigma}(A)$.

Dado un homomorfismo identidad id_A de $Mod_{\mathcal{E}^D}(\Sigma)$, tenemos trivialmente la conmutatividad buscada, ya que obtenemos un Σ_{imp}-homomorfismo identidad también sobre el conjunto unipuntual en el género oculto de las correspondientes Σ_{imp}-álgebras. ■

Capítulo 3

Diagramas alternativos

3.1 Introducción

En el capítulo anterior obtuvimos un diagrama de morfismos de instituciones unidireccionales. A través de éste pretendemos interpretar la modelización de las estructuras de datos presentes en EAT desde distintas perspectivas. No obstante, dicho diagrama requiere algunas discusiones. Por un lado, tenemos que la institución que actúa como origen de los tres morfismos principales es la institución algebraica ecuacional definida sobre un universo de datos, que denotamos por $\mathcal{E}^D$. Observamos que esta institución es muy limitada, no sólo por presentar un dominio de datos prefijado, sino también porque las posibles relaciones entre signaturas de la institución se limitan a identidades entre las mismas. Partir de un dominio de datos prefijado no puede ser considerado verdaderamente una limitación, ya que, desde un punto de vista de las aplicaciones a la programación concreta, la idea es que representan a los tipos de datos predefinidos en un lenguaje de programación. Sin embargo, exigir que los morfismos entre signaturas se limiten a identidades nos obliga a prescindir de relaciones muy naturales entre signaturas como pueden ser las inclusiones de signaturas. Un primer motivo de discusión es, entonces, estudiar la causa por la cual esta institución es tan limitada y si es posible introducir en su lugar una institución algo más general. Por otro lado, el tipo de morfismo de instituciones que se ha elegido para establecer las relaciones entre las mismas es el morfismo de instituciones unidireccional y no el habitual morfismo de instituciones. Otra cuestión que surge entonces es si podemos redefinir el diagrama utilizando esta noción habitual de morfismo o cualquier otro de los tipos de morfismos entre instituciones que aparecen en la literatura. En este capítulo discutiremos estas y otras cuestiones que surgieron al desarrollar el diagrama anterior. Para ello dividimos el capítulo en dos secciones. En la primera sección nos centraremos en el estudio de los nodos que forman el diagrama, mientras que en la segunda la dedicaremos a reflexionar sobre el tipo de morfismos que lo integran.

3.2 Estudio de los nodos del diagrama

Nos centramos ahora en el estudio las instituciones que aparecen como nodos en el diagrama presentado en el capítulo anterior. El objetivo es valorar sus características y discutir posibles modificaciones en las mismas que nos permitan obtener un diagrama más general.

Las instituciones que juegan un papel más importante dentro de nuestro diagrama son la institución algebraica ecuacional definida sobre un universo de datos $\mathcal{E}^D$ y la institución ecuacional oculta definida sobre un universo de datos $\mathcal{H}^D$. La primera es la institución de origen de los tres morfismos con los que pretendemos modelar la construcción *imp* y supone el marco en el que representamos el sustrato de datos sobre el que se realiza esta construcción. La segunda representa posiblemente el marco más adecuado en el que explicar los resultados de dicha construcción. Comenzaremos entonces esta sección analizando primero la institución $\mathcal{H}^D$ y posteriormente la institución $\mathcal{E}^D$.

Para estudiar la institución oculta $\mathcal{H}^D$ realizamos a continuación una comparación de la misma con otras instituciones ocultas que aparecen en la literatura.

3.2.1 Otras instituciones ocultas

Si revisamos la literatura podemos encontrar distintas instituciones ocultas "básicas" dependiendo fundamentalmente del papel que juegue el dominio de datos D fijado.

La primera referencia a este tipo de instituciones aparece en [38]. En este artículo se define una institución oculta en la que no aparece prefijado un dominio de datos para todas las signaturas ocultas de la institución, ni siquiera una signatura visible común, sino que cada signatura oculta lo es sobre una signatura visible particular. Se define entonces un morfismo de signaturas oculto entre dos signaturas ocultas que en principio pueden venir definidas sobre signaturas visibles distintas. Esto permite pensar en una institución en la que cada signatura oculta $H\Sigma$ lo es sobre un álgebra de datos D diferente. No obstante, puede ser interesante partir de un universo de datos prefijado que corresponda con el dominio de todas las posibles álgebras de datos que se pueden utilizar, restricción que como se ha comentado, parece natural cuando pretendemos fijar los tipos de datos de base en nuestro sistema de software.

Por otro lado en [41, 15, 22] se presenta otra institución oculta[1] en la que partiendo de un álgebra de datos D prefijada y una signatura visible para D, todas las signaturas ocultas de la institución comparten esta signatura y dominio de datos. Los morfismos de signaturas ocultos son morfismos de signaturas que entre otras propiedades deben ser identidades sobre la parte visible de las signaturas. Si consideramos esta institución oculta en nuestro diagrama de morfismos de instituciones nos vemos obligados a incluir

[1] En una versión posterior del artículo [38] publicada en la página Web personal del autor, se reconoce haber omitido de forma involuntaria parte del material que contiene esta nueva versión. En particular, se cambia la institución oculta que aparece en [38] por la institución que comparten estos otros artículos.

como constantes en nuestras signaturas todos los datos del universo, lo cual es defendible si se considera que cualquier dato del universo está definido. No obstante, supone o bien manejar signaturas enormes de forma innecesaria ya que incluimos datos que dichas signaturas no van a utilizar al no presentar operaciones sobre los mismos, o bien reducir el universo de datos a un conjunto pequeño, de forma que restringimos las posibles signaturas ocultas a aquellas que utilizan esos datos.

La institución oculta que hemos adoptado supone una posición intermedia entre ambas. Partimos de un universo de géneros visibles y un universo de datos D para estos géneros prefijados, entonces las signaturas ocultas pueden estar definidas sobre diferentes signaturas visibles y dominios de datos cuyos géneros y conjuntos están elegidos entre los prefijados por el universo. De este modo no incluimos en todas las signaturas todos los datos de D, sino sólo aquéllos para los cuales existan operaciones que permitan utilizarlos. No obstante, debemos mantener la restricción de que si dos signaturas están relacionadas mediante un morfismo de signaturas, entonces estas signaturas deben compartir la misma signatura visible y dominio de datos.

3.2.2 Estudio de la institución $\mathcal{E}^D$

En esta sección estudiamos con más detalle la institución algebraica ecuacional definida sobre un universo de datos, $\mathcal{E}^D$. El objetivo de la sección es aclarar cuál es el motivo de las restricciones que esta institución presenta con respecto a la institución algebraica ecuacional y si es posible relajar alguna de sus condiciones.

El papel que juega la institución $\mathcal{E}^D$ en el diagrama anterior es el de modelar, dentro del marco de especificación algebraica ecuacional, estructuras de datos de las que denominábamos de *segundo tipo*, es decir, estructuras de datos como grupos, anillos o conjuntos simpliciales. Estas estructuras se construyen sobre estructuras de datos de *primer tipo* que se corresponden con los datos usuales, como números o listas. No obstante, en el sistema EAT y en general en los paquetes de Computación Simbólica, no se trabaja con una única estructura de segundo tipo, un grupo por ejemplo, sino que se manejan varios grupos a la vez que se construyen sobre un mismo conjunto de datos. De esta forma se presenta un nuevo tipo de dato que representa a este conjunto de grupos. Los morfismos anteriores entre instituciones, que parten de $\mathcal{E}^D$, se encargan de hacer explícito este tipo invisible u oculto (la operación de hacer explícito este nuevo género la denominábamos operación *imp*). Es por ello por lo que considerábamos que el morfismo que refleja de forma más adecuada esta operación es el que tiene como institución de llegada a la institución oculta $\mathcal{H}^D$. Las características particulares de la institución $\mathcal{H}^D$ nos obligan a introducir restricciones en la institución $\mathcal{E}^D$ necesarias para poder obtener este morfismo de instituciones unidireccional.

A continuación vamos a analizar brevemente las peculiaridades de esta operación *imp*, como una construcción dentro del marco de especificación algebraico ecuacional, prescindiendo de las restricciones de la institución $\mathcal{E}^D$. En este marco más sencillo distinguiremos las características, tanto de tipo sintáctico como semántico que son inherentes a la operación *imp*, de las restricciones que debemos incluir por situarnos dentro del marco que nos proporciona la institución oculta. Históricamente comenzamos el estudio del problema de este modo y desarrollamos un endomorfismo de la institución algebraica ecuacional $\mathcal{E}$ que podemos encontrar en [29].

Comenzamos recordando la definición, a un nivel puramente sintáctico, de la operación *imp*.

Dada una signatura $\Sigma = (S, \Omega)$, se construye una signatura Σ_{imp} dada por (S_{imp}, Ω_{imp}). El conjunto de géneros $S_{imp} = \{imp_{\Sigma}\} \cup S$, con imp_{Σ} un nuevo género no presente en S. Y para cada operación $\omega\colon s_1 \dots s_n \to s \in \Omega$, la operación $imp\_\omega\colon imp_{\Sigma} s_1 \dots s_n \to s \in \Omega_{imp}$.

Si interpretamos esta última signatura en el marco oculto, es lógico definir los géneros de S como géneros visibles. Estos géneros permanecen presentes en todo momento y modelan los datos del *primer tipo*. Entonces, se debe verificar que $S \subseteq U$, donde U es el universo de géneros (visibles). Ésta es la primera condición que exigíamos a las signaturas de $\mathcal{E}^D$. Además, una vez prefijado el dominio de datos D, la signatura oculta Σ_{imp} debe incluir una constante $d\colon \to s$ por cada elemento $d \in D_s$, con $s \in S$. Estas constantes se corresponden con los posibles datos particulares (visibles) que podemos utilizar. Es por ello natural, aunque no necesario, que incluyamos también estos elementos en la signatura Σ de $\mathcal{E}^D$.

Dado un morfismo de signaturas $\mu\colon \Sigma \to \Sigma'$, entre las signaturas $\Sigma = (S, \Omega)$ y $\Sigma' = (S', \Omega')$, es posible extender μ a un morfismo $\mu_{imp}\colon \Sigma_{imp} \to \Sigma'_{imp}$. Este morfismo se define por la función sobre los géneros $\mu_{impS}(s) := \mu_S(s)$, si $s \in S$ y $\mu_{impS}(imp_{\Sigma}) := imp_{\Sigma'}$ y la función sobre las operaciones $\mu_{imp\Omega}(imp\_\omega) := imp\_\omega'$, con $\omega' = \mu_{\Omega}(\omega)$.

Si situamos el morfismo de signaturas μ_{imp} en el marco oculto, debemos exigir, por un lado, que este morfismo de signaturas sea la identidad sobre la parte visible. Esta condición nos lleva a imponer que la función μ_S sea la identidad, por lo que $S = S'$. Además, para cada operación $imp\_\omega'\colon imp_{\Sigma'} s'_1 s'_2 \dots s'_n \to s'$, $n \geq 0$, de Σ'_{imp} se verifica que su primer argumento es imagen de un género oculto de Σ_{imp}. Entonces, debemos exigir que exista una operación imp_ω en Σ_{imp} tal que $imp\_\omega' = \mu_{imp\Omega}(imp\_\omega)$. Si traducimos esta condición al nivel del morfismo de signaturas μ, obtenemos que para cada operación ω' de Σ' debe existir una operación ω de Σ tal que $\omega' = \mu_{\Omega}(\omega)$. Es decir, la función μ_{Ω} debe ser sobreyectiva.

Observamos que en la institución $\mathcal{E}^D$ restringíamos los morfismos de signaturas a morfismos identidad, mientras que para nuestros objetivos es suficiente, según el párrafo anterior, exigir a los morfismos que su función sobre géneros sea la identidad y que su función sobre operaciones sea sobreyectiva. Así, podemos ampliar nuestra institución de origen para que incluya estos morfismos de signaturas de modo que obtenemos igualmente un morfismo de instituciones entre ella y $\mathcal{H}^D$.

Nos centramos a continuación en las características de tipo semántico de la operación *imp*.

Dada una signatura $\Sigma = (S, \Omega)$, a cada Σ-álgebra A le asociamos una Σ_{imp}-álgebra A_{imp} que definimos por los conjuntos $(A_{imp})_{imp_\Sigma} := \{*\}$ en el nuevo género, y $(A_{imp})_s := A_s$ para el resto de los géneros $s \in S$. La interpretación de las operaciones de Σ_{imp} en A_{imp} es la natural.

Cuando situamos esta álgebra en el marco oculto es necesario que los conjuntos asociados a los géneros visibles sean los conjuntos correspondientes en el domino de datos prefijado D. Estos conjuntos representan conjuntos de datos del *primer tipo*, por lo tanto prefijados y conocidos, sobre los que vamos a trabajar. Es por ello lógico y necesario exigir a las Σ-álgebras de $\mathcal{E}^D$ la condición $A_s = D_s$, para cada $s \in S$.

Por otro lado, a cada morfismo de Σ-álgebras $f = (f_s)_{s\in S} \colon A \to B$, la construcción *imp* le asocia el morfismo $f_{imp} \colon A_{imp} \to B_{imp}$ que se define por la familia de funciones $(id_{\{*\}}, (f_s)_{s\in S})$. Nuevamente, al situarnos dentro del marco oculto, es necesario que las funciones del morfismo sean la identidad sobre los géneros visibles. Por ello, es necesario imponer que los morfismos de Σ-álgebras sean identidades.

Observamos que todas las condiciones que hemos exigido a $\mathcal{E}^D$ se justifican de una manera lógica por el papel que juega esta institución dentro del morfismo que la relaciona con $\mathcal{H}^D$, excepto la condición que indica que los morfismos entre signaturas deben ser identidades. En realidad, según lo comentado anteriormente, podemos elegir morfismos de signaturas algo más generales, por ejemplo, morfismos de signaturas cuya aplicación sobre los géneros sea la identidad y sobre las operaciones sea sobreyectiva. Sin embargo estos morfismos tampoco cubren morfismos de signaturas muy naturales como pueden ser las inclusiones entre dos signaturas como la que mostramos en el siguiente ejemplo.

Ejemplo 3.2.1. Las siguientes signaturas constituyen, respectivamente, la base para la especificación algebraica de un semigrupo y de un grupo construidos sobre un mismo conjunto D_e:

signatura `SGRP`
 géneros e
 operaciones
  $* \colon e\ e \to e$
finsig,

signatura `GRP`
 géneros e
 operaciones
  $prd \colon e\ e \to e$
  $inv \colon e \to e$
  $unt \colon\ \to e$
finsig.

Podemos definir el morfismo de signaturas inclusión $i \colon \mathtt{SGRP} \hookrightarrow \mathtt{GRP}$.

Consideramos ahora las signaturas ocultas $\mathtt{SGRP}_{imp}$ y $\mathtt{GRP}_{imp}$ que se construyen a partir de `SGRP` y `GRP` por la operación *imp*:

signatura $\texttt{SGRP}_{imp}$
  géneros e, $imp_{\texttt{SGRP}}$
  operaciones
    $imp\_*\colon imp_{\texttt{SGRP}}\ e\ e \to e$
finsig,

signatura $\texttt{GRP}_{imp}$
  géneros e, $imp_{\texttt{GRP}}$
  operaciones
    $imp\_prd\colon imp_{\texttt{GRP}}\ e\ e \to e$
    $imp\_inv\colon imp_{\texttt{GRP}}\ e \to e$
    $imp\_unt\colon imp_{\texttt{GRP}} \to e$
finsig.

Entonces, el morfismo de signaturas inclusión entre $\texttt{SGRP}_{imp}$ y $\texttt{GRP}_{imp}$, $i_{imp}\colon \texttt{SGRP}_{imp} \hookrightarrow \texttt{GRP}_{imp}$, que se obtiene al aplicar la operación *imp* al morfismo i, no es un morfismo de signaturas oculto. La signatura oculta $\texttt{GRP}_{imp}$ tiene operaciones ocultas que no son imagen de ninguna operación oculta en $\texttt{SGRP}_{imp}$, entonces no se verifica la cuarta propiedad de la definición de este tipo de morfismos. □

Trataremos a continuación de generalizar la institución $\mathcal{E}^D$, con las características mínimas necesarias para poder situar nuestra construcción dentro de un marco oculto, de modo que incluyamos además los morfismos de signaturas anteriores.

3.2.3 Generalización de la institución $\mathcal{E}^D$

En principio parece conveniente incluir dentro de la institución $\mathcal{E}^D$ morfismos de signaturas del tipo de inclusión que hemos presentado en el ejemplo de la sección anterior. No obstante, nuestro objetivo ha sido relacionar los morfismos de esta institución con los de una institución oculta, de modo que tal relación represente la operación *imp*. Así, si pensamos en el papel que juega un morfismo de signaturas de $\mathcal{E}^D$ dentro de esta construcción, parece imprescindible mantener la restricción que impone que la aplicación correspondiente a los géneros de este morfismo de signaturas sea la identidad. Esto es así ya que se corresponde con la aplicación sobre los géneros visibles de un morfismo de signaturas oculto y todos los autores que abordan estos temas están de acuerdo en que dicha aplicación en un morfismo de signaturas oculto sea la identidad. Donde existe más controversia es en la cuarta condición que se exige a un morfismo de signaturas oculto. Esta condición, denominada *condición de encapsulamiento*, no permite definir *nuevas* operaciones ocultas en la signatura de llegada que contengan entre sus argumentos un género oculto que sea imagen de un género oculto de la signatura de salida. Los propios autores fundadores de la teoría oculta reconocen en [41] haber recibido críticas sobre esta condición por considerarla muy restrictiva. No obstante, es imprescindible para poder demostrar la condición de satisfacibilidad de la institución oculta $\mathcal{H}^D$. Es exactamente esta condición la que nos impide incluir el morfismo de signaturas inclusión, referido en el Ejemplo 3.2.1, en $\mathcal{E}^D$.

En esta sección generalizaremos la institución $\mathcal{E}^D$ de modo que ampliamos la clase de sus morfismos de signaturas, permitiendo morfismos que no sean identidades entre las operaciones de las signaturas. A continuación generalizaremos la institución oculta $\mathcal{H}^D$ para mantener la relación entre ambas.

Consideramos de nuevo fijados un conjunto U, que es el universo de géneros, y una familia de conjuntos $D = \{D_s\}_{s\in U}$, que es el universo de datos. Estos conjuntos permanecerán fijados a lo largo del capítulo, salvo que se indique lo contrario.

Definimos la *institución algebraica ecuacional generalizada definida sobre un universo de datos D*, que denotaremos por $\bar{\mathcal{E}}^D = (SIG_{\bar{\mathcal{E}}^D}, Sen_{\bar{\mathcal{E}}^D}, Mod_{\bar{\mathcal{E}}^D}, \models^{\bar{\mathcal{E}}^D})$, de la siguiente forma:

La categoría $SIG_{\bar{\mathcal{E}}^D}$ tiene como objetos las signaturas $\Sigma = (S, \Omega)$, donde el conjunto S de géneros satisface que $S \subseteq U$. Los morfismos de $SIG_{\bar{\mathcal{E}}^D}$ son los morfismos de signaturas cuya componente correspondiente a la aplicación entre géneros es la identidad.

El funtor $Mod_{\bar{\mathcal{E}}^D} \colon SIG_{\bar{\mathcal{E}}^D} \to Cat^{op}$ asocia a cada objeto $\Sigma = (S, \Omega)$ de $SIG_{\bar{\mathcal{E}}^D}$, la categoría $Mod_{\bar{\mathcal{E}}^D}(\Sigma)$. Esta categoría tiene como objetos las Σ-álgebras A de modo que verifiquen que $A_s = D_s$, para cada género $s \in S$. Los morfismos de la categoría $Mod_{\bar{\mathcal{E}}^D}(\Sigma)$ son los Σ-homomorfismos identidad. Por su parte, si μ es un morfismo de signaturas de $SIG_{\bar{\mathcal{E}}^D}$, $Mod_{\bar{\mathcal{E}}^D}(\mu)$ es el funtor definido al considerar los correspondientes reductos de álgebras.

El funtor sentencias $Sen_{\bar{\mathcal{E}}^D}$ y la relación de satisfacción $\models^{\bar{\mathcal{E}}^D}$ se definen de la misma forma que las respectivas componentes de la institución algebraica ecuacional $\mathcal{E}$, sobre cada signatura o morfismo de signaturas de $SIG_{\bar{\mathcal{E}}^D}$.

Es inmediato comprobar que las cuatro componentes anteriores están bien definidas y que verifican la condición de satisfacibilidad, lo que se recoge en el siguiente teorema.

Teorema 3.2.2. *La tupla $\bar{\mathcal{E}}^D = (SIG_{\bar{\mathcal{E}}^D}, Sen_{\bar{\mathcal{E}}^D}, Mod_{\bar{\mathcal{E}}^D}, \models^{\bar{\mathcal{E}}^D})$ constituye una institución.*

3.2.4 Generalización de la institución $\mathcal{H}^D$

Si pretendemos redefinir nuestro morfismo de instituciones unidireccional de modo que tengamos como origen la institución $\bar{\mathcal{E}}^D$ que hemos definido en la sección anterior, es necesario generalizar también la institución $\mathcal{H}^D$. Esta institución debe incluir al menos los morfismos de signaturas ocultos que se obtienen por la operación *imp* a partir de los morfismos de signaturas de $\bar{\mathcal{E}}^D$.

El origen de nuestra restricción sobre los morfismos de signaturas se encuentra en la cuarta condición que se impone a un morfismo de signaturas oculto. Debemos notar que asociada a esta condición es necesario exigir que la aplicación correspondiente a los géneros sea inyectiva. Sin esta condición no podemos definir una categoría de signaturas ocultas con estos morfismos como demuestra el Ejemplo 2.4.3.

Diversos autores [45, 27, 47] han introducido generalizaciones de este morfismo de signaturas oculto. Por ejemplo en [45] se define una signatura oculta en la que se destacan una serie de operaciones que se llaman *operaciones de comportamiento* ("behavioral operations"). La idea es utilizar únicamente estas operaciones destacadas para construir los contextos con los que realizar las observaciones. De este modo únicamente es necesario que los morfismos de signaturas ocultos verifiquen la condición de encapsulamiento sobre las operaciones de comportamiento. Además, en estos artículos se generalizan otros conceptos de la teoría oculta. Por ejemplo, las signaturas ocultas pueden contener operaciones con más de un género oculto en su argumento. Con estas definiciones obtienen nuevas instituciones ocultas. No obstante, pierden propiedades interesantes como es la obtención del álgebra final y por lo tanto la conexión con la teoría coalgebraica.

En esta sección trataremos de redefinir la institución $\mathcal{H}^D$ para que incluya nuestro caso particular, siendo conscientes de que no alcanzamos la generalidad de las instituciones obtenidas en los artículos que acabamos de mencionar. No obstante, no perdemos ninguna propiedad respecto de $\mathcal{H}^D$ y utilizamos un lenguaje bastante sencillo e intuitivo en el que continuamos usando los conceptos habituales de la especificación oculta.

Una de las características fundamentales de la signatura oculta Σ_{imp} que construimos a partir de una signatura Σ, es que todas sus operaciones son deconstructoras. Esta signatura contiene por lo tanto un tipo especial de géneros ocultos que llamaremos *géneros deconstructores*, que son aquéllos que no están presentes como género resultado de ninguna operación.

Definición 3.2.3 (Género deconstructor). Un género oculto h de una signatura oculta $H\Sigma$ se dice *deconstructor*, si h no aparece como género resultado de ninguna operación en $H\Sigma$.

La siguiente definición es una generalización de la definición de morfismo de signaturas oculto.

Definición 3.2.4 (Morfismo de signaturas deconstructor débil). Sean $H\Sigma = (S, \Omega)$ y $H\Sigma' = (S', \Omega')$ dos signaturas ocultas sobre la misma signatura visible $V\Sigma = (VS, V\Omega)$ y dominio de datos D. Un *morfismo de signaturas deconstructor débil* $\mu = (\mu_S, \mu_\Omega)\colon H\Sigma \to H\Sigma'$ es un morfismo de signaturas tal que:

1. $\mu_S(v) = v$, para cada $v \in VS$ y $\mu_\Omega(\phi) = \phi$, para cada $\phi \in V\Omega$,
2. $\mu_S(HS) \subseteq HS'$,
3. si $s_1 \in HS$ no es un género deconstructor, $s_2 \in HS$ y $\mu_S(s_1) = \mu_S(s_2)$, entonces $s_1 = s_2$,
4. para cada operación $\omega'\colon s'_1 \dots s'_n \to s' \in \Omega'$, $n \geq 1$, con $s'_1 = \mu_S(s_1)$, $s_1 \in HS$, si s_1 no es un género deconstructor de $H\Sigma$, entonces existe alguna operación ω en $H\Sigma$ tal que $\omega' = \mu_\Omega(\omega)$.

Un morfismo de signaturas deconstructor débil es un morfismo de signaturas ocultas en el que únicamente exigimos la condición de encapsulamiento en las operaciones cuyo primer argumento sea la imagen de algún género no deconstructor. Además, únicamente es necesario exigir inyectividad cuando se involucra a los géneros no deconstructores.

Comentamos de nuevo que este morfismo es un caso particular del morfismo de signaturas presente en [45], si consideramos como operaciones de comportamiento todas las operaciones de la signatura excepto aquéllas que sean deconstructoras sobre un género deconstructor. No obstante, en este trabajo no se tiene en cuenta la condición de inyectividad necesaria si pretendemos definir una categoría de signaturas ocultas con este tipo de morfismos.

Si $H\Sigma$ es una signatura *deconstructora*, es decir, una signatura oculta en la que todas sus operaciones son deconstructoras, entonces todos sus géneros ocultos son deconstructores y por lo tanto podemos obviar las condiciones 3 y 4 en la definición anterior. Obtenemos en este caso un morfismo vertical de signaturas ocultas.

Observamos que este morfismo de signaturas permite relacionar una signatura deconstructora con una no deconstructora. En un apartado posterior veremos como esta posibilidad nos impide reproducir el morfismo de instituciones entre la nueva institución oculta que pretendemos definir (la cual va a incorporar este tipo de morfismos de signaturas) y la institución algebraica ecuacional. Por ello, definimos a continuación un nuevo tipo de morfismo de signaturas ocultas que no permite que se establezcan estas relaciones entre signaturas.

Definición 3.2.5 (Morfismo de signaturas deconstructor fuerte). Un *morfismo de signaturas deconstructor fuerte* entre dos signaturas ocultas $H\Sigma$ y $H\Sigma'$ es un morfismo de signaturas deconstructor débil $\mu = (\mu_S, \mu_\Omega)\colon H\Sigma \to H\Sigma'$ en el que para cada género oculto $s \in H\Sigma$, si s es un género deconstructor de $H\Sigma$, entonces $\mu_S(s)$ es un género deconstructor de $H\Sigma'$.

Definimos ahora dos categorías de signaturas ocultas en las que utilizamos estas nuevas definiciones de morfismo de signatura oculto.

Definición 3.2.6 (Categoría de signaturas ocultas con morfismos deconstructores débiles (respectivamente, fuertes)). Definimos la *categoría de signaturas ocultas con morfismos deconstructores débiles sobre un universo de datos D*, que denotamos por $SIG_{\widetilde{\mathcal{H}}^D}$, como la categoría cuyos objetos son las signaturas ocultas sobre una signatura visible y álgebra de datos en D y cuyos morfismos son los morfismos de signaturas deconstructores débiles.

Análogamente, definimos la *categoría de signaturas ocultas con morfismos deconstructores fuertes sobre un universo de datos D*, que denotamos por $SIG_{\bar{\mathcal{H}}^D}$, cuyos morfismos son los morfismos de signaturas deconstructores fuertes.

Con el siguiente lema demostramos que las categorías anteriores están bien definidas.

Lema 3.2.7. *$SIG_{\widetilde{\mathcal{H}}^D}$ y $SIG_{\bar{\mathcal{H}}^D}$ son categorías.*

Demostración. La única propiedad que no se obtiene de forma inmediata a partir de la categoría $SIG_{\mathcal{H}^D}$ es que la composición de dos morfismos de signaturas débiles (respectivamente, fuertes) es un morfismo de signaturas deconstructor débil (respectivamente, fuerte).

Es claro que la imagen de un género no deconstructor es a su vez no deconstructor. Por ello, la composición de aplicaciones inyectivas sobre dos géneros en los que uno de ellos no es deconstructor es inyectiva sobre estos géneros.

Dados dos morfismos de signaturas deconstructores débiles $\mu\colon H\Sigma \to H\Sigma'$ y $\mu'\colon H\Sigma' \to H\Sigma''$ y dada una operación $\omega''\colon s''_1 \dots s''_n \to s'' \in H\Omega''$, $n \geq 1$, con $s''_1 = \mu' \circ \mu_S(s_1), s_1 \in HS$, si suponemos que existe una operación $\sigma\colon r_1 \dots r_m \to s_1 \in H\Omega$, entonces debemos probar que existe una operación ω en $H\Sigma$ tal que $\omega'' = \mu' \circ \mu_\Omega(\omega)$.

Tenemos por un lado que $\omega''\colon s''_1 \dots s''_n \to s'' \in H\Omega''$, con $s''_1 = \mu'_S(\mu_S(s_1))$, $\mu_S(s_1) \in HS'$, y por otro que $\sigma\colon r_1 \dots r_m \to s_1 \in H\Omega$, es decir $\mu_\Omega(\sigma)\colon \mu_S(r_1) \dots \mu_S(r_m) \to \mu_S(s_1) \in H\Omega'$. Entonces, como μ' es un morfismo de signaturas deconstructor débil, existe una operación $\omega'\colon s'_1 \dots s'_n \to s'$ en $H\Sigma'$ tal que $\omega'' = \mu'_\Omega(\omega')$, con $\mu'_S(s'_1) = s''_1$. Además, $\mu'_S(\mu_S(s_1)) = s''_1$ y $\mu_S(s_1)$ no es deconstructor, entonces $s'_1 = \mu_S(s_1)$, $s_1 \in HS$. Ahora, como μ es morfismo de signaturas deconstructor débil y $\sigma \in H\Omega$, entonces existe una operación $\omega \in H\Omega$ tal que $\omega' = \mu_\Omega(\omega)$, es decir, $\omega'' = \mu'_\Omega(\mu_\Omega(\omega))$.

Además, la composición de dos morfismos que conservan géneros deconstructores conserva géneros deconstructores. Así, la composición de dos morfismos deconstructores fuertes es un morfismo deconstructor fuerte. ■

Podemos definir dos nuevas instituciones ocultas que incluyan como categoría de signaturas las categorías que acabamos de definir. Para ello tenemos que el Teorema 2.4.13 también es cierto para morfismos de signaturas deconstructores débiles (o fuertes), siempre y cuando no se aplique a ecuaciones que sean igualdades entre variables ocultas. Para evitar esa posibilidad dichas igualdades serán excluidas del conjunto de sentencias. Observamos no obstante que no perdemos generalidad en las especificaciones que podemos construir. Las ecuaciones de la forma $(\{z\}, z, z)$, identidad de una variable oculta, no aportan significación a los modelos. Una ecuación del tipo $(\{z, z'\}, z, z')$, de variables de género oculto, obliga a que la observación a través de los contextos de cualquier elemento en ese género produzca el mismo resultado, de modo que se pierde la capacidad para diferenciar elementos a partir de observaciones.

En el siguiente teorema recogemos esta propiedad de satisfacibilidad para morfismos de signaturas deconstructores débiles. Ofrecemos además su demostración, que es una ligera modificación de la demostración realizada por Goguen y Diaconescu en [41] con morfismos de signaturas ocultos, para poner de relieve el papel que juega la condición de encapsulamiento en la misma.

La idea fundamental en este desarrollo es que los géneros deconstructores son *exclusivamente* de observación. Como no existen operaciones con estos géneros como resultado, tampoco existen términos distintos de las variables en dicho género. Por lo tanto, las únicas ecuaciones que podemos construir de género deconstructor son igualdades entre variables, que como hemos comentado no interesan en nuestro estudio. Entonces, la única manera de poder obtener información de un soporte para un género deconstructor es a través de la observación "directa" de una operación deconstructora sobre dicho conjunto. De este modo, podemos dotar de estructura a los conjuntos correspondientes a estos géneros en las álgebras ocultas si exigimos, vía ecuaciones visibles, que estas observaciones nos ofrezcan resultados esperados. Por ejemplo, podemos retomar la especificación de una familia de grupos que utilizamos en el capítulo anterior. En esta especificación exigimos que cada grupo de la familia, que esperamos que sea un elemento del soporte asignado al género (deconstructor) destacado, verifique la propiedad asociativa sobre los elementos del dominio de datos asignado al género visible.

Teorema 3.2.8. *Sean $H\Sigma$ y $H\Sigma'$ dos signaturas ocultas definidas sobre la misma signatura y álgebra visibles $V\Sigma$ y D, y sea $\mu\colon H\Sigma \to H\Sigma'$ un morfismo de signaturas deconstructor débil entre ellas. Entonces,*

$$\mu(A') \models\!\!\!\equiv_{H\Sigma} e \text{ si y sólo si } A' \models\!\!\!\equiv_{H\Sigma'} \mu(e)$$

para toda $H\Sigma'$-álgebra oculta A' y toda $H\Sigma$-ecuación e que no sea una igualdad entre variables ocultas.

Demostración. En el caso en el que la ecuación e sea visible se tiene que la satisfacción de comportamiento es equivalente a la satisfacción ecuacional ordinaria, por lo que se obtiene el resultado a partir de ésta.

Sean $\mu\colon H\Sigma \to H\Sigma'$ un morfismo de signaturas deconstructor débil, A' una $H\Sigma'$-álgebra oculta y $e = (X, t, u)$ una $H\Sigma$-ecuación oculta que no sea igualdad entre variables. Debemos probar que la siguiente equivalencia se cumple, donde el lado izquierdo está cuantificado por los contextos c apropiados para t y u, y el lado derecho por los contextos c' apropiados para $\mu(t)$ y $\mu(u)$:

$$\mu(A') \models_{H\Sigma} (X, c(t), c(u)) \text{ si y sólo si } A' \models_{H\Sigma'} (\mu(X), c'(\mu(t)), c'(\mu(u))).$$

Nos fijamos primero en la implicación inversa. Esta implicación es exactamente la misma que aparece en [41], ya que no interviene en su proceso la condición de encapsulamiento.

Sea c un contexto apropiado para t; si suponemos que t es un $H\Sigma$-término de género s_1, entonces c es un $H\Sigma$-término visible que contiene una única ocurrencia de una variable z de género s_1. Entonces $\mu(c)$ es un $H\Sigma'$-término visible con una única ocurrencia de la variable z de género $\mu(s_1)$. Por lo tanto, $\mu(c)$ es un contexto apropiado para $\mu(t)$ y $\mu(u)$ y por hipótesis se verifica que

$$A' \models_{H\Sigma'} (\mu(X), \mu(c)(\mu(t)), \mu(c)(\mu(u))).$$

Ahora bien, por extensión del morfismo de signaturas a términos se tiene que $\mu(c(t)) = \mu(c)(\mu(t))$ y si aplicamos la relación de satisfacibilidad de la institución algebraica ecuacional (el morfismo de signaturas deconstructor es, en particular, un morfismo de signaturas ordinario; lo mismo sucede con la $H\Sigma'$-álgebra oculta A' y la $H\Sigma$-ecuación e) se tiene

$$\mu(A') \models_{H\Sigma} (X, c(t), c(u)).$$

Demostraremos a continuación la implicación directa.

Sea $e = (X, t, u)$ una $H\Sigma$-ecuación de género oculto s_1, que no es una igualdad entre variables. Entonces en la signatura $H\Sigma$ debe existir una operación oculta $\sigma\colon r_1 \dots r_m \to s_1$, $m \geq 0$, que actúe como última operación en al menos uno de los dos términos de la ecuación e. Sea c' un contexto adecuado para $\mu(t)$, es decir, c' es un $H\Sigma'$-término visible con una única ocurrencia de una variable z' de género $\mu(s_1)$. Entonces c' debe contener entre sus subtérminos a un término $\omega'(z', t'_2, \dots, t'_n)$ con $\omega'\colon s'_1 s'_2 \dots s'_n \to s'$, $n \geq 1$, donde $s'_1 = \mu(s_1)$ y s'_i es visible, $\forall i = 2, \dots, n$. Por definición de morfismo de signaturas deconstructor débil, existe una operación $\omega\colon s_1 s_2 \dots s_n \to s \in H\Omega$ tal que $\omega' = \mu(\omega)$ (es en este punto donde necesitamos la condición de encapsulamiento sin la cual no podemos garantizar la existencia de esta operación). Es importante observar que para nuestro caso particular es suficiente que esta condición se cumpla para operaciones sobre géneros no deconstructores. El resto de la demostración corresponde nuevamente con la demostración de [41].

Suponemos en un primer momento que $c' = \omega'(z', t'_2, \dots, t'_n)$. Debemos probar si

$$A' \models_{H\Sigma'} (\mu(X), \omega'(\mu(t), t'_2, \dots, t'_n), \omega'(\mu(u), t'_2, \dots, t'_n))$$

es decir, si para toda valoración $q\colon \mu(X) \to A'$ se verifica

$$\overline{q}(\omega'(\mu(t), t'_2, \dots, t'_n)) = \overline{q}(\omega'(\mu(u), t'_2, \dots, t'_n)).$$

Sea $q\colon \mu(X) \to A'$ una valoración. Por definición de la interpretación de un término por la valoración tenemos que $\overline{q}(\omega'(\mu(t), t'_2, \dots, t'_n)) = \omega'_{A'}(\overline{q}(\mu(t)), \overline{q}(t'_2), \dots, \overline{q}(t'_n))) = \omega'_{A'}(\overline{q}(\mu(t)), d_{t'_2}, \dots, d_{t'_n}))$, donde hemos denotado $\overline{q}(t'_i) = d_{t'_i}$, con $d_{t'_i}$ una constante visible, ya que t'_i es un término visible, $\forall i = 2, \dots, n$, (en el caso $n = 1$, no es necesaria esta notación ya que no aparece ningún género visible). Por ello es suficiente con probar si

$$\omega'_{A'}(\overline{q}(\mu(t)), d_{t'_2}, \dots, d_{t'_n}) = \omega'_{A'}(\overline{q}(\mu(u)), d_{t'_2}, \dots, d_{t'_n}).$$

Consideramos ahora el contexto $\omega(z, d_{t'_2}, \dots, d_{t'_n})$ tal que $\omega' = \mu(\omega)$. Este contexto es adecuado para t y u. Entonces por hipótesis se verifica que

$$\mu(A') \models_{H\Sigma} (X, \omega(t, d_{t'_2}, \dots, d_{t'_n}), \omega(u, d_{t'_2}, \dots, d_{t'_n})).$$

Si aplicamos la relación de satisfacibilidad de la institución algebraica ecuacional obtenemos que

$$A' \models_{H\Sigma'} (\mu(X), \omega'(\mu(t), d_{t'_2}, \dots, d_{t'_n}), \omega'(\mu(u), d_{t'_2}, \dots, d_{t'_n})),$$

y como $q\colon \mu(X) \to A'$ es una valoración, tenemos que

$$\overline{q}(\omega'(\mu(t), d_{t'_2}, \dots, d_{t'_n})) = \overline{q}(\omega'(\mu(u), d_{t'_2}, \dots, d_{t'_n})).$$

Obtenemos directamente la igualdad buscada a partir de esta última, ya que

$$\overline{q}(\omega'(\mu(t), d_{t'_2}, \ldots, d_{t'_n})) = \omega'_{A'}(\overline{q}(\mu(t)), d_{t'_2}, \ldots, d_{t'_n}).$$

Consideramos en segundo lugar un contexto general $c' = \omega''(\omega'(z', t'_2, \ldots, t'_n))$, que contiene como subtérmino a $\omega'(z', t'_2, \ldots, t'_n)$. Además éste es el único subtérmino de c' donde aparece una variable. Debemos probar que

$$A' \models_{H\Sigma'} (\mu(X), \omega''(\omega'(\mu(t), t'_2, \ldots, t'_n)), \omega''(\omega'(\mu(u), t'_2, \ldots, t'_n)))$$

es decir, si para toda valoración $q\colon \mu(X) \to A'$ se verifica que

$$\overline{q}(\omega''(\omega'(\mu(t), t'_2, \ldots, t'_n))) = \overline{q}(\omega''(\omega'(\mu(u), t'_2, \ldots, t'_n))).$$

Sea $q\colon \mu(X) \to A'$ una valoración. Como la única variable de c' es z' entonces es suficiente con probar si

$$\omega''_{A'}(\overline{q}(\omega'(\mu(t), t'_2, \ldots, t'_n))) = \omega''_{A'}(\overline{q}(\omega'(\mu(u), t'_2, \ldots, t'_n))).$$

Pero, esta igualdad es cierta ya que como paso previo se ha probado la igualdad

$$\overline{q}(\omega'(\mu(t), t'_2, \ldots, t'_n)) = \overline{q}(\omega'(\mu(u), t'_2, \ldots, t'_n)).$$

■

Podemos definir entonces la siguiente institución oculta.

Definición 3.2.9 (Institución ecuacional oculta con morfismos deconstructores débiles). La *institución ecuacional oculta con morfismos deconstructores débiles definida sobre un universo de datos D*, que denotamos por $\widetilde{\mathcal{H}}^D = (SIG_{\widetilde{\mathcal{H}}^D}, Sen_{\widetilde{\mathcal{H}}^D}, Mod_{\widetilde{\mathcal{H}}^D}, \models^{\widetilde{\mathcal{H}}^D})$ es la institución formada por:

1. la categoría $SIG_{\widetilde{\mathcal{H}}^D}$ de las signaturas ocultas con morfismos deconstructores débiles sobre un universo de datos D,

2. el funtor $Sen_{\widetilde{\mathcal{H}}^D}\colon SIG_{\widetilde{\mathcal{H}}^D} \to Set$, asocia a cada signatura oculta $H\Sigma$ de $SIG_{\widetilde{\mathcal{H}}^D}$, el conjunto de ecuaciones $Sen_{\mathcal{H}^D}(H\Sigma)$ que no sean identidades entre variables de género oculto, y a cada morfismo de signaturas μ de $SIG_{\widetilde{\mathcal{H}}^D}$, la aplicación $Sen_{\mathcal{H}^D}(\mu)$,

3. el funtor $Mod_{\widetilde{\mathcal{H}}^D}\colon SIG_{\widetilde{\mathcal{H}}^D} \to Cat^{op}$ se define de la misma forma que el funtor $Mod_{\mathcal{H}^D}$ sobre los objetos y morfismos de $SIG_{\widetilde{\mathcal{H}}^D}$,

4. y la relación de satisfacibilidad $\models^{\widetilde{\mathcal{H}}^D}$ es la satisfacción de comportamiento $\models^{\mathcal{H}^D}$.

Además, notamos que el teorema anterior también es cierto para morfismos de signaturas deconstructores fuertes (ya que también son débiles). Entonces, de modo análogo podemos definir una *institución ecuacional oculta con morfismos deconstructores fuertes definida sobre un universo de datos D*, que denotaremos por $\bar{\mathcal{H}}^D$.

3.2.5 Un morfismo de instituciones unidireccional entre las instituciones $\bar{\mathcal{E}}^D$ y $\widetilde{\mathcal{H}}^D$

La institución algebraica ecuacional generalizada definida sobre un universo de datos D, que denotamos por $\bar{\mathcal{E}}^D$, supone una ampliación de la institución $\mathcal{E}^D$ que utilizamos en el capítulo anterior para representar el sustrato de datos presente en el sistema que se pretende modelar, EAT en nuestro caso. La diferencia fundamental entre estas dos instituciones es que la generalizada permite trabajar con morfismos entre signaturas distintos de la identidad. En esta sección definiremos un morfismo de instituciones entre esta nueva institución y una institución oculta que modele de nuevo la construcción *imp* de modo que representamos el sustrato de datos de un modo más conveniente. Recordamos que nos veíamos obligados a limitar, en nuestra institución algebraica ecuacional, los morfismos de signaturas a identidades por la condición de encapsulamiento presente en los morfismos de signaturas ocultos. En la sección anterior hemos definido dos instituciones ocultas generalizadas que prescinden en parte de dicha condición. Ambas instituciones nos servirán como marco de llegada para nuestra construcción.

Dadas la institución algebraica ecuacional generalizada definida sobre un universo de datos D, $\bar{\mathcal{E}}^D = (SIG_{\bar{\mathcal{E}}^D}, Sen_{\bar{\mathcal{E}}^D}, Mod_{\bar{\mathcal{E}}^D}, \models^{\bar{\mathcal{E}}^D})$, y la institución ecuacional oculta con morfismos deconstructores débiles definida sobre un universo de datos D, $\widetilde{\mathcal{H}}^D = (SIG_{\widetilde{\mathcal{H}}^D}, Sen_{\widetilde{\mathcal{H}}^D}, Mod_{\widetilde{\mathcal{H}}^D}, \models^{\widetilde{\mathcal{H}}^D})$, definimos a continuación un morfismo de instituciones unidireccional entre ambas.

Comenzamos definiendo un funtor $\Phi\colon SIG_{\bar{\mathcal{E}}^D} \to SIG_{\widetilde{\mathcal{H}}^D}$ entre las respectivas categorías de signaturas. Dada una signatura $\Sigma = (S, \Omega)$ de $SIG_{\bar{\mathcal{E}}^D}$, definimos $\Phi(\Sigma)$ como la signatura oculta $\Sigma_{imp} = (S_{imp}, \Omega_{imp} \cup C)$. Esta signatura oculta es la misma signatura que obteníamos como imagen del funtor $\Phi^{\mathcal{E}^D-\mathcal{H}^D}$ sobre la signatura $(S, \Omega \cup C)$. Observamos que podemos incorporar las constantes de C en cada género a Σ_{imp} sin necesidad de que estén presentes en Σ.

Para completar la definición del funtor Φ debemos establecer la imagen de los morfismos de $SIG_{\bar{\mathcal{E}}^D}$ que ahora no son triviales. Dado un morfismo de signaturas de $SIG_{\bar{\mathcal{E}}^D}$, $\mu\colon \Sigma \to \Sigma'$ entre las signaturas $\Sigma = (S, \Omega)$ y $\Sigma' = (S, \Omega')$, definimos $\Phi(\mu)$ como el morfismo de signaturas deconstructor débil $\mu_{imp}\colon \Sigma_{imp} \to \Sigma'_{imp}$ entre las signaturas $\Sigma_{imp} = (S_{imp}, \Omega_{imp} \cup C)$ y $\Sigma'_{imp} = (S'_{imp}, \Omega'_{imp} \cup C)$. Este morfismo está definido por la función sobre los géneros $\mu_{impS}(s) := \mu_S(s)$ si $s \in S$ y $\mu_{impS}(imp_{\Sigma}) := imp_{\Sigma'}$, y la función sobre las operaciones $\mu_{imp\Omega}(imp\_\omega) := imp\_\omega'$, con $\omega' = \mu_\Omega(\omega)$, para cada operación $imp\_\omega \in \Omega_{imp}$, y $\mu_{imp\Omega}(d) = d$ para cada constante $d \in C$. Observamos que realmente obtenemos un morfismo de signaturas deconstructor débil (es más, también es fuerte), ya que μ_{imp} es la identidad sobre la parte visible (tenemos que μ_S es la identidad por pertenecer μ a $SIG_{\bar{\mathcal{E}}^D}$), es biyectiva sobre géneros ocultos (sólo hay uno en cada signatura y lo conserva) y todas las operaciones ocultas de ambas signaturas son deconstructoras.

Es claro que se tiene el siguiente resultado.

Lema 3.2.10. *La aplicación Φ es un funtor.*

La segunda componente del morfismo de instituciones unidireccional será una transformación natural $\alpha\colon Sen_{\bar{\mathcal{E}}^D} \Rightarrow Sen_{\widetilde{\mathcal{H}}^D} \circ \Phi$, que estará formada por una familia de aplicaciones $\alpha_\Sigma\colon Sen_{\bar{\mathcal{E}}^D}(\Sigma) \to Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\Sigma))$, una por cada $\Sigma \in SIG_{\bar{\mathcal{E}}^D}$, verificando la condición de naturalidad respecto de los morfismos de signaturas.

Dada $\Sigma = (S, \Omega) \in SIG_{\bar{\mathcal{E}}^D}$ consideramos como α_Σ la misma aplicación $\alpha^{\mathcal{E}^D - \mathcal{H}^D}_{(S, \Omega \cup C)}$ correspondiente a la trasformación natural entre sentencias del morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$, sobre las sentencias de $Sen_{\bar{\mathcal{E}}^D}(\Sigma)$. Observamos que la aplicación anterior está bien definida, ya que la imagen de una ecuación de $Sen_{\bar{\mathcal{E}}^D}(\Sigma)$ nunca es una igualdad de variables ocultas.

Para comprobar que la familia de aplicaciones anterior forma una transformación natural debemos comprobar que verifica la propiedad de naturalidad. En este caso tenemos morfismos de signaturas distintos de la identidad, por lo que esta propiedad no es trivial.

Lema 3.2.11. *La familia de aplicaciones $\alpha\colon Sen_{\bar{\mathcal{E}}^D} \Rightarrow Sen_{\widetilde{\mathcal{H}}^D} \circ \Phi$ constituye una transformación natural.*

Demostración. Debemos comprobar si para cada morfismo $\mu\colon \Sigma \to \Sigma'$ de $SIG_{\bar{\mathcal{E}}^D}$, entre las signaturas $\Sigma = (S, \Omega)$ y $\Sigma' = (S', \Omega')$, el siguiente diagrama es conmutativo:

$$\begin{tikzcd}
Sen_{\bar{\mathcal{E}}^D}(\Sigma) \arrow[r, "\alpha_\Sigma"] \arrow[d, "Sen_{\bar{\mathcal{E}}^D}(\mu)"'] & Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\Sigma)) \arrow[d, "Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\mu))"] \\
Sen_{\bar{\mathcal{E}}^D}(\Sigma') \arrow[r, "\alpha_{\Sigma'}"'] & Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\Sigma'))
\end{tikzcd}$$

es decir, debemos comprobar si para cualquier sentencia $e = (X, t, u)$ de $Sen_{\bar{\mathcal{E}}^D}(\Sigma)$, con $t, u \in T_{\Sigma(X), s}$, $s \in S$, se verifica que

$$Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\mu)) \circ \alpha_\Sigma(e) = \alpha_{\Sigma'} \circ Sen_{\bar{\mathcal{E}}^D}(\mu)(e).$$

Por un lado, tenemos que por definición de α_Σ, $Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\mu))(\alpha_\Sigma((X, t, u)))$ es igual a $Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\mu))((X \cup \{z_{imp}\}, \theta(t), \theta(u))$. Esta sentencia es igual a $(\Phi(\mu)(X \cup \{z_{imp}\}),$ $\Phi(\mu)(\theta(t)),\ \Phi(\mu)(\theta(u)))$ por definición de $Sen_{\widetilde{\mathcal{H}}^D}(\Phi(\mu))$.

Por otro lado, tenemos que por definición de $Sen_{\bar{\mathcal{E}}^D}(\mu)$, $\alpha_{\Sigma'}(Sen_{\bar{\mathcal{E}}^D}(\mu)((X, t, u)))$ es igual a $\alpha_{\Sigma'}((\mu(X), \mu(t), \mu(u)))$. Esta sentencia es igual a $(\mu(X) \cup \{z_{imp}\}, \theta(\mu(t)), \theta(\mu(u)))$ por definición de $\alpha_{\Sigma'}$.

Basta entonces con probar si para cualquier término $t \in T_{\Sigma(X)}$

$$\Phi(\mu)(\theta(t)) = \theta(\mu(t)).$$

Demostramos la igualdad anterior por inducción sobre la estructura de los $\Sigma(X)$-términos:

- si $t = x \in X_s$, $s \in S$, entonces $\Phi(\mu)(\theta(x)) = \mu_{imp}(x) = \mu(x) = \theta(\mu(x))$;
- y si $t = \omega(t_1, \ldots, t_n)$, con $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 0$ y $t_i \in T_{\Sigma(X),s_i}$, $i = 1, \ldots, n$, entonces

$$\begin{aligned}
&\Phi(\mu)(\theta(\omega(t_1, \ldots, t_n))) = \\
&= \mu_{imp}(imp_\omega(z_{imp}, \theta(t_1), \ldots, \theta(t_n))) \\
&= \mu_{imp}(imp_\omega)(\mu_{imp}(z_{imp}), \mu_{imp}(\theta(t_1)), \ldots, \mu_{imp}(\theta(t_n))) \\
&= imp_\mu(\omega)(z_{imp}, \mu_{imp}(\theta(t_1)), \ldots, \mu_{imp}(\theta(t_n))) \\
&= imp_\mu(\omega)(z_{imp}, \theta(\mu(t_1)), \ldots, \theta(\mu(t_n))) \text{ por hipótesis de inducción,} \\
&= \theta(\mu(\omega)(\mu(t_1), \ldots, \mu(t_n))) \\
&= \theta(\mu(\omega(t_1, \ldots, t_n))).
\end{aligned}$$

En cuanto a las variables tenemos que $\Phi(\mu)(X \cup \{z_{imp}\}) = \mu_{imp}(X) \cup \mu_{imp}(z_{imp}) = \mu(X) \cup z_{imp}$. Recordamos que la familia de aplicaciones α, añade al conjunto de variables una *nueva* variable que no es usada en ninguno de los géneros del universo U de géneros. ■

Por último, vamos a definir una transformación natural $\beta\colon Mod_{\bar{\mathcal{E}}^D} \Rightarrow Mod_{\widetilde{\mathcal{H}}^D} \circ \Phi$. Esta transformación natural vendrá dada por una familia de funtores $\beta_\Sigma\colon Mod_{\bar{\mathcal{E}}^D}(\Sigma) \to Mod_{\widetilde{\mathcal{H}}^D}(\Phi(\Sigma))$, un funtor β_Σ para cada signatura Σ de $SIG_{\bar{\mathcal{E}}^D}$. Definimos nuevamente el funtor β_Σ de la misma forma que el funtor $\beta^{\mathcal{E}^D-\mathcal{H}^D}_{(S,\Omega\cup C)}$ correspondiente a la trasformación natural entre modelos del morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$ (recordar que se obtiene un álgebra oculta a partir de un álgebra sin más que añadir un conjunto soporte con un único elemento para el género oculto).

Otra vez no es trivial comprobar que la familia de funtores anterior forma una transformación natural ya que tenemos morfismos distintos de la identidad.

Lema 3.2.12. *La familia de aplicaciones $\beta\colon Mod_{\bar{\mathcal{E}}^D} \Rightarrow Mod_{\widetilde{\mathcal{H}}^D} \circ \Phi$ constituye una transformación natural.*

Demostración. Debemos comprobar si para cada morfismo $\mu\colon \Sigma \to \Sigma'$ de $SIG_{\bar{\mathcal{E}}^D}$, entre las signaturas $\Sigma = (S, \Omega)$ y $\Sigma' = (S', \Omega')$, el siguiente diagrama es conmutativo:

$$\begin{array}{ccc}
Mod_{\bar{\mathcal{E}}^D}(\Sigma) & \xrightarrow{\beta_\Sigma} & Mod_{\widetilde{\mathcal{H}}^D}(\Phi(\Sigma)) \\
{\scriptstyle Mod_{\bar{\mathcal{E}}^D}(\mu)}\uparrow & & \uparrow{\scriptstyle Mod_{\widetilde{\mathcal{H}}^D}(\Phi(\mu))} \\
Mod_{\bar{\mathcal{E}}^D}(\Sigma') & \xrightarrow[\beta_{\Sigma'}]{} & Mod_{\widetilde{\mathcal{H}}^D}(\Phi(\Sigma'))
\end{array}$$

El diagrama de funtores anterior conmuta trivialmente sobre los Σ'-homomorfismos de $Mod_{\bar{\mathcal{E}}^D}(\Sigma')$ ya que estos son identidades. Falta entonces por comprobar que el diagrama conmuta sobre las Σ'-álgebras, es decir, debemos comprobar si para cualquier Σ'-álgebra A' de $Mod_{\bar{\mathcal{E}}^D}(\Sigma')$

$$Mod_{\widetilde{\mathcal{H}}^D}(\Phi(\mu)) \circ \beta_{\Sigma'}(A') = \beta_\Sigma \circ Mod_{\bar{\mathcal{E}}^D}(\mu)(A').$$

Es claro que las dos Σ_{imp}-álgebra ocultas anteriores tienen los mismos soportes (para los géneros visibles están fijados y para el único género oculto ambas tienen como soporte el conjunto unipuntual $\{*\}$).

Para cada operación $imp\_\omega\colon imp_{\Sigma} s_1 \dots s_n \to s \in \Omega_{imp}$, $n \geq 0$, las interpretaciones de imp_ω en las Σ_{imp}-álgebras anteriores son las funciones:

$$\begin{aligned} & imp\_\omega_{Mod_{\widetilde{\mathcal{H}}^D}(\Phi(\mu))(\beta_{\Sigma'}(A'))}(*, a_1, \dots, a_n) = \\ = \;& \mu_{imp}(imp\_\omega)_{\beta_{\Sigma'}(A')}(*, a_1, \dots, a_n) \\ = \;& imp\_\mu(\omega)_{\beta_{\Sigma'}(A')}(*, a_1, \dots, a_n) \\ = \;& \mu(\omega)_{A'}(a_1, \dots, a_n) \\ = \;& \omega_{Mod_{\bar{\mathcal{E}}^D}(\mu)(A')}(a_1, \dots, a_n) \\ = \;& imp\_\omega_{\beta_{\Sigma}(Mod_{\bar{\mathcal{E}}^D}(\mu)(A'))}(*, a_1, \dots, a_n), \end{aligned}$$

y para cada constante $d \in C$, $d_{Mod_{\widetilde{\mathcal{H}}^D}(\Phi(\mu))(\beta_{\Sigma'}(A'))} = d = d_{\beta_{\Sigma}(Mod_{\bar{\mathcal{E}}^D}(\mu)(A'))}$. ■

Obtenemos directamente que las tres componentes que acabamos de describir verifican la propiedad de satisfacibilidad exigida a un morfismo de instituciones unidireccional. La prueba se basa en que dicha propiedad es cierta para las componentes del morfismo de instituciones unidireccional entre $\mathcal{E}^D$ y $\mathcal{H}^D$.

Teorema 3.2.13. *La terna (Φ, α, β) que hemos definido anteriormente constituye un morfismo de instituciones unidireccional entre $\bar{\mathcal{E}}^D$ y $\widetilde{\mathcal{H}}^D$.*

Observamos que podemos utilizar en las definiciones anteriores la institución ecuacional oculta con morfismos deconstructores fuertes. Así, de modo análogo se obtiene un morfismo de instituciones unidireccional entre $\bar{\bar{\mathcal{E}}}^D$ y $\bar{\mathcal{H}}^D$.

3.2.6 Un morfismo de instituciones unidireccional entre las instituciones $\bar{\mathcal{H}}^D$ y $\mathcal{E}$

En esta sección definimos un morfismo de instituciones unidireccional entre $\bar{\mathcal{H}}^D$ y $\mathcal{E}$, similar al morfismo entre $\mathcal{H}^D$ y $\mathcal{E}$ que hemos considerado en el capítulo anterior. Bastará con probar que las construcciones del morfismo anterior pueden extenderse a morfismos de signaturas deconstructores fuertes. Sin embargo, observaremos que no podemos obtener un morfismo similar para $\widetilde{\mathcal{H}}^D$, ya que sus morfismos de signaturas no respetan los géneros deconstructores.

Dadas la institución oculta con morfismos deconstructores fuertes sobre un universo de datos D, $\bar{\mathcal{H}}^D = (SIG_{\bar{\mathcal{H}}^D}, Sen_{\bar{\mathcal{H}}^D}, Mod_{\bar{\mathcal{H}}^D}, \models^{\bar{\mathcal{H}}^D})$ y la institución algebraica ecuacional, $\mathcal{E} = (SIG_{\mathcal{E}}, Sen_{\mathcal{E}}, Mod_{\mathcal{E}}, \models^{\mathcal{E}})$, definimos a continuación un morfismo de instituciones unidireccional entre ambas.

A nivel de signaturas consideramos como $\Phi^{\bar{\mathcal{H}}^D - \mathcal{E}}$ la correspondiente componente del morfismo de instituciones unidireccional entre $\mathcal{H}^D$ y $\mathcal{E}$. Es decir, tomamos el funtor olvido $i\colon SIG_{\bar{\mathcal{H}}^D} \to SIG_{\mathcal{E}}$ entre las respectivas categorías de signaturas.

A nivel de sentencias consideramos la transformación natural $\alpha\colon Sen_{\bar{\mathcal{H}}^D} \Rightarrow Sen_{\mathcal{E}} \circ i$, que viene dada por la inclusión $\alpha_{H\Sigma}\colon Sen_{\bar{\mathcal{H}}^D}(H\Sigma) \hookrightarrow Sen_{\mathcal{E}}(i(H\Sigma))$, para cada signatura oculta $H\Sigma$ de $SIG_{\bar{\mathcal{H}}^D}$. Estas aplicaciones no se definen como identidades ya que recordamos que excluimos las ecuaciones entre variables ocultas en $Sen_{\bar{\mathcal{H}}^D}(H\Sigma)$.

Por último, a nivel de modelos, definimos una transformación natural $\beta\colon Mod_{\bar{\mathcal{H}}^D} \Rightarrow Mod_{\mathcal{E}} \circ i$ dada por un funtor $\beta_{H\Sigma}\colon Mod_{\bar{\mathcal{H}}^D}(H\Sigma) \to Mod_{\mathcal{E}}(i(H\Sigma))$, por cada signatura oculta $H\Sigma$ de $SIG_{\bar{\mathcal{H}}^D}$. Para definir estos funtores necesitamos la noción de *equivalencia deconstructora*, una variante de la equivalencia de comportamiento que definimos a continuación.

La equivalencia deconstructora se define como la igualdad en los géneros que sean deconstructores y como la equivalencia de comportamiento en el resto. En realidad, esta equivalencia es un caso particular de la equivalencia de comportamiento para las operaciones de comportamiento presentada en [45] o en [43], si consideramos como operaciones de comportamiento todas las operaciones de la signatura excepto aquellas que sean deconstructoras sobre un género deconstructor.

Definición 3.2.14 (Equivalencia deconstructora). Sean $H\Sigma = (S, \Omega)$ una signatura oculta definida sobre $V\Sigma$ y D, y A una $H\Sigma$-álgebra oculta. Decimos que dos elementos $a, a' \in A_s$, con $s \in S$, son *equivalentes deconstructores* si y sólo si $a = a'$ si s es un género deconstructor de $H\Sigma$ y $a \equiv^A_{H\Sigma,s} a'$ en otro caso. Denotaremos que $a, a' \in A_s$ son equivalentes deconstructores por $a \equiv^{A,d}_{H\Sigma,s} a'$. En ocasiones lo denotaremos por $a \equiv^d_{H\Sigma,s} a'$, o incluso $a \equiv^d_{H\Sigma} a'$, si no hay lugar a confusión.

Es claro que la familia de relaciones de equivalencia $\equiv^{A,d}_{H\Sigma,s}$ define una relación de congruencia en un $H\Sigma$-álgebra oculta A, que denotamos $\equiv^{A,d}_{H\Sigma}$ o $\equiv^d_{H\Sigma}$. Además, dado un $H\Sigma$-homomorfismo oculto $h\colon A \to B$, tenemos trivialmente que h respeta los cocientes originados en A y B por $\equiv^d_{H\Sigma}$, es decir que para cada $a, a' \in A_s$, con $s \in S$, si $a \equiv^d_{H\Sigma} a'$ en A entonces $h_s(a) \equiv^d_{H\Sigma} h_s(a')$ en B, ya que si s es deconstructor entonces tenemos igualdades y si s no es deconstructor entonces podemos aplicar la doble implicación de la equivalencia de comportamiento. (Observamos que no obtenemos para la equivalencia deconstructora la doble implicación, ya que podemos definir $H\Sigma$-homomorfismos ocultos no inyectivos en un género no deconstructor. No obstante, únicamente es necesaria esta implicación en la Proposición 1.4.13 para poder definir el homomorfismo cociente entre las respectivas álgebras cocientes.)

Ahora, podemos definir el funtor $\beta_{H\Sigma}$ como la aplicación sobre objetos que asocia a cada $H\Sigma$-álgebra oculta A, la $H\Sigma$-álgebra $A/_{\equiv^d_{H\Sigma}}$. El $H\Sigma$-homomorfismo $h/_{\equiv^d_{H\Sigma}}$ es la imagen de h por $\beta_{H\Sigma}$.

Para morfismos de signaturas deconstructores fuertes obtenemos el siguiente resultado que se corresponde con la Proposición 2.4.19 para morfismos de signaturas ocultos. A partir esta proposición se obtiene que la familia de funtores que hemos definido forma una transformación natural de forma similar a como demostramos que la correspondiente familia de funtores para el morfismo entre $\mathcal{H}^D$ y $\mathcal{E}$ definen una transformación natural.

Proposición 3.2.15. *Sean $H\Sigma = (S, \Omega)$ y $H\Sigma' = (S', \Omega')$ dos signaturas ocultas definidas sobre $V\Sigma$ y D, $\mu\colon H\Sigma \to H\Sigma'$ un morfismo de signaturas deconstructor fuerte y A' una $H\Sigma'$-álgebra oculta. Entonces,*

$$a \equiv^{Mod_{\bar{\mathcal{H}}^D}(\mu)(A'),d}_{H\Sigma',\mu(s)} b \text{ si y sólo si } a \equiv^{A',d}_{H\Sigma,s} b$$

para cada $s \in S$ y $a, b \in A'_{\mu(s)}$.

Demostración. Como μ un morfismo de signaturas deconstructor fuerte, entonces $s \in S$ es deconstructor si y sólo si $\mu(s)$ es deconstructor. Ahora, si s es deconstructor, tenemos la igualdad en ambos miembros, y si s no es deconstructor aplicamos la Proposición 2.4.19. ■

Debemos observar que no es cierto un resultado parecido para morfismos de signaturas deconstructores débiles. Si para un género deconstructor s, permitimos que $\mu(s)$ sea no deconstructor, entonces $a \equiv^d_{H\Sigma',\mu(s)} b$ no implica en general $a = b$. Por ello no podemos definir de una forma similar la trasformación natural entre los funtores modelos correspondiente a un morfismo de instituciones unidireccional entre $\widetilde{\mathcal{H}}^D$ y $\mathcal{E}$.

Para completar la definición del morfismo de instituciones unidireccional debemos comprobar que la terna que acabamos de considerar (Φ, α, β) verifica la condición de satisfacibilidad. Como no existen ecuaciones distintas de las igualdades entre variables para los géneros deconstructores, obtenemos el siguiente lema que marca la relación que existe entre las nociones de equivalencia deconstructora y satisfacción de comportamiento. La demostración de la condición de satisfacibilidad que buscamos es inmediata a partir del mismo.

Lema 3.2.16. *Sean $H\Sigma = (S, \Omega)$ una signatura oculta definida sobre $V\Sigma$ y D, A una $H\Sigma$-álgebra oculta y (X, t, u) una $H\Sigma$-ecuación que no sea una igualdad de variables ocultas. Entonces,*

$$A \models\!\!\!\equiv_{H\Sigma} (X, t, u) \text{ si y sólo si } \overline{p}(t) \equiv^d_{H\Sigma} \overline{p}(u), \text{ para toda valoración } p : X \to A.$$

Teorema 3.2.17. *Sea $H\Sigma$ una signatura oculta de $\bar{\mathcal{H}}^D$. Entonces,*

$$A \models^{\bar{\mathcal{H}}^D}_{H\Sigma} e \text{ si y sólo si } \beta_{H\Sigma}(A) \models^{\mathcal{E}}_{\Phi(H\Sigma)} \alpha_{H\Sigma}(e)$$

para cada $A \in Obj(Mod_{\bar{\mathcal{H}}^D}(H\Sigma))$ y $e \in Sen_{\bar{\mathcal{H}}^D}(H\Sigma)$.

3.2.7 Primer diagrama alternativo

Nuestro objetivo es construir un diagrama alternativo al presentado en el capítulo anterior, en el que aparezcan las instituciones que acabamos de definir. Observamos que únicamente vamos a poder completar el diagrama si utilizamos como institución oculta la institución que tiene como morfismos de signaturas los morfismos deconstructores fuertes $\bar{\mathcal{H}}^D$, ya que es la institución para la que hemos definido un morfismo de instituciones unidireccional entre ella y la institución algebraica ecuacional $\mathcal{E}$.

En las secciones anteriores, hemos estudiado con detalle el morfismo de instituciones unidireccional entre $\bar{\mathcal{E}}^D$ y $\bar{\mathcal{H}}^D$, que nuevamente se considera el morfismo central del diagrama. En esta sección comentaremos brevemente las definiciones del resto de los morfismos del diagrama.

La definición de un morfismo de instituciones unidireccional entre las instituciones $\bar{\mathcal{E}}^D$ y $\mathcal{E}$ podemos realizarla de forma similar a como hemos construido el morfismo entre $\bar{\mathcal{E}}^D$ y $\bar{\mathcal{H}}^D$, pero sin tener en cuenta la distinción entre parte visible y parte oculta.

El morfismo de instituciones unidireccional entre $\bar{\mathcal{E}}^D$ y $CoAlg(\bar{\mathcal{E}}^D)$ podemos obtenerlo a partir de la definición general de un morfismo entre una institución y su institución coalgebraica asociada, ya que $\bar{\mathcal{E}}^D$ verifica las condiciones necesarias para ello. Recordamos que estas condiciones son que para cada signatura Σ de $SIG_{\bar{\mathcal{E}}^D}$, se verifique que $Mod_{\bar{\mathcal{E}}^D}(\Sigma)$ sea una categoría pequeña, y que cada uno de sus morfismos sean endomorfismos. Ambas son ciertas, ya que tenemos prefijado el dominio de datos D por lo que la categoría anterior es pequeña, y además sus morfismos son identidades.

Para completar nuestro diagrama, únicamente falta por definir un morfismo de instituciones unidireccional entre $CoAlg(\bar{\mathcal{E}}^D)$ y $\bar{\mathcal{H}}^D$. Este morfismo se construye a partir del morfismo entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$ de forma similar a como hemos definido el morfismo entre $\bar{\mathcal{E}}^D$ y $\bar{\mathcal{H}}^D$ a partir del morfismo entre $\mathcal{E}^D$ y $\mathcal{H}^D$.

Obtenemos el siguiente diagrama de morfismos de instituciones unidireccionales conmutativo:

$$\begin{array}{ccc}
 & & CoAlg(\bar{\mathcal{E}}^D) \\
 & \nearrow & \downarrow \\
\bar{\mathcal{E}}^D & \longrightarrow & \bar{\mathcal{H}}^D \\
 & \searrow & \downarrow \\
 & & \mathcal{E}
\end{array}$$

Podemos comprobar que el diagrama conmuta sobre los nuevos morfismos de signaturas que se han introducido en la institución $\bar{\mathcal{E}}^D$, sin más que aplicar la definición de los morfismos de instituciones sobre ellos. La conmutatividad del resto del diagrama se obtiene de forma similar al diagrama construido en el capítulo anterior.

3.3 Estudio de los morfismos

En esta sección nos centramos en el estudio del tipo de morfismo de instituciones que hemos utilizado para construir nuestro diagrama. El objetivo del mismo es explicar por qué se han elegido morfismos de instituciones unidireccionales y si es posible construir un diagrama similar utilizando la noción habitual de morfismo de instituciones o cualquier otra de las relaciones entre instituciones que han sido definidas por distintos autores.

Cuando se revisa la literatura resulta en principio sorprendente la gran variedad de tipos de morfismos diferentes que se han utilizado para abordar la noción de relación entre instituciones. Incluso se han publicado varios artículos [46, 95] que pretenden recopilar y poner un poco de orden entre esta diversidad de morfismos. No obstante, si nos paramos a pensar más detenidamente en la noción de institución, deducimos que esto no es tan extraordinario. Bajo el concepto de institución se engloban todas las herramientas que son necesarias para hacer especificación. No debe resultar por ello extraño que las posibilidades para relacionar dos marcos de especificación puedan ser variadas.

En nuestro recorrido para modelar la operación *imp* hemos estudiado diferentes posibilidades para situar una signatura Σ en un marco de especificación y la correspondiente signatura Σ_{imp} en otro marco de especificación distinto. De esta forma entendemos que, vía nuestra operación, podemos establecer una relación entre dichos marcos. Es decir, se construye un morfismo entre instituciones. Surge entonces el problema de decidir qué tipo de morfismo de instituciones es el más adecuado en nuestro caso. Si nos fijamos en la operación *imp* observamos que, además de relacionar signaturas, tenemos que a partir de sentencias para una signatura Σ, se construyen sentencias para la signatura Σ_{imp} y lo mismo respecto de los modelos. Es por ello lógico pensar en un morfismo de instituciones en el que las relaciones entre signaturas, sentencias y modelos vayan en la *misma dirección*. Este morfismo de instituciones es el morfismo de instituciones unidireccional. En el capítulo anterior hemos utilizado este tipo de morfismos para modelar la operación *imp*. Sin embargo, también podemos pensar en utilizar la noción de morfismo de instituciones habitual, en el que el *paso* entre signaturas y modelos se realiza en la misma dirección y el de sentencias se realiza en dirección contraria. Estableceríamos de esta forma, a través de nuestra operación, una relación entre dos marcos de especificación vía este otro tipo de morfismo de instituciones. En este caso, la representación anterior de la operación *imp* sobre sentencias se pospone al trabajo con especificaciones.

A continuación, trataremos de obtener un segundo diagrama alternativo utilizando morfismos de instituciones. Al ir el paso entre sentencias en sentido contrario nos veremos obligados a prescindir de las igualdades entre variables ocultas en la institución oculta. Curiosamente esta condición también era necesaria, aunque por otros motivos, en la construcción del primer diagrama alternativo. No obstante, no nos va a ser posible completar el diagrama. El punto clave para ello es que no se verifica la condición de satisfacibilidad del morfismo que relaciona la teoría coalgebraica y la teoría de especificaciones ocultas. Como veremos, dicha condición sólo es cierta para ecuaciones que presenten una única variable oculta. Estas ecuaciones aparecen en [22] dentro de las *especificaciones coalgebraicas*, a través de las cuales se establece una relación entre estas dos teorías. En general, los trabajos que consideran ecuaciones para coálgebras (ver, por ejemplo, [51, 78]) exigen esta condición. (En el diagrama de instituciones que incluimos en [28] pasamos por alto este detalle al dar por hecho que únicamente trabajábamos con este tipo de ecuaciones.) Definiremos entonces un nuevo tipo de morfismo de instituciones, al restringir el conjunto de sentencias sobre el que se verifica la condición de satisfacibilidad. Con este nueva idea trataremos de completar el diagrama.

Comenzamos, como hicimos en el primer diagrama alternativo, definiendo el morfismo de instituciones entre $\mathcal{E}^D$ y $\mathcal{H}^D$, que nuevamente consideramos el morfismo central del diagrama.

3.3.1 Un morfismo de instituciones entre la institución $\mathcal{E}^D$ y la institución $\mathcal{H}^D$

En esta sección vamos a establecer un morfismo de instituciones entre la institución algebraica ecuacional definida sobre un universo de datos D, $\mathcal{E}^D = (SIG_{\mathcal{E}^D}, Sen_{\mathcal{E}^D}, Mod_{\mathcal{E}^D}, \models^{\mathcal{E}^D})$, y la institución ecuacional oculta sobre un universo de datos D, $\mathcal{H}^D = (SIG_{\mathcal{H}^D}, Sen_{\mathcal{H}^D}, Mod_{\mathcal{H}^D}, \models^{\mathcal{H}^D})$.

Observamos que la diferencia entre un morfismo de instituciones y un morfismo de instituciones unidireccional radica en el sentido de la transformación natural que relaciona las sentencias de ambas instituciones, junto con el consiguiente cambio en la relación de satisfacibilidad. Entonces, para definir este morfismo de instituciones, que denotamos nuevamente por (Φ, α, β), utilizaremos las componentes que son comunes con el morfismo de instituciones unidireccional que hemos definido en el capítulo anterior entre estas mismas instituciones. Es decir, el funtor Φ entre las categorías de signaturas y la transformación natural β entre los funtores modelos son los mismos que las respectivas componentes del morfismo de instituciones unidireccional definido anteriormente entre ambas instituciones. Por ello, únicamente falta por definir la componente que relaciona los funtores sentencias de las mismas.

Definimos a continuación una transformación natural $\alpha\colon Sen_{\mathcal{H}^D} \circ \Phi \Rightarrow Sen_{\mathcal{E}^D}$. Esta transformación natural consistirá en una aplicación $\alpha_\Sigma\colon Sen_{\mathcal{H}^D}(\Phi(\Sigma)) \to Sen_{\mathcal{E}^D}(\Sigma)$ por cada signatura Σ de $SIG_{\mathcal{E}^D}$, de modo que se verifique la condición de naturalidad.

Dada una signatura Σ de $SIG_{\mathcal{E}^D}$ definimos la aplicación $\alpha_\Sigma\colon Sen_{\mathcal{H}^D}(\Sigma_{imp}) \to Sen_{\mathcal{E}^D}(\Sigma)$ como la relación que asocia a cada Σ_{imp}-sentencia e (que no sea igualdad entre variables ocultas) la Σ-sentencia que se obtiene si eliminamos en e todos los términos de género imp_Σ y sustituimos todas las operaciones imp_ω de Σ_{imp} en e por la correspondiente operación ω de Σ.

Utilizamos el siguiente ejemplo para ilustrar esta aplicación.

Ejemplo 3.3.1. Dada la categoría $SIG_{\mathcal{E}^D}$, elegimos un género g del universo de géneros (para este género tendremos fijado un conjunto de datos D_g). Nos fijamos nuevamente en la signatura `GRP` que hemos venido utilizando para especificar un grupo sobre este conjunto de datos y la correspondiente signatura $\mathtt{GRP}_{imp}$ que obtenemos al aplicarle la operación *imp*.

Dadas las variables de género g: x, x_1, x_2, x_3, y las variables de género $imp_{\mathtt{GRP}}$: z_1, z_2, z_3, z_4, algunas ecuaciones para la signatura $\mathtt{GRP}_{imp}$ son:

$$imp\_prd(z_1, imp\_prd(z_2, x1, x2), x3) = imp\_prd(z_3, x1, imp\_prd(z_4, x2, x3))$$
$$imp\_prd(z_1, imp\_unt(z_2), x) = x$$
$$imp\_prd(z_1, imp\_inv(z_2, x), x) = imp\_unt(z_3)$$

entonces, las ecuaciones correspondientes por la aplicación $\alpha_{\mathtt{GRP}}$ para la signatura `GRP` son:

$$prd(prd(x1, x2), x3) = prd(x1, prd(x2, x3))$$
$$prd(unt, x) = x \qquad \square$$
$$prd(inv(x), x) = unt$$

Podemos definir la aplicación anterior de un modo más preciso a través de un proceso de inducción sobre la forma de los términos que constituyen la ecuación. Dada una signatura $\Sigma = (S, \Omega \cup C)$ de $SIG_{\mathcal{E}^D}$, la aplicación $\alpha_\Sigma\colon Sen_{\mathcal{H}^D}(\Sigma_{imp}) \to Sen_{\mathcal{E}^D}(\Sigma)$ asocia a cada Σ_{imp}-ecuación $e = (X, t, u)$ de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$, la Σ-ecuación $\alpha_\Sigma(e) := (X \setminus X_{imp_\Sigma}, \phi(t), \phi(u))$, donde $\phi = (\phi_s\colon T_{\Sigma_{imp}(X),s} \to T_{\Sigma(X \setminus X_{imp_\Sigma}),s})_{s \in S}$ es la familia de aplicaciones entre términos que damos a continuación. Dado $s \in S$, definimos la aplicación ϕ_s por inducción sobre la estructura de un término $t \in T_{\Sigma_{imp}(X),s}$ del siguiente modo:

- si $t = x \in X_s$ variable de género s, $\phi_s(x) = x$;
- si $t = d$, con $d\colon \to s$ una constante de C, entonces $\phi_s(d) = d$;
- y si $t = imp\_\omega(z, t_1, \ldots, t_n)$ con $imp\_\omega\colon imp_\Sigma s_1 \ldots s_n \to s \in \Omega_{imp}$, $n \geq 0$, z variable de X_{imp_Σ} y $t_i \in T_{\Sigma_{imp}(X),s_i}$, $1 \leq i \leq n$, entonces $\phi_s(imp\_\omega(z, t_1, \ldots, t_n)) = \omega(\phi_{s_1}(t_1), \ldots, \phi_{s_n}(t_n))$.

Como los únicos morfismos de $SIG_{\mathcal{E}^D}$ son las identidades, obtenemos de modo trivial que α verifica la condición de naturalidad.

Es importante observar que la familia de aplicaciones ϕ no contiene una aplicación que actúe sobre términos de género distinguido imp_Σ. Esta aplicación no es necesaria ya que recordamos que no existen términos distintos de las variables en este género y por lo tanto tampoco ecuaciones distintas de las igualdades entre variables, y hemos excluido estas sentencias de $\mathcal{H}^D$.

En esta sección vamos a excluir, tanto en $\mathcal{H}^D$ como en $\mathcal{E}^D$, todas las sentencias que sean igualdades entre variables de cualquier género. Para obtener el diagrama también las excluiremos de $\mathcal{E}$. Obtenemos de esta forma tres instituciones distintas que continuaremos llamando y denotando de la misma manera, ya que no perdemos capacidad de expresión respecto de las especificaciones que podemos construir. Las ecuaciones de la forma $(\{z\}, z, z)$, identidad de una variable, no aportan significación a los modelos en ninguna de las tres instituciones. Una ecuación del tipo $(\{z, z'\}, z, z')$, de variables de género s de $\mathcal{H}^D$ visible o de $\mathcal{E}^D$ y $\mathcal{E}$, se puede eliminar construyendo una especificación equivalente en la que los términos de género s sean identificados con una única constante.

Esta transformación natural entre sentencias completa nuestro morfismo de instituciones que recogemos en el siguiente teorema.

Teorema 3.3.2. *Sean $\mathcal{E}^D$ y $\mathcal{H}^D$, respectivamente, la institución algebraica ecuacional y la institución oculta definidas sobre D, de modo que se excluyen en ambas las ecuaciones que identifican variables. Entonces, la terna (Φ, α, β) que acabamos de definir es un morfismo de instituciones entre $\mathcal{E}^D$ y $\mathcal{H}^D$.*

Demostración. Falta por comprobar que se verifica la condición de satisfacibilidad. Es decir, que para cada signatura Σ de $SIG_{\mathcal{E}^D}$,

$$A \models^{\mathcal{E}^D}_{\Sigma} \alpha_\Sigma(e) \text{ si y sólo si } \beta_\Sigma(A) \models^{\mathcal{H}^D}_{\Phi(\Sigma)} e$$

para cualquier álgebra $A \in Obj(Mod_{\mathcal{E}^D}(\Sigma))$ y cualquier sentencia $e \in Sen_{\mathcal{H}^D}(\Phi(\Sigma))$.

Nos centramos primero en la demostración de la implicación directa.

Sean $\Sigma = (S, \Omega \cup C)$ una signatura de $SIG_{\mathcal{E}^D}$, A un álgebra de $Mod_{\mathcal{E}^D}(\Sigma)$ y $e = (X, t, u)$, con $t, u \in T_{\Sigma_{imp}(X),s}$, $s \in S$, una sentencia de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$. Suponemos que se cumple $A \models^{\mathcal{E}^D}_{\Sigma} \alpha_\Sigma((X, t, u))$, es decir, que para toda valoración $p\colon X \setminus X_{imp_\Sigma} \to A$ se verifica que

$$\overline{p}(\phi(t)) = \overline{p}(\phi(u)).$$

Tenemos que demostrar que $\beta_\Sigma(A) \models^{\mathcal{H}^D}_{\Phi(\Sigma)} (X, t, u)$. Como $e = (X, t, u)$ es una ecuación visible, entonces únicamente debemos comprobar que para cada valoración $q\colon X \to \beta_\Sigma(A)$, se verifica que

$$\overline{q}(t) = \overline{q}(u).$$

Sea $q\colon X \to \beta_\Sigma(A)$ una valoración. Podemos definir la valoración $q_{|S}\colon X \setminus X_{imp_\Sigma} \to A$ sin más que restringir la valoración q a los géneros de S. Si aplicamos la hipótesis con esta valoración obtenemos que

$$\overline{q_{|S}}(\phi(t)) = \overline{q_{|S}}(\phi(u)).$$

Observamos entonces que es suficiente probar que

$$\overline{q_{|S}}(\phi(t)) = \overline{q}(t)$$

para cualquier término $t \in T_{\Sigma_{imp}(X),s}$, $s \in S$.

Demostramos a continuación la igualdad anterior por inducción sobre la estructura de los Σ_{imp}-términos t que puedan formar parte de una ecuación de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$. Recordamos que estos términos no pueden ser de género imp_Σ y por lo tanto únicamente contienen variables como subtérminos de este género.

- si $t = x \in X_s$, entonces $\overline{q_{|S}}(\phi(x)) = \overline{q_{|S}}(x) = q_{|S}(x) = q(x) = \overline{q}(x)$;
- si $t = d$, con $d\colon \to s$ una constante de C, entonces $\overline{q_{|S}}(\phi(d)) = \overline{q_{|S}}(d) = d = \overline{q}(d)$;

- y si $t = imp\_\omega(z, t_1, \ldots, t_n)$, con $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 0$, z variable arbitraria de género imp_Σ y $t_i \in T_{\Sigma_{imp}(X),s_i}$, $1 \leq i \leq n$, entonces:

$$\begin{aligned}
&\overline{q_{|S}}(\phi(imp_\omega(z, t_1, \ldots, t_n))) = \\
&= \overline{q_{|S}}(\omega(\phi(t_1), \ldots, \phi(t_n))) \\
&= \omega_A(\overline{q_{|S}}(\phi(t_1)), \ldots, \overline{q_{|S}}(\phi(t_n))) \\
&= \omega_A(\overline{q}(\phi(t_1)), \ldots, \overline{q}(\phi(t_n))) \text{ por hipótesis de inducción,} \\
&= imp_\omega_{\beta_\Sigma(A)}(*, \overline{q}(\phi(t_1)), \ldots, \overline{q}(\phi(t_n))) \text{ por definición de } \beta_\Sigma(A), \\
&= imp_\omega_{\beta_\Sigma(A)}(\overline{q}(z), \overline{q}(\phi(t_1)), \ldots, \overline{q}(\phi(t_n))) \\
&= \overline{q}(imp_\omega(z, t_1, \ldots, t_n)).
\end{aligned}$$

La demostración de la implicación inversa es totalmente análoga por inducción sobre los términos, tras la observación de que cada valoración $X \setminus X_{imp_\Sigma} \to A$ puede ser trivialmente extendida a una valoración $q\colon X \to \beta_\Sigma(A)$, asignando a cada variable de X_{imp_Σ} el único valor posible del álgebra $\beta_\Sigma(A)$ en dicho género. ■

3.3.2 Los otros morfismos de instituciones del diagrama

Una vez que hemos definido el morfismo de instituciones central del diagrama, en esta sección trataremos de estudiar el resto de los morfismos que lo integran.

El morfismo de instituciones entre $\mathcal{E}^D$ y $\mathcal{E}$ se define de forma similar al morfismo de la sección anterior entre $\mathcal{E}^D$ y $\mathcal{H}^D$, pero sin realizar la distinción entre géneros ocultos y géneros visibles.

Los morfismos de instituciones entre $\mathcal{H}^D$ y $\mathcal{E}$, y entre $\mathcal{E}^D$ y $CoAlg(\mathcal{E}^D)$ se obtienen directamente a partir de sus correspondientes morfismos de instituciones unidireccionales en el diagrama inicial. Basta tener en cuenta que las transformaciones naturales entre sentencias eran en los dos casos aplicaciones identidad, por lo que no hay problema en tomar la dirección opuesta. Para ello recordar que hemos restringido las sentencias de las instituciones $\mathcal{H}^D$ y $\mathcal{E}$, a aquellas sentencias que no sean igualdades entre variables. Entonces, podemos redefinir de forma inmediata el morfismo de instituciones unidireccional entre $\mathcal{H}^D$ y $\mathcal{E}$, de modo que la transformación natural entre las sentencias continúe siendo la identidad.

El morfismo de instituciones entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$ requiere de un estudio más detallado, que realizamos en la siguiente sección.

3.3.3 Estudio de un morfismo de instituciones entre la institución $CoAlg(\mathcal{E}^D)$ y la institución $\mathcal{H}^D$

En esta sección trataremos de definir un morfismo de instituciones entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$. Para ello, podemos nuevamente aprovechar las componentes que son comunes con el morfismo de instituciones unidireccional que hemos definido entre ellas en el capítulo anterior. Es decir, definimos el funtor entre signaturas Φ y la transformación natural β entre modelos de la misma forma que las correspondientes componentes del morfismo de instituciones unidireccional entre dichas instituciones. Entonces, únicamente falta por determinar la componente que relaciona las sentencias de las mismas.

Una transformación natural $\alpha\colon Sen_{\mathcal{H}^D} \circ \Phi \Rightarrow Sen_{CoAlg(\mathcal{E}^D)}$ constará de una aplicación $\alpha_\Sigma\colon Sen_{\mathcal{H}^D}(\Phi(\Sigma)) \to Sen_{CoAlg(\mathcal{E}^D)}(\Sigma)$ por cada signatura Σ de $SIG_{CoAlg(\mathcal{E}^D)}$. Recordamos que $Sen_{\mathcal{H}^D}(\Phi(\Sigma)) = Sen_{\mathcal{H}^D}(\Phi^{\mathcal{E}^D-\mathcal{H}^D}(\Sigma))$ y que $Sen_{CoAlg(\mathcal{E}^D)}(\Sigma) = Sen_{\mathcal{E}^D}(\Sigma)$. Entonces, hacemos un primer intento de definir un morfismo de instituciones considerando $\alpha_\Sigma := \alpha_\Sigma^{\mathcal{E}^D-\mathcal{H}^D}$ para cada signatura Σ de $SIG_{CoAlg(\mathcal{E}^D)}$, donde $\alpha^{\mathcal{E}^D-\mathcal{H}^D}$ es la transformación natural que forma parte del morfismo de instituciones (sin apellidos) entre $\mathcal{E}^D$ y $\mathcal{H}^D$.

Para obtener un morfismo de instituciones falta por comprobar que se verifica la condición de satisfacibilidad. No obstante, como probamos a continuación únicamente podemos demostrar una de las dos implicaciones de esta condición.

Teorema 3.3.3. *Sea Σ una signatura de $SIG_{CoAlg(\mathcal{E}^D)}$. Entonces,*

$$\beta_\Sigma((\mathcal{X},\gamma)) \models^{\mathcal{H}^D}_{\Phi(\Sigma)} e \quad\Longrightarrow\quad (\mathcal{X},\gamma) \models^{CoAlg(\mathcal{E}^D)}_{\Sigma} \alpha_\Sigma(e)$$

para cualquier coálgebra $(\mathcal{X},\gamma)$ de $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$ y cualquier sentencia e de $Sen_{\mathcal{H}^D}(\Phi(\Sigma))$.

Demostración. Sean $\Sigma = (S, \Omega \cup C)$ una signatura de $SIG_{CoAlg(\mathcal{E}^D)}$, $(\mathcal{X},\gamma)$ una coálgebra de $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$ y $e = (X, t, u)$, con $t, u \in T_{\Sigma_{imp}(X),s}$, $s \in S$, una ecuación de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$ arbitrarias.

Por un lado, tenemos que

$$(\mathcal{X},\gamma) \models^{CoAlg(\mathcal{E}^D)}_{\Sigma} \alpha_\Sigma((X,t,u)) \quad \text{si y sólo si} \quad \gamma(a) \models^{\mathcal{E}^D}_{\Sigma} (X \setminus X_{imp_\Sigma}, \phi(t), \phi(u)),\ \forall a \in \mathcal{X},$$

sin más que aplicar las definiciones de $\models^{CoAlg(\mathcal{E}^D)}_{\Sigma}$ y de α_Σ.

Por otro lado, tenemos que

$$\beta_\Sigma((\mathcal{X},\gamma)) \models^{\mathcal{H}^D}_{\Sigma_{imp}} (X,t,u) \quad \text{si y sólo si} \quad \beta_\Sigma((\mathcal{X},\gamma)) \models^{\mathcal{E}^D}_{\Sigma_{imp}} (X,t,u),$$

ya que (X, t, u) es una ecuación visible.

Entonces es suficiente con demostrar que:

$$\beta_\Sigma((\mathcal{X},\gamma)) \models^{\mathcal{E}^D}_{\Sigma_{imp}} (X,t,u) \implies \gamma(a) \models^{\mathcal{E}^D}_{\Sigma} (X \setminus X_{imp_\Sigma}, \phi(t), \phi(u)),\ \forall a \in \mathcal{X}.$$

Fijado $a \in \mathcal{X}$, debemos demostrar que para cada valoración $p\colon X \setminus X_{imp_\Sigma} \to \gamma(a)$ se verifica que $\overline{p}(\phi(t)) = \overline{p}(\phi(u))$. Dada p, consideramos la valoración por $\beta_\Sigma((\mathcal{X},\gamma))$,

$q\colon X \to \beta_\Sigma((\mathcal{X},\gamma))$, dada por $q(z) = a$ para todo $z \in X_{imp_\Sigma}$ y $q(x) = p(x)$ para cada $x \in X \setminus X_{imp_\Sigma}$. Entonces por hipótesis se verifica que $\overline{q}(t) = \overline{q}(u)$.

Si aplicamos ahora inducción sobre la estructura de los términos, obtenemos nuevamente que, para cualquier término t que pueda formar parte de una ecuación de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$, se verifica que $\overline{p}(\phi(t)) = \overline{q}(t)$. Esta igualdad prueba la implicación.

Este proceso de inducción es similar al que hemos realizado en el morfismo de instituciones entre $\mathcal{E}^D$ y $\mathcal{H}^D$, de modo que ahora nos quedamos con un único elemento a de $\beta_\Sigma((\mathcal{X},\gamma))$ en el género destacado imp_Σ, al cual se le asocia la Σ-álgebra $\gamma(a)$. Detallamos a continuación dicho proceso de inducción sobre la estructura de un término $t \in T_{\Sigma_{imp}(X),s}$, $s \in S$:

- si $t = x \in X_s$, entonces $\overline{p}(\phi(x)) = \overline{p}(x) = p(x) = q(x) = \overline{q}(x)$;
- si $t = d$, con $d\colon \to s$ constante de C, entonces $\overline{p}(\phi(d)) = \overline{p}(d) = d = \overline{q}(d)$;
- y si $t = imp\_\omega(z, t_1, \ldots, t_n)$, con $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 0$, z variable arbitraria de género imp_Σ y $t_i \in T_{\Sigma_{imp}(X),s_i}$, $i = 1, \ldots, n$, entonces

$$\begin{aligned}
&\overline{p}(\phi(imp_\omega(z, t_1, \ldots, t_n))) = \\
&\quad = \overline{p}(\omega(\phi(t_1), \ldots, \phi(t_n))) \\
&\quad = \omega_{\gamma(a)}(\overline{p}(\phi(t_1)), \ldots, \overline{p}(\phi(t_n))) \\
&\quad = \omega_{\gamma(a)}(\overline{q}(t_1), \ldots, \overline{q}(t_n)) \text{ por hipótesis de inducción,} \\
&\quad \text{ ya que la valoración } q \text{ está definida de la misma forma para todo } z \in X_{imp_\Sigma}, \\
&\quad = imp_\omega_{\beta_\Sigma((\mathcal{X},\gamma))}(a, \overline{q}(t_1), \ldots, \overline{q}(t_n)) \text{ por definición de } \beta_\Sigma((\mathcal{X},\gamma)), \\
&\quad = imp_\omega_{\beta_\Sigma((\mathcal{X},\gamma))}(\overline{q}(z), \overline{q}(t_1), \ldots, \overline{q}(t_n)) \\
&\quad = \overline{q}(imp_\omega(z, t_1, \ldots, t_n)).
\end{aligned}$$

■

La implicación en el otro sentido no es cierta para toda sentencia de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$, contando que en el universo de datos D exista un conjunto con más de un elemento. Si intentamos aplicar un proceso de inducción similar sobre la estructura de un término, tenemos que a partir de una valoración q por $\beta_\Sigma((\mathcal{X},\gamma))$, la valoración que necesitamos por una Σ-álgebra depende del valor que adopte q en las variables ocultas de la ecuación. Si esta ecuación presenta más de una variable oculta no tenemos ninguna valoración que nos sirva para dicho proceso de inducción.

Ejemplo 3.3.4. Sea *nat* un género del universo de géneros que tiene asociado un conjunto con más de un elemento. Por ejemplo, suponemos que $D_{nat} = \mathbb{N}$.

Consideramos una signatura Σ de $SIG_{\mathcal{E}^D}$ que tiene como único género a *nat* y como operaciones una única constante $const\colon \to nat$, además del conjunto de constantes que se derivan de D, $\{i\colon \to nat\}_{i\in\mathbb{N}}$.

El endofuntor constante $F_\Sigma\colon Set \to Set$ asociado a Σ está definido por $F_\Sigma(X) = Obj(Mod_{\mathcal{E}^D}(\Sigma))$, para cada $X \in Set$. Definimos la F_Σ-coálgebra $(\{1,2\}, \gamma\colon \{1,2\} \to F_\Sigma(\{1,2\}))$ dada por $\gamma(1) = A_1$ y $\gamma(2) = A_2$, donde A_n, para $n = 1, 2$, es la Σ-álgebra que se define por $const_{A_n} = n$, e $i_{A_n} = i$, $i \in \mathbb{N}$.

Tenemos que $\Phi(\Sigma) = \Sigma_{imp}$. Esta signatura es una signatura oculta que tiene un género visible nat, un género oculto imp_Σ, un conjunto de constantes visibles $\{i\colon \to nat\}_{i\in\mathbb{N}}$, y una operación oculta $imp\_const\colon imp_\Sigma \to nat$. Entonces, $\beta_\Sigma((\{1,2\},\gamma))$ es la Σ_{imp}-álgebra A que se define por los conjuntos: $A_{imp_\Sigma} = \{1,2\}$ y $A_{nat} = \mathbb{N}$, y las operaciones: $i_A = i$, para $i \in \mathbb{N}$ e $imp\_const_A\colon \{1,2\} \to \mathbb{N}$, tal que $imp\_const_A(1) = const_{A_1} = 1$ e $imp\_const_A(2) = const_{A_2} = 2$.

Elegimos la Σ_{imp}-sentencia $(\{x,y\}, imp\_const(x), imp\_const(y))$ que presenta dos variables ocultas. Entonces se verifica que

$$(\{1,2\},\gamma) \models_\Sigma^{CoAlg(\mathcal{E}^D)} \alpha_\Sigma((\{x,y\}, imp\_const(x), imp\_const(y)),$$

ya que trivialmente $\gamma(a) \models_\Sigma^{\mathcal{E}^D} const = const$, $\forall a \in \{1,2\}$.

Sin embargo, no se verifica que

$$A \models_{\Sigma_{imp}}^{\mathcal{H}^D} (\{x,y\}, imp\_const(x), imp\_const(y)),$$

es decir, si para cualquier valoración $q\colon \{x,y\} \to A$, $\overline{q}(imp\_const(x)) = \overline{q}(imp\_const(y))$. Ya que si elegimos la valoración dada por $q_{imp_\Sigma}\colon \{x,y\} \to A_{imp_\Sigma}$ tal que $q_{imp_\Sigma}(x) = 1$ y $q_{imp_\Sigma}(y) = 2$, entonces tenemos que $\overline{q}(imp\_const(x)) = imp\_const_A(1) = 1$ que es distinto de $\overline{q}(imp\_const(y)) = imp\_const_A(2) = 2$. □

No obstante, están perfectamente delimitadas las sentencias de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$ para las cuales se verifica la implicación en el otro sentido. Se trata de aquellas ecuaciones en las que aparece una única variable de género oculto. Estas ecuaciones son también elegidas por Cîrstea en [22] para establecer una relación entre la teoría coalgebraica y la teoría oculta a través de las *especificaciones coalgebraicas*. No obstante, este estudio se realiza fuera del ámbito de las instituciones.

Teorema 3.3.5. *Sea Σ una signatura de $SIG_{CoAlg(\mathcal{E}^D)}$ y sea e una ecuación de $Sen_{\mathcal{H}^D}(\Phi(\Sigma))$ con una única variable de género oculto. Entonces,*

$$(\mathcal{X},\gamma) \models_\Sigma^{CoAlg(\mathcal{E}^D)} \alpha_\Sigma(e) \quad \Longrightarrow \quad \beta_\Sigma((\mathcal{X},\gamma)) \models_{\Phi(\Sigma)}^{\mathcal{H}^D} e$$

para cualquier coálgebra $(\mathcal{X},\gamma)$ de $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$.

Demostración. Sea $\Sigma = (S, \Omega \cup C)$ una signatura de $SIG_{CoAlg(\mathcal{E}^D)}$, sea $e = (X, t, u)$ una ecuación de $Sen_{\mathcal{H}^D}(\Sigma_{imp})$ con $X_{imp_\Sigma} = \{z\}$, y $t, u \in T_{\Sigma_{imp}(X),s}$, $s \in S$, y sea $(\mathcal{X},\gamma)$ una coálgebra de $Mod_{CoAlg(\mathcal{E}^D)}(\Sigma)$. Debemos demostrar que para cada valoración $q\colon X \to \beta_\Sigma((\mathcal{X},\gamma))$, se verifica que $\overline{q}(t) = \overline{q}(u)$.

Sea $q\colon X \to \beta_\Sigma((\mathcal{X},\gamma))$ una valoración. Tenemos que $q_{imp_\Sigma}(z) = a$ para algún $a \in \beta_\Sigma((\mathcal{X},\gamma))_{imp_\Sigma}$. Entonces la restricción de q a los géneros visibles en S, $q_{|S}\colon X \setminus X_{imp_\Sigma} \to \beta_\Sigma((\mathcal{X},\gamma))_{|S}$, es una valoración por la Σ-álgebra $\gamma(a)$. Por hipótesis, obtenemos que $\overline{q_{|S}}(\phi(t)) = \overline{q_{|S}}(\phi(u))$. Si procedemos como antes por inducción sobre la estructura de un término t con una única variable oculta z, obtenemos que $\overline{q_{|S}}(\phi(t)) = \overline{q}(t)$, lo que prueba la implicación. ■

3.3.4 Otros tipos de morfismos

A pesar de que no hemos definido un morfismo de instituciones entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$, sí que obtenemos una versión más débil de relación entre instituciones llamada *pre-morfismo de instituciones*, noción que definimos a continuación. Las siguientes definiciones han sido extraídas de [68], y son una modificación de las publicadas en [85].

Definición 3.3.6 (Pre-institución). Una *pre-institución* $\mathcal{I} = (SIG, Sen, Mod, \models)$ consta de:

1. una categoría SIG, llamada *categoría de signaturas*,
2. un funtor $Sen\colon SIG \to Set$, llamado *funtor sentencias*,
3. un funtor $Mod\colon SIG \to Cat^{op}$, llamado *funtor modelos*, y
4. una relación $\models_\Sigma \subseteq Obj(Mod(\Sigma)) \times Sen(\Sigma)$ para cada $\Sigma \in Obj(SIG)$, llamada Σ-*satisfacción*.

Se dice que en una pre-institución $\mathcal{I}$

i. la *reducción preserva la satisfacción* (condición rps) si

$$A' \models_{\Sigma'} Sen(\mu)(e) \quad \Longrightarrow \quad Mod(\mu)(A') \models_\Sigma e$$

ii. la *expansión preserva la satisfacción* (condición eps) si

$$Mod(\mu)(A') \models_\Sigma e \quad \Longrightarrow \quad A' \models_{\Sigma'} Sen(\mu)(e)$$

para cada morfismo $\mu\colon \Sigma \to \Sigma'$ en SIG, cada $A' \in Obj(Mod(\Sigma'))$ y cada $e \in Sen(\Sigma)$.

Obviamente, una institución es una pre-institución que verifica las condiciones rps y eps.

Un ejemplo de pre-institución es la institución oculta en la que se cambian los morfismos de signaturas ocultos por morfismos verticales de signaturas ocultas. Si no exigimos la condición de encapsulamiento en los morfismos de signaturas ocultos obtenemos que la institución verifica la condición rps, pero no la condición eps (ver [25]).

Se puede definir ahora una relación entre dos pre-instituciones, lo que origina el concepto de pre-morfismo de pre-instituciones.

Definición 3.3.7 (Pre-morfismo de pre-instituciones). Sean $\mathcal{I} = (SIG, Sen, Mod, \models)$, $\mathcal{I}' = (SIG', Sen', Mod', \models')$ dos pre-instituciones. Un *pre-morfismo de pre-instituciones* entre $\mathcal{I}$ e $\mathcal{I}'$, que denotamos por $\Phi\colon \mathcal{I} \dashrightarrow \mathcal{I}'$, consta de:

1. un funtor $\Phi\colon SIG \to SIG'$,
2. una transformación natural $\alpha\colon Sen' \circ \Phi \Rightarrow Sen$, y
3. una transformación natural $\beta\colon Mod \Rightarrow Mod' \circ \Phi$.

Se dice que en un pre-morfismo de pre-instituciones

i. que cumplen la condición rps, *la condición de morfismo es rps* si

$$A \models_\Sigma \alpha_\Sigma(e') \quad \Longrightarrow \quad \beta_\Sigma(A) \models'_{\Phi(\Sigma)} e'$$

ii. que cumplen la condición eps, *la condición de morfismo es eps* si

$$\beta_\Sigma(A) \models'_{\Phi(\Sigma)} e' \implies A \models_\Sigma \alpha_\Sigma(e')$$

para cualquier signatura Σ de $\mathcal{I}$, cualquier Σ-modelo A de $\mathcal{I}$ y cualquier $\Phi(\Sigma)$-sentencia e' de $\mathcal{I}'$.

Obviamente, un morfismo de instituciones es un pre-morfismo de instituciones en el que la condición de morfismo es rps y eps.

Se pueden componer pre-morfismos de pre-instituciones (verificando la condición rps o eps) de la forma esperada, de modo que los pre-morfismos de pre-instituciones junto con las pre-instituciones (que cumplan la respectiva condición rps o eps) forman una categoría.

3.3.5 Segundo diagrama alternativo

Si utilizamos los conceptos que hemos definido en la sección anterior, observamos que la relación que hemos determinado anteriormente entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$ es un pre-morfismo de instituciones que verifica la condición eps. Si utilizamos este morfismo podemos completar el siguiente diagrama conmutativo de pre-morfismos de instituciones que verifican dicha condición:

$$\begin{array}{ccc} & & CoAlg(\mathcal{E}^D) \\ & \nearrow & \downarrow \\ \mathcal{E}^D & \longrightarrow & \mathcal{H}^D \\ & \searrow & \downarrow \\ & & \mathcal{E} \end{array}$$

Para comprobar que el diagrama realmente conmuta únicamente falta por verificar que conmutan las trasformaciones naturales de las sentencias. En el fragmento inferior del diagrama conmutan de modo inmediato ya que la transformación natural que relaciona las sentencias entre las instituciones $\mathcal{H}^D$ y $\mathcal{E}$ es la identidad y las transformaciones naturales de los otros dos morfismos están definidas de la misma forma. Lo mismo ocurre con la parte superior, en el que la transformación natural que es la identidad en este caso es la transformación que relaciona las sentencias entre las instituciones $\mathcal{E}^D$ y $CoAlg(\mathcal{E}^D)$.

Como tenemos perfectamente localizado el conjunto de sentencias para las cuales no se tiene la condición de morfismo rps, surge la idea de definir un nuevo tipo de morfismo de instituciones en el que no sea necesario que se verifique la condición de satisfacibilidad sobre todas las sentencias de una institución sino únicamente sobre un subconjunto de las mismas. Una idea parecida fue propuesta en [25] para delimitar los modelos y las sentencias sobre las que se verifica la condición eps de la pre-institución oculta con morfismos verticales de signaturas ocultas.

Definición 3.3.8 (Morfismo de instituciones restringido a un conjunto de sentencias). Sean $\mathcal{I} = (SIG, Sen, Mod, \models)$, $\mathcal{I}' = (SIG', Sen', Mod', \models')$ dos instituciones y sea $\mathcal{C}' = \{\mathcal{C}'_{\Sigma'} \subseteq Sen'(\Sigma')\}_{\Sigma' \in Obj(SIG_{\mathcal{I}'})}$ un conjunto indexado de sentencias. Un *morfismo de instituciones restringido al conjunto de sentencias* $\mathcal{C}'$ entre $\mathcal{I}$ e $\mathcal{I}'$, que denotamos por $\Phi\colon \mathcal{I} \xrightarrow{\mathcal{C}'} \mathcal{I}'$, es un pre-morfismo de instituciones $\Phi = (\Phi, \alpha, \beta)\colon \mathcal{I} \dashrightarrow \mathcal{I}'$ tal que para cualquier signatura Σ de $\mathcal{I}$, se verifica que

$$A \models_\Sigma \alpha_\Sigma(e') \quad \text{si y sólo si} \quad \beta_\Sigma(A) \models'_{\Phi(\Sigma)} e'$$

para cualquier Σ-modelo A de $\mathcal{I}$ y cualquier sentencia $e' \in \mathcal{C}'_{\Phi(\Sigma)}$

Es evidente que un morfismo de instituciones restringido a un conjunto de sentencias $\mathcal{C}'$ en el que $\mathcal{C}'_{\Sigma'} = \emptyset$, para todo Σ' de $SIG_{\mathcal{I}'}$ es un pre-morfismo de instituciones, y en el que $\mathcal{C}'_{\Sigma'} = Sen'(\Sigma')$, para todo Σ' de $SIG_{\mathcal{I}'}$ es un morfismo de instituciones (sin apellidos).

Además, es claro que un morfismo de instituciones es, en particular, un morfismo de instituciones restringido a cualquier subconjunto de sentencias de la institución de llegada. No obstante, en caso de que no se verifique esta condición de satisfacibilidad sobre todas las sentencias, lo interesante es tratar de localizar el subconjunto mayor de sentencias sobre las que sí se verifica. Por ejemplo, es claro que es posible elegir todas las sentencias para aquellas signaturas de la institución de llegada que no sean imagen de una signatura de la institución de partida. Debemos también notar que la condición que hemos impuesto no afecta a la transformación natural sobre las sentencias que forma parte del morfismo, únicamente lo hace sobre la condición de satisfacibilidad.

Posteriormente, a partir de un morfismo de instituciones restringido a un conjunto de sentencias, trataremos de definir un morfismo de instituciones, prescindiendo en la segunda institución de las sentencias fuera de ese conjunto. Para realizar esta construcción no nos servirán todos los subconjuntos de sentencias, sino sólo aquéllos que verifiquen unas ciertas condiciones.

Si utilizamos la definición anterior podemos enunciar el siguiente corolario.

Corolario 3.3.9. *El pre-morfismo de instituciones entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$ que hemos definido anteriormente es un morfismo de instituciones restringido a un conjunto de sentencias $\mathcal{C}^h$, $\Phi\colon CoAlg(\mathcal{E}^D) \xrightarrow{\mathcal{C}^h} \mathcal{H}^D$, donde $\mathcal{C}^h$ está formado por:*

- $\mathcal{C}^h_{H\Sigma} = \{e \in Sen_{\mathcal{H}^D}(H\Sigma) \mid e$ *es una ecuación en una variable oculta*$\}$, *si $H\Sigma$ es una signatura deconstructora, con un único género oculto, y*
- $\mathcal{C}^h_{H\Sigma} = Sen_{\mathcal{H}^D}(H\Sigma)$, *en otro caso.*

En el corolario anterior utilizamos implícitamente que únicamente son imagen por el funtor de signaturas del pre-morfismo de instituciones aquellas signaturas ocultas deconstructoras con un único género oculto. De esta forma nos quedamos con las sentencias para las cuales se establece una buena relación entre el marco coalgebraico y el marco oculto, que coinciden con las propuestas por Cîrstea en [22] para nuestro caso particular.

El objetivo que nos proponemos es incluir en la parte superior de nuestro diagrama esta nueva versión de morfismo de instituciones. Para ello es necesario definir la composición de dos morfismos de instituciones restringidos a un conjunto de sentencias y, más en general, comprobar si las instituciones junto con los morfismos de instituciones restringidos a un conjunto de sentencias definen una categoría.

Definimos la composición de dos morfismos de instituciones restringidos a un conjunto de sentencias, como un morfismo de instituciones restringido al mayor conjunto de sentencias posible para que dicho morfismo esté bien definido. Por lo tanto, estas sentencias deben elegirse entre el conjunto de sentencias del segundo morfismo cuya imagen por la transformación natural correspondiente caiga dentro del conjunto de sentencias del primer morfismo.

Definición 3.3.10. Dadas tres instituciones $\mathcal{I}$, $\mathcal{I}'$ y $\mathcal{I}''$ y dos pre-morfismos de instituciones restringidos a un conjunto de sentencias $\Phi\colon \mathcal{I} \xrightarrow{\mathcal{C}'} \mathcal{I}'$ y $\Phi' = (\Phi', \alpha', \beta')\colon \mathcal{I}' \xrightarrow{\mathcal{C}''} \mathcal{I}''$, donde $\mathcal{C}' = \{\mathcal{C}'_{\Sigma'} \subseteq Sen'(\Sigma')\}_{\Sigma' \in Obj(SIG_{\mathcal{I}'})}$, y $\mathcal{C}'' = \{\mathcal{C}''_{\Sigma''} \subseteq Sen''(\Sigma'')\}_{\Sigma'' \in Obj(SIG_{\mathcal{I}''})}$, se define el *morfismo de instituciones restringido a un conjunto de sentencias composición* $\Phi' \circ \Phi$ como el morfismo de instituciones composición restringido al conjunto de sentencias $\mathcal{C}^2 = \{\mathcal{C}^2_{\Sigma''} \subseteq Sen''(\Sigma'')\}_{\Sigma'' \in Obj(SIG_{\mathcal{I}''})}$, donde

$$\mathcal{C}^2_{\Sigma''} = \{e \in \mathcal{C}''_{\Sigma''} \mid \alpha'_{\Sigma'}(e) \in \mathcal{C}'_{\Sigma'},\ \forall \Sigma' \in Obj(SIG_{\mathcal{I}'}) \text{ tal que } \Phi'(\Sigma') = \Sigma''\}$$

para cada $\Sigma'' \in Obj(SIG_{\mathcal{I}''})$.

Si consideramos, como es natural, el morfismo de instituciones restringido a un conjunto de sentencias identidad $id_{\mathcal{I}}$ como el morfismo de instituciones identidad (sin restringir a ninguna sentencia), entonces no es difícil comprobar que se verifica la regla de las identidades y que la composición de estos morfismos es asociativa. Obtenemos de este modo el siguiente teorema.

Teorema 3.3.11. *Las instituciones junto con los morfismos de instituciones restringidos a un conjunto de sentencias forman una categoría.*

Podemos incluir ahora en la sección superior del diagrama el morfismo de instituciones restringido a un conjunto de sentencias $\mathcal{C}^h$ entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$. Observamos que según la anterior definición de composición de morfismos de instituciones de este tipo, el morfismo entre $\mathcal{E}^D$ y $\mathcal{H}^D$ debe restringirse a este mismo conjunto de sentencias $\mathcal{C}^h$. Obtenemos el siguiente diagrama conmutativo:

$$\begin{tikzcd} & CoAlg(\mathcal{E}^D) \arrow[d, "\mathcal{C}^h"] \\ \mathcal{E}^D \arrow[ur] \arrow[r, "\mathcal{C}^h"] & \mathcal{H}^D \end{tikzcd}$$

Los conjuntos de sentencias a los que se restringen los anteriores morfismos hacen explícita la idea implícita de nuestra operación *imp*: sólo utilizaremos en la institución oculta aquellas ecuaciones en una única variable oculta. El equivalente en la institución algebraica ecuacional son las ecuaciones en una única variable en el género destacado. No obstante, no tenemos una forma natural de expresar este conjunto de sentencias dentro del marco algebraico ecuacional ya que hemos perdido la distinción entre géneros ocultos y visibles. Este hecho indica de nuevo que el marco adecuado en el que situar nuestra operación es el marco oculto.

Podemos pensar ahora en restringir el conjunto de sentencias de la institución $\mathcal{H}^D$, de modo que contenga solamente las sentencias que nos interesan. Obtenemos de esta forma una nueva institución: el correspondiente funtor sentencias asocia a cada signatura el conjunto de sus sentencias con una única variable oculta, y a cada morfismo de signaturas la correspondiente aplicación entre estas sentencias. Entonces se define de manera natural un morfismo de instituciones entre $CoAlg(\mathcal{E}^D)$ y esta nueva institución a partir del morfismo de instituciones restringido a un conjunto de sentencias entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$.

La construcción anterior puede hacerse en general, de modo que a partir de un morfismo de instituciones restringido a un conjunto de sentencias entre dos instituciones podemos definir un morfismo de instituciones si modificamos la institución de llegada mediante la restricción de sus sentencias a dicho conjunto. Para que esto esté bien definido, es necesario que este conjunto sea respetado por las aplicaciones que define el funtor sentencias para cada morfismo de signaturas.

Aunque se trata de una construcción bastante natural, comenzamos por presentar lo que se entiende por institución definida a partir de otra mediante restricción de sentencias.

Definición 3.3.12 (Institución definida a partir de otra restringiendo su conjunto de sentencias). Sea $\mathcal{I} = (SIG_{\mathcal{I}}, Sen_{\mathcal{I}}, Mod_{\mathcal{I}}, \models^{\mathcal{I}})$ una institución y sea $\mathcal{C} = \{\mathcal{C}_\Sigma \subseteq Sen_{\mathcal{I}}(\Sigma)\}_{\Sigma \in Obj(SIG_{\mathcal{I}})}$ un conjunto indexado de sentencias, tal que para cada morfismo $\mu\colon \Sigma \to \Sigma'$ de $SIG_{\mathcal{I}}$, se cumpla que $Sen(\mu)(e) \in \mathcal{C}_{\Sigma'}$ para cada $e \in \mathcal{C}_\Sigma$. La *institución definida a partir de $\mathcal{I}$ con sentencias en $\mathcal{C}$*, que denotamos por $\mathcal{I}|_{\mathcal{C}} = (SIG_{\mathcal{I}|_{\mathcal{C}}}, Sen_{\mathcal{I}|_{\mathcal{C}}}, Mod_{\mathcal{I}|_{\mathcal{C}}}, \models^{\mathcal{I}|_{\mathcal{C}}})$, es la institución que tiene como categoría de signaturas, funtor modelos y relación de satisfacción las respectivas componentes de $\mathcal{I}$. El funtor sentencias $Sen_{\mathcal{I}|_{\mathcal{C}}}\colon SIG_{\mathcal{I}} \to Set$ asocia a cada signatura Σ de $SIG_{\mathcal{I}}$ el conjunto $Sen_{\mathcal{I}|_{\mathcal{C}}}(\Sigma) = \mathcal{C}_\Sigma$, y a cada morfismo de signaturas $\mu\colon \Sigma \to \Sigma'$ de $SIG_{\mathcal{I}}$ la aplicación $Sen_{\mathcal{I}|_{\mathcal{C}}}(\mu)\colon \mathcal{C}_\Sigma \to \mathcal{C}_{\Sigma'}$ tal que $Sen_{\mathcal{I}|_{\mathcal{C}}}(\mu)(e) := Sen_{\mathcal{I}}(\mu)(e)$, para cada $e \in \mathcal{C}_\Sigma$.

La institución anterior está bien definida ya que $Sen_{\mathcal{I}|_{\mathcal{C}}}$ es un funtor. Preserva las identidades y la composición de morfismos de signaturas, al ser éstas preservadas por $Sen_{\mathcal{I}}$. La relación de satisfacibilidad se verifica trivialmente sobre un subconjunto de sentencias al verificarse sobre todo el conjunto.

La condición que exige que $Sen(\mu)(e) \in \mathcal{C}_{\Sigma'}$ para cada $e \in \mathcal{C}_\Sigma$ es necesaria para poder definir $Sen_{\mathcal{I}|_{\mathcal{C}}}$. Esta fuerte condición se obtiene de modo natural en nuestro caso particular.

Ahora, podemos definir un morfismo de instituciones entre una institución y cualquier institución que se origine a partir de ella tras restringir su conjunto de sentencias. Este hecho queda patente en el siguiente resultado, cuya demostración es evidente.

Lema 3.3.13. *Sean $\mathcal{I}$ una institución e $\mathcal{I}|_{\mathcal{C}}$ la institución definida a partir de $\mathcal{I}$ con sentencias en un conjunto $\mathcal{C}$. Entonces, la inclusión natural $i = (id_{SIG_{\mathcal{I}}}, i_{\mathcal{I}|_{\mathcal{C}}}, id_{Mod_{\mathcal{I}}})$ dada por la identidad en el nivel de signaturas y modelos, y la inclusión en el nivel de sentencias es un morfismo de instituciones.*

Por fin, en el siguiente resultado construimos a partir de un morfismo de instituciones restringido a un conjunto de sentencias entre dos instituciones, un morfismo de instituciones entre la primera institución y la segunda institución restringida a este conjunto de sentencias.

Teorema 3.3.14. *Dadas dos instituciones $\mathcal{I}$, $\mathcal{I}'$ y un morfismo de instituciones restringido a un conjunto de sentencias $\mathcal{C}'$, $\Phi\colon \mathcal{I} \xrightarrow{\mathcal{C}'} \mathcal{I}'$, tal que para cada morfismo*

$\mu'\colon \Sigma'_1 \to \Sigma'_2$ de $SIG_{\mathcal{I}'}$, se cumpla que $Sen_{\mathcal{I}'}(\mu')(e') \in \mathcal{C}'_{\Sigma'_2}$ para cada $e' \in \mathcal{C}'_{\Sigma'_1}$, entonces podemos definir un morfismo entre $\mathcal{I}$ e $\mathcal{I}'|_{\mathcal{C}'}$ que hace conmutar el siguiente diagrama:

$$\begin{tikzcd}
\mathcal{I} \arrow[r, "\mathcal{C}'"] \arrow[dr] & \mathcal{I}' \arrow[d, "i"] \\
 & \mathcal{I}'|_{\mathcal{C}'}
\end{tikzcd}$$

donde i es el morfismo de instituciones inclusión entre $\mathcal{I}'$ e $\mathcal{I}'|_{\mathcal{C}'}$.

Demostración. Sea $\Phi = (\Phi, \alpha, \beta)\colon \mathcal{I} \xrightarrow{\mathcal{C}'} \mathcal{I}'$ un morfismo de instituciones restringido a un conjunto de sentencias $\mathcal{C}'$. Definimos un morfismo de instituciones $\Phi^{\mathcal{C}'} = (\Phi^{\mathcal{C}'}, \alpha^{\mathcal{C}'}, \beta^{\mathcal{C}'})\colon \mathcal{I} \to \mathcal{I}'|_{\mathcal{C}'}$ de la forma que indicamos a continuación.

El funtor entre las categorías de signaturas y la transformación natural entre los funtores modelos son los mismos que los de Φ, es decir, $\Phi^{\mathcal{C}'} := \Phi$ y $\beta^{\mathcal{C}'} := \beta$. La transformación natural $\alpha^{\mathcal{C}'}\colon Sen_{\mathcal{I}'|_{\mathcal{C}'}} \circ \Phi \Rightarrow Sen_{\mathcal{I}}$ viene dada por una familia de aplicaciones $\alpha^{\mathcal{C}'}_{\Sigma}\colon Sen_{\mathcal{I}'|_{\mathcal{C}'}}(\Phi(\Sigma)) \to Sen_{\mathcal{I}}(\Sigma)$, una para cada Σ de $SIG_{\mathcal{I}}$, siendo $\alpha^{\mathcal{C}'}_{\Sigma}(e) := \alpha_{\Sigma}(e)$ para cada $e \in Sen_{\mathcal{I}'|_{\mathcal{C}'}}(\Phi(\Sigma))$. Es directo probar que estas tres piezas definen un morfismo de instituciones. ■

Podemos aplicar este resultado a los morfismos de instituciones entre $CoAlg(\mathcal{E}^D)$ y $\mathcal{H}^D$, y entre $\mathcal{E}^D$ y $\mathcal{H}^D$, restringidos ambos al conjunto de sentencias $\mathcal{C}^h$ anteriormente fijado, ya que para cada morfismo de signaturas oculto $\mu\colon H\Sigma \to H\Sigma'$ de $\mathcal{H}^D$, y para cada $e \in \mathcal{C}^h_{H\Sigma}$ se verifica que $Sen_{\mathcal{H}^D}(\mu)(e) \in \mathcal{C}^h_{H\Sigma'}$. Observamos que si $H\Sigma$ es una signatura deconstructora con único género oculto y e es una ecuación en una variable oculta, entonces, si $H\Sigma'$ es deconstructora con único género oculto, $Sen_{\mathcal{H}^D}(\mu)(e)$ es una ecuación en una variable oculta.

Si aplicamos el teorema anterior a estos morfismos obtenemos el siguiente diagrama conmutativo:

$$\begin{tikzcd}
 & CoAlg(\mathcal{E}^D) \arrow[d, "\mathcal{C}^h"] \arrow[dr] & \\
\mathcal{E}^D \arrow[ur] \arrow[r, "\mathcal{C}^h"] \arrow[rr, bend right] & \mathcal{H}^D \arrow[r] & \mathcal{H}^D|_{\mathcal{C}^h}
\end{tikzcd}$$

En nuestro contexto tiene sentido definir un morfismo entre instituciones en el que restringimos el conjunto de sentencias sobre el que se verifica la condición de satisfacibilidad. No obstante, podemos también generalizar la definición de morfismo de instituciones de modo que la condición de satisfacibilidad se verifique, en lugar de sobre un conjunto particular de sentencias, sobre un conjunto particular de modelos o incluso sobre un conjunto particular de signaturas. Obtenemos de esta forma dos nuevas nociones de morfismo de instituciones. Debemos observar que las restricciones sobre modelos y signaturas en estos morfismos se deben realizar sobre la institución de salida del morfismo y no sobre la institución de llegada como ocurre en el caso de las sentencias.

Estos morfismos restringidos a una subcategoría de modelos o signaturas también originan otras instituciones, que se definen restringiendo sus modelos o signaturas a dichas categorías, siempre que verifiquen una condición parecida al caso de las sentencias. Y, como es de suponer, a partir de un morfismo de instituciones restringido a una categoría de modelos o signaturas podemos definir un morfismo de instituciones entre la institución restringida (en esta ocasión la restricción se realiza sobre la institución de salida) y la institución de llegada, de modo que junto con el morfismo de instituciones inclusión originan también un diagrama conmutativo.

En realidad, podemos definir un morfismo entre dos instituciones que permita posibles restricciones en los tres ingredientes que intervienen en la condición de satisfacibilidad: signaturas, sentencias y morfismos.

Para terminar el capítulo, indicaremos que podemos hacer un estudio de nuestro diagrama que incorpore por un lado la generalización de los nodos que lo integran, que hemos conseguido en el primer diagrama alternativo, y por otro el cambio en los morfismos, que hemos realizado en el segundo diagrama alternativo. De este modo, podemos obtener un diagrama de morfismos de instituciones (sin apellidos) en el que los nodos sean más generales (por ejemplo, que la institución $\mathcal{E}^D$ presente morfismos de signaturas distintos de la identidad).

Capítulo 4

Especificación algebraica de relaciones entre estructuras de datos en EAT

4.1 Introducción

En los capítulos anteriores hemos tratado, desde la perspectiva de las instituciones, diferentes marcos de especificación que han sido usados en [72] para modelizar las estructuras de datos presentes en EAT. En todos ellos se parte de un universo de datos fijado sobre el que se construyen los modelos. Inicialmente, esto puede parecer bastante restrictivo. Sin embargo, refleja lo que habitualmente se hace en la práctica en programación, y en particular es así como se trabaja en EAT. Este universo de datos se corresponde con los tipos de datos que tiene el lenguaje de programación empleado, y que son utilizados en los programas que se implementan usando dicho lenguaje.

No obstante, la aparente rigidez que supone el trabajar con un conjunto de datos prefijado, puede ser superada por al menos dos procedimientos.

Por un lado, es posible que nos interese que distintas expresiones de estos datos básicos representen dentro de los modelos al mismo elemento. Para ello podemos incluir dentro del conjunto de datos relaciones de equivalencia que definan igualdades en este conjunto. De esta forma, a pesar de trabajar con un conjunto de datos fijado, podemos obtener distintos dominios de datos si consideramos cocientes en los mismos respecto de estas relaciones de equivalencia. Así, si consideramos como conjunto de datos $D = \mathbb{Z}$, que representa el tipo de datos `integer`, entonces podemos trabajar con la clase de estructuras, por ejemplo grupos, que se pueden construir con conjunto base D. En principio, en esta clase no están incluidos los grupos finitos como $\mathbb{Z}/n\mathbb{Z}$. Sin embargo, sí que podemos *modelar* estos grupos finitos sobre D si permitimos definir relaciones de equivalencia en D, que identifiquen de forma adecuada sus elementos.

Por otro lado, podemos permitir que las operaciones de los modelos no estén definidas sobre todos los elementos del dominio de datos D, es decir, admitimos *parcialidad* en nuestros modelos. De este modo ampliamos las posibilidades de especificación ya que, en la práctica, los programas raramente están bien definidos para cada dato de entrada sintácticamente correcto. En particular, en el campo de la Topología Algebraica, muchas de las estructuras que se manejan están basadas en conjuntos *graduados*, y esto implica que las operaciones sólo están definidas entre elementos del mismo grado. Más adelante estudiaremos los casos particulares de conjunto simplicial y complejo de cadenas.

Una vez tratado el modelado de las estructuras de datos de EAT, el siguiente paso será estudiar cómo tratar las relaciones entre esas estructuras. Por ejemplo, en EAT se dispone de un operador que a partir de un conjunto simplicial construye el complejo de cadenas asociado a ese conjunto simplicial. Debemos observar que este operador corresponde a un funtor (aunque sólo nos interesa su definición en el nivel de objetos) entre la categoría de los conjuntos simpliciales y la categoría de los complejos de cadenas. Este capítulo se dedica a la formalización de este tipo de relaciones entre estructuras de datos.

Para dar una especificación de los conjuntos simpliciales y de los complejos de cadenas podemos utilizar la construcción *imp* desarrollada en los capítulos anteriores. Dada una signatura pensada para especificar un determinado tipo de estructuras, la operación *imp* daba lugar a una signatura con un género oculto, que representaba a familias de estructuras del tipo de partida. No obstante, para dotar a nuestro sistema de mayor capacidad necesitamos incorporar las dos características que hemos comentado, que amplían la capacidad de especificación de la teoría oculta, como son la introducción de clases de equivalencia y la parcialidad. Estas dos características serán utilizadas en la especificación de los conjuntos simpliciales y los complejos de cadenas. Lo natural es pensar que el modo de especificar el operador al que nos hemos referido entre conjuntos simpliciales y complejos de cadenas, consistirá en añadir una nueva operación entre los géneros destacados de las signaturas para estas estructuras. Habrá que estudiar cómo trasladar a nivel de las categorías *imp* la construcción funtorial entre las estructuras de partida.

Debemos destacar que nosotros no estamos defendiendo que los conjuntos simpliciales, los complejos de cadenas o el funtor entre ambos no puedan ser especificados de otras formas, por ejemplo usando álgebras totales o especificación de orden superior (como las presentadas en [11] o [3]). Cuando una estructura algebraica es lo suficientemente rica, la definición de una signatura para ella requiere de numerosas decisiones de diseño (ver en [65] las muy diversas presentaciones que son posibles para la categoría de los conjuntos simpliciales, por ejemplo). Sin embargo, el objetivo de este trabajo no es analizar las diferentes alternativas para especificar estas estructuras de datos. Nuestro objetivo es reflejar, de modo tan fiel como sea posible, las características de un sistema de software real que está ofreciendo resultados muy satisfactorios como es EAT [81].

Aunque se han definido instituciones que tienen las características que pretendemos incorporar a nuestro sistema de especificación, como pueden ser instituciones de álgebras parciales, instituciones de álgebras con igualdades e incluso instituciones que incluyen ambas características (ver, por ejemplo, [39, 96, 98]), preferimos situarnos en un marco en el que nos resulte cómodo realizar nuestros propósitos de modelado sin preocuparnos de estudiar si éste constituye una institución o si la introducción de un determinado tipo de operadores tiene o no alguna repercusión en la relación entre instituciones. Será preciso realizar un trabajo posterior para responder a estas cuestiones.

En el capítulo de preliminares definimos una especificación como una pareja formada por signatura y un subconjunto de sus ecuaciones (que suele interpretarse como un conjunto de axiomas). Asignar significado a una especificación requiere definir qué clase de objeto matemático va a representar dicho significado, es decir, decidir qué *denota* una especificación. Cualquiera que sea la respuesta que elijamos tenemos que, como mínimo, una especificación debe determinar el conjunto de operaciones del objeto a especificar, es decir, debe contener al menos una signatura. A partir de esta signatura, en [86] se indica que podemos elegir la semántica para una especificación en al menos tres niveles:

- Lo que allí se denomina *semántica de presentación*, esto es, una especificación denota una signatura y un conjunto de axiomas para esta signatura. En este caso, una especificación es algo puramente sintáctico.

- *Semántica de teorías* en la que una especificación está formada por una signatura y un conjunto de axiomas que forman una teoría. En este caso el significado de la especificación no es exclusivamente sintáctico.

- *Semántica basada en modelos* en la que una especificación es una signatura junto con una clase de álgebras para esa signatura.

En nuestro caso, el objetivo último de una especificación es determinar una clase de álgebras, que es lo que nos permite modelar las estructuras sobre las que los programas trabajan. De este modo estamos de acuerdo con lo expresado por Sannella y Tarlecki en [86] y Reichel en [75], al considerar que la semántica basada en modelos es la más conveniente.

Entre las aproximaciones semánticas basadas en modelos de las especificaciones destacamos: la *laxa* ("loose semantics"), en la que la clase de modelos lo constituyen aquellas álgebras de la signatura que satisfacen los axiomas de la especificación [63, 75]; la *inicial*, en la que el modelo definido por la especificación es el álgebra inicial dentro de la categoría de modelos que satisfacen los axiomas [36, 63]; la *final*, en la que el modelo definido por la especificación es el objeto final [99, 74]; la *basada en modelos pura*, en la que la especificación incorpora una clase de álgebras cerrada por isomorfismos, esto es, un TAD, un Tipo Abstracto de Datos [63, 75, 101, 36].

A partir de ahora, en lugar de dar la presentación de una especificación a través de una signatura y de un conjunto de axiomas, nosotros usaremos directamente su significado. El significado que adoptaremos para una especificación será el de una categoría de modelos cerrada por isomorfismo, es decir, el de un TAD. La forma de señalar las álgebras que integran la clase será a través de restricciones semánticas. De este modo dispondremos de más herramientas matemáticas que nos permitan modelizar de forma más cómoda nuestras estructuras. Así, nuestro enfoque va a ser basado exclusivamente en modelos, como en [63] (Capítulo 2) o en [61], en lugar de un enfoque axiomático, continuando con el marco usado por V. Pascual en [72] para modelar las estructuras de EAT. En lo que sigue, cuando hablemos de "especificación", lo que queremos decir es una signatura junto con una clase de álgebras para ella.

La estructura de este capítulo es la siguiente. En la Sección 4.4 incorporaremos a nuestro sistema de especificación las dos características comentadas en esta introducción, que permiten ampliar las posibilidades de especificación cuando trabajamos con dominios prefijados, como son la parcialidad y la posibilidad de establecer equivalencias dentro del dominio de datos. Previamente dedicamos las dos próximas secciones a definir una serie de conceptos básicos que nos serán útiles para llevar a cabo esta tarea. En concreto en la Sección 4.2 introducimos las álgebras parciales y en la Sección 4.3 las álgebras con igualdad. En la Sección 4.5 estudiaremos las repercusiones que la anterior ampliación tiene en la operación *imp*. La última sección del capítulo se dedica a utilizar el tratamiento anterior para el modelado de un tipo especial de relaciones entre estructuras presente en EAT.

4.2 Álgebras parciales

Cuando se trata de especificar sistemas de programación reales es frecuente encontrar operadores que son parciales. Estos operadores no están definidos para ciertos argumentos, produciendo normalmente errores si se intentan utilizar sobre ellos. Se puede especificar estos operadores a través de operaciones totales. Para ello es suficiente añadir a cada conjunto soporte un elemento especial que represente un "valor error". No obstante, el añadir este elemento, además de no ser una solución natural, origina multitud de problemas por el trato especial que debe darse al mismo (por ejemplo, dicho elemento pasa a formar parte de los argumentos de las operaciones). Resulta más natural permitir que las operaciones de las álgebras no tengan por qué estar definidas en todo su dominio, es decir, permitir operaciones parciales.

El marco de la especificación algebraica ecuacional que hemos presentado en el capítulo de preliminares es el marco clásico de especificación algebraica. A partir de éste se han introducido una gran variedad de modificaciones para incrementar su poder expresivo de modo que tengan en cuenta la variedad de características de los sistemas de software. No obstante, como es normal, a medida que se va enriqueciendo el marco de especificación la teoría se hace a su vez más compleja y más difícil de entender.

En esta sección estudiaremos una ampliación de la especificación algebraica ecuacional, para poder trabajar con parcialidad dentro de las álgebras. Existen numerosos trabajos que han desarrollado este tema. Nosotros hemos utilizado entre otros [20], [14], [11] y [63]. A continuación ofrecemos una serie de definiciones básicas que pueden encontrarse en cualquiera de ellos.

Definición 4.2.1 (Álgebra parcial). Sea $\Sigma = (S, \Omega)$ una signatura. Un *álgebra parcial* A para Σ (o Σ-*álgebra parcial*) es un par de familias $A = \langle (A_s)_{s \in S}, (\omega_A, Def(\omega_A))_{\omega \in \Omega} \rangle$. Para cada género $s \in S$, A_s es un conjunto, y para cada operación $\omega \in \Omega$, de perfil $\omega\colon s_1 \ldots s_n \to s$, ω_A es una función parcial, $\omega_A\colon A_{s_1} \times \cdots \times A_{s_n} \to A_s$, con dominio de definición $Def(\omega_A)$. Como en el caso total, el conjunto A_s se llama *soporte* en A de género s, y la función parcial ω_A se llama *interpretación* en A de la operación ω.

Una Σ-álgebra parcial en el que todas sus operaciones sean totales es una Σ-álgebra total.

Ejemplo 4.2.2. Un ejemplo típico que sirve para ilustrar la noción de álgebra parcial viene dado a través del álgebra de los números naturales que contiene, además de la operación "siguiente", la operación "predecesor":

signatura `NAT`
  géneros *nat*
  operaciones
    $0\colon \ \to nat$
    $sig\colon nat \to nat$
    $pred\colon nat \to nat$
finsig,

Un álgebra parcial A para esta signatura viene dada por: el soporte $A_{nat} = \mathbb{N}$, la constante $0_A = 0$, la función total $sig_A(n) = n+1$, y la función parcial $pred_A(n) = n-1$, con $Def(pred_A) = \mathbb{N} \setminus \{0\}$. □

De la misma forma que se ha generalizado la noción de álgebra, es necesario modificar la noción de morfismo entre álgebras, de manera que sea adecuada para establecer relaciones entre álgebras parciales. Existen en la literatura varias definiciones de morfismo entre álgebras parciales según el distinto tratamiento que podemos dar a los elementos que no están dentro del dominio de definición de alguna operación. Presentamos a continuación dos definiciones diferentes de morfismo que utilizaremos posteriormente.

Definición 4.2.3 (Homomorfismo débil y fuerte). Sean $\Sigma = (S, \Omega)$ una signatura y A, B dos Σ-álgebras parciales.

i) Un Σ-*homomorfismo débil* de A en B, $f\colon A \to B$, es una familia de funciones totales, $(f_s\colon A_s \to B_s)_{s \in S}$, tal que para cada operación $\omega\colon s_1 \ldots s_n \to s \in \Omega$, $n \geq 0$, la siguiente condición es cierta:

   si $(a_1, \ldots, a_n) \in Def(\omega_A)$, entonces $(f_{s_1}(a_1), \ldots, f_{s_n}(a_n)) \in Def(\omega_B)$ y en este caso
   $$f_s(\omega_A(a_1, \ldots, a_n)) = \omega_B(f_{s_1}(a_1), \ldots, f_{s_n}(a_n)).$$

ii) Un Σ-*homomorfismo fuerte* de A en B, $f\colon A \to B$, es un Σ-homomorfismo débil con la siguiente característica adicional:

$$\text{si } (a_1, \ldots, a_n) \notin Def(\omega_A), \text{ entonces } (f_{s_1}(a_1), \ldots, f_{s_n}(a_n)) \notin Def(\omega_B).$$

Un homomorfismo débil es una familia de funciones *totales* entre los conjuntos soportes de dos álgebras parciales, de manera que se dé la condición de homomorfismo para los elementos en el dominio de definición de la primera álgebra, sin preocuparse de los elementos que no estén en dicho dominio de definición. El homomorfismo fuerte es, como su nombre sugiere, más exigente. Obliga a que los elementos no definidos de un álgebra queden asociados con elementos no definidos en la otra. De esta forma, da lugar a un homomorfismo sobre álgebras totales si asociamos todos los elementos sobre los que no se tiene definición con un elemento destacado, por ejemplo, un elemento "error".

Ejemplo 4.2.4. Para ilustrar los conceptos anteriores definimos dos álgebras parciales B y C para la signatura `NAT`.

1. La `NAT`-álgebra B está definida por el soporte para el género nat $B_{nat} = \mathbb{Z}$, y las funciones totales para las operaciones $0_B = 0$, $sig_B(n) = n+1$ y $pred_B(n) = n-1$, para todo $n \in B_{nat}$.

2. La `NAT`-álgebra C viene dada por el mismo soporte para el género nat y la misma constante para 0 que B y las funciones parciales $sig_C(n) = n + 1$, con dominio de definición $Def(sig_C) = \{n \in C_{nat} \mid n \geq 0\}$, y $pred_C(n) = n - 1$, con dominio de definición $Def(sig_C) = \{n \in C_{nat} \mid n > 0\}$.

Los morfismos de signaturas inclusión $A \hookrightarrow B$ y $A \hookrightarrow C$ que vienen inducidos por la función inclusión de $\mathbb{N}$ en $\mathbb{Z}$ (donde A es la `NAT`-álgebra definida en el ejemplo anterior) constituyen respectivamente un homomorfismo de signaturas débil que no es fuerte de A en B, y un homomorfismo de signaturas fuerte de A en C. □

Para una signatura Σ, las Σ-álgebras parciales junto con los Σ-homomorfismos débiles (fuertes, respectivamente) definen una categoría que denotaremos por $PAlg(\Sigma)$ ($P_f Alg(\Sigma)$, respectivamente).

Dada una signatura Σ, la categoría de álgebras (totales), $Alg(\Sigma)$, y las dos categorías anteriores están relacionadas de modo que $Alg(\Sigma)$ es una subcategoría no plena de $P_f Alg(\Sigma)$, y ésta a su vez es una subcategoría no plena pero con los mismos objetos de $PAlg(\Sigma)$.

Un Σ-homomorfismo débil es un isomorfismo si las funciones que lo forman son biyectivas y las inversas de éstas forman un Σ-homomorfismo débil. Un Σ-homomorfismo fuerte es un isomorfismo si es biyectivo.

Debemos observar que la noción de TAD como subcategoría de álgebras cerrada por isomorfismo depende de la noción de morfismo que se considere dentro de la categoría de álgebras para una signatura, fundamentalmente porque puede variar la noción de isomorfismo que se tenga para ella. Esto ocurre por ejemplo con $PAlg(\Sigma)$ y $P_fAlg(\Sigma)$.

Definimos a continuación los conceptos de relación de congruencia y de álgebra cociente en el contexto parcial. Éstos se basan en el concepto de relación de equivalencia parcial que hemos extraído de [76].

Definición 4.2.5 (Relación de equivalencia parcial). Sean A un conjunto y R una relación en A. Se dice que R es una *relación de equivalencia parcial* si es simétrica y transitiva.

Una relación de equivalencia parcial R en un conjunto A es en realidad una relación de equivalencia (total) en un subconjunto de A (el formado por aquéllos elementos que aparecen en algún par de R).

La definición de relación de congruencia parcial en un álgebra parcial es la adaptación al caso parcial de la definición de congruencia habitual en álgebras totales. Como en ese caso, una relación de congruencia parcial en un álgebra parcial permite definir un álgebra cociente (parcial). Estas definiciones pueden encontrarse por ejemplo en [20] o en [14].

Definición 4.2.6 (Relación de congruencia parcial). Sean $\Sigma = (S, \Omega)$ una signatura y A una Σ-álgebra parcial. Una *relación de congruencia parcial en un álgebra parcial* A es una familia $Q = (Q_s)_{s\in S}$ de relaciones de equivalencia parciales, Q_s en A_s, $s \in S$, tal que para cada operación $\omega\colon s_1 \dots s_n \to s \in \Omega$, $n \geq 0$, se verifica que si $a_i\ Q_{s_i}\ b_i$ con $a_i, b_i \in A_{s_i}$, $1 \leq i \leq n$, y $(a_1, \dots, a_n), (b_1, \dots, b_n) \in Def(\omega_A)$, entonces $\omega_A(a_1, \dots, a_n)\ Q_s\ \omega_A(b_1, \dots, b_n)$.

Dada una relación de congruencia parcial Q en una Σ-álgebra parcial A, con $\Sigma = (S, \Omega)$, se llama *dominio* de Q a la familia de conjuntos $\{\{a \in A_s \mid a\ Q_s\ a\}\}_{s\in S}$. Si el dominio de una relación parcial en un álgebra coincide con los conjuntos soporte del álgebra en cada género, entonces la familia de relaciones de equivalencia parciales verifican la propiedad reflexiva por lo que son relaciones de equivalencia (totales). En este caso llamaremos a la relación de congruencia parcial sobre un álgebra parcial, simplemente relación de congruencia (total) sobre un álgebra parcial.

Definición 4.2.7 (Álgebra parcial cociente). Sean una signatura $\Sigma = (S, \Omega)$, una Σ-álgebra parcial A y una relación de congruencia parcial Q en A. El *álgebra parcial cociente* de A por Q es la Σ-álgebra parcial $A/_Q$ definida por:

- para cada género $s \in S$, $(A/_Q)_s = \{[a]_{Q_s} \mid a\ Q_s\ a\}$,
- para cada operación $\omega\colon s_1 \dots s_n \to s \in \Omega$, $n \geq 0$, $Def(\omega_{A/_Q}) = \{([a_1]_{Q_{s_1}}, \dots, [a_n]_{Q_{s_n}}) \in (A/_Q)_{s_1} \times \dots \times (A/_Q)_{s_n} \mid \text{existe } x_i \in [a_i]_{Q_{s_i}},\ i = 1, \dots, n, \text{ tal que } (x_1, \dots, x_n) \in Def(\omega_A)\}$; y para cada $([a_1]_{Q_{s_1}}, \dots, [a_n]_{Q_{s_n}}) \in Def(\omega_{A/_Q})$,
$$\omega_{A/_Q}([a_1]_{Q_{s_1}}, \dots, [a_n]_{Q_{s_n}}) = [\omega_A(x_1, \dots, x_n)]_{Q_s},$$
con $x_i \in [a_i]_{Q_{s_i}}$, $i = 1, \dots, n$, tal que $(x_1, \dots, x_n) \in Def(\omega_A)$.

4.3 Álgebras con igualdad

En esta sección trataremos la segunda de las características que nos permitirán ampliar la capacidad expresiva de nuestro sistema de especificación: la posibilidad de establecer identificaciones dentro de un conjunto de datos.

La definición fundamental es la de *signatura con igualdad.* A partir de una signatura, podemos definir una nueva signatura, si incorporamos a la misma un predicado binario por cada uno de sus géneros.

Definición 4.3.1 (Signatura con igualdad). Sea $\Sigma = (S, \Omega)$ una signatura. La *signatura con igualdad que se obtiene a partir de* Σ, que denotamos por $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$, se obtiene enriqueciendo Σ con el género *bool* y con una operación binaria $eq^s \colon s\ s \to bool$ por cada género $s \in S$. Es decir, $S^{eq} = S \cup \{bool\}$ y $\Omega^{eq} = \Omega \cup \{eq^s\}_{s \in S}$.

Para evitar confusión con los nombres, utilizaremos el renombrado de las operaciones de Ω en el caso en que el nombre de alguna de ellas coincida con el de las nuevas operaciones que se incorporan a Ω^{eq}.

Ejemplo 4.3.2. Si retomamos la signatura de grupo `GRP`, que tiene un único género $\{g\}$ y tres operaciones $\{prd\colon g\ g \to g, inv\colon g \to g, unt\colon\ \to g\}$, entonces la signatura $\mathtt{GRP}^{eq}$ está formada por los géneros $\{g, bool\}$ y las operaciones $\{prd\colon g\ g \to g,\ inv\colon g \to g,\ unt\colon\ \to g,\ eq^g\colon g\ g \to bool\}$. □

La presencia de predicados en las signaturas está ampliamente desarrollada en la literatura, usando variantes de la lógica de predicados [87, 39] para la construcción de las sentencias y la relación de satisfacibilidad. Un hecho especialmente importante para este tipo de signaturas es que el llamado *Lenguaje de Especificación Algebraica Común* CASL [98] soporta también predicados dentro de una lógica de primer orden parcial [20].

Debemos hacer notar que habitualmente se prefiere presentar estos predicados reunidos en un tercer conjunto que se desglosa en las signaturas acompañando a los conjuntos de géneros y operaciones. De este modo no es necesario representar estos predicados como operaciones con género resultado *bool*, sino que es suficiente con atribuir a cada uno de ellos los géneros que correspondan a sus argumentos. Entonces, la interpretación de un predicado de una signatura Σ^{eq} en un álgebra no será una función, sino una relación dentro del soporte correspondiente. Esta relación representará las tuplas de elementos en los que el predicado es verdad.

En [39] se usa una lógica de primer orden con igualdad, en la que se introduce en cada signatura una relación de igualdad por cada género de la misma. Estas signaturas están muy próximas a la nuestras. No obstante, se exige que los modelos para estas signaturas tengan como interpretación de estas relaciones igualdades literales sobre los soportes.

La diferencia fundamental con respecto a nuestras signaturas con igualdad es que las operaciones *eq* que se añaden representan test booleanos (posiblemente parciales y distintos de las igualdades literales). Estos operadores booleanos fueron implementados como componentes de las estructuras en EAT (ver [81]). Por ello pensamos que es necesario darles la misma interpretación sintáctica que al resto de los operadores. Estas operaciones nos permiten identificar elementos que se consideren "iguales" dentro de los soportes. Tras esta identificación, se obtiene el soporte para la estructura que pretendemos modelar. De esta forma su cometido semántico está claro. Las álgebras para una signatura con igualdad deben verificar que las funciones (posiblemente parciales) correspondientes a las igualdades deben constituir relaciones de congruencia (parciales) que permitan definir un álgebra cociente (parcial) a partir de ellas. Esta álgebra debe entonces representar a la estructura que pretendemos modelar. Observamos que a partir de un dominio de datos fijado podemos obtener diversos conjuntos de datos dependiendo de las identificaciones.

En muchas ocasiones al trabajar con parcialidad se suele distinguir en las signaturas dos conjuntos de operaciones, uno para las operaciones totales y otro para las parciales [20, 98]. Nosotros trabajaremos con signaturas parciales con igualdad. No obstante, preferimos no realizar este tipo de separaciones a nivel sintáctico (entre operaciones totales y parciales, o entre predicados y operaciones) de modo que no complicamos la notación, ya que para nuestro estudio tales distinciones no resultan beneficiosas.

Definimos a continuación el concepto de álgebra parcial con igualdad. Abordamos directamente el caso de álgebras parciales, ya que es éste el que necesitaremos, que supone una extensión del caso total.

Definición 4.3.3 (Álgebra parcial con igualdad). Sea $\Sigma = (S, \Omega)$ una signatura, y $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$ la correspondiente signatura con igualdad. Un *álgebra parcial con igualdad* para Σ^{eq} (o Σ^{eq}*-álgebra parcial con igualdad*) es una Σ^{eq}-álgebra parcial $A = \langle (A_s)_{s \in S^{eq}}, (\omega_A, Def(\omega_A))_{\omega \in \Omega^{eq}} \rangle$ tal que:

- $A_{bool} := \{true, false\}$ y eq_{bool} es la igualdad literal;
- para cada género $s \in S$, la función parcial $eq_A^s \colon A_s \times A_s \to \{true, false\}$ determina una relación de equivalencia parcial en A_s;
- la familia de relaciones $eq_A = \{eq_A^s\}_{s \in S^{eq}}$ es una relación de congruencia parcial en A.

Si $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$ es la signatura asociada a $\Sigma = (S, \Omega)$ y A es una Σ^{eq}-álgebra parcial con igualdad, podemos considerar la Σ^{eq}-álgebra parcial cociente $A/_{eq_A}$. Debemos observar entonces que las funciones parciales que interpretan a las operaciones eq son ahora en esta álgebra cociente igualdades literales parciales que no nos aportan información. Si prescindimos ahora de estas funciones obtenemos un álgebra para la signatura inicial Σ, que denotaremos por $\tilde{A}$.

La definición de homomorfismo de álgebras parciales con igualdad no varía con respecto a la de homomorfismo de álgebras parciales. De esta forma, dada una signatura

Σ^{eq} con igualdad obtenemos dos categorías de álgebras parciales con igualdad, que denotaremos $P\tilde{A}lg(\Sigma^{eq})$ y $P_f\tilde{A}lg(\Sigma^{eq})$, subcategorías plenas de $PAlg(\Sigma^{eq})$ y $P_fAlg(\Sigma^{eq})$ respectivamente, dependiendo de si utilizamos homomorfismos débiles o fuertes.

Aunque como hemos comentado no es nuestro objetivo analizar si el marco en el que nos situamos constituye o no una institución, la siguiente definición de morfismo de signaturas con igualdad invita a pensar en una institución en la que a las signaturas y álgebras con igualdad les acompañe algún tipo de ecuación parcial. Entonces, la relación de satisfacibilidad se definirá de modo que un álgebra parcial con igualdad verifica una ecuación si el álgebra cociente que se origina satisface, según la lógica parcial, dicha ecuación parcial.

Definición 4.3.4 (Morfismo de signaturas con igualdad). Sea $\mu\colon \Sigma \to \Sigma'$ un morfismo de signaturas, entre dos signaturas $\Sigma = (S, \Omega)$ y $\Sigma' = (S', \Omega')$, que está definido por la pareja de aplicaciones $\mu = (\mu_S\colon S \to S', \mu_\Omega\colon \Omega \to \Omega')$. El *morfismo de signaturas con igualdad que se obtiene a partir de* μ, que denotamos por $\mu^{eq}\colon \Sigma^{eq} \to \Sigma'^{eq}$, entre las signaturas $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$ y $\Sigma'^{eq} = (S'^{eq}, \Omega'^{eq})$, es el morfismo de signaturas dado por la pareja de aplicaciones $\mu^{eq} = (\mu^{eq}_S\colon S^{eq} \to S'^{eq}, \mu^{eq}_\Omega\colon \Omega^{eq} \to \Omega'^{eq})$ que se definen por: $\mu^{eq}_S(s) := \mu_S(s)$, para todo $s \in S$, y $\mu^{eq}_S(bool) := bool$; y $\mu^{eq}_\Omega(\omega) := \mu_\Omega(\omega)$, para todo $\omega \in \Omega$, y $\mu^{eq}_\Omega(eq^s) := eq^{\mu_S(s)}$, para todo $s \in S$.

A partir de la categoría $SIG_{\mathcal{E}}$ formada por las signaturas y morfismos de signaturas podemos definir una categoría $SIG^{eq}_{\mathcal{E}}$ formada por las signaturas y morfismos de signaturas que se obtienen a partir de los de $SIG_{\mathcal{E}}$ por la construcción anterior.

4.4 Incorporación de la parcialidad y de operaciones de igualdad

En esta sección ampliaremos la capacidad expresiva de los formalismos a la hora de realizar especificaciones. Para ello cambiaremos el marco de la especificación algebraica ecuacional por otro más complicado, pero que nos permitirá modelar estructuras también más complejas. En particular, vamos a incluir las dos nuevas características que hemos comentado previamente en nuestras signaturas y modelos. Por un lado, permitiremos parcialidad para las funciones en nuestros modelos. Esto nos obliga a trabajar con álgebras y homomorfismos parciales. Por otro lado, incluiremos en nuestras signaturas una operación de igualdad por cada género de las mismas. Esta operación nos permitirá definir identificaciones sobre los dominios de datos. De esta forma, como hemos comentado, a pesar de continuar trabajando sobre dominios de datos prefijados, podremos recoger distintos dominios de datos, bien porque establezcamos un cociente en este conjunto de datos o bien porque, a través de la parcialidad, trabajemos sobre subconjuntos de los mismos.

El punto de partida es una categoría $\mathcal{T}$, o más bien una clase de objetos, que pretendemos modelar (la clase de los grupos, la clase de los conjuntos simpliciales, etc.).

Entonces, como es habitual, es necesario determinar una signatura Σ con el objetivo de *obtener* Σ-álgebras que representen (en algún sentido) los objetos de $\mathcal{T}$. Sin embargo, repetimos que no es posible pensar en representar *cualquier* objeto de $\mathcal{T}$ (cualquier grupo, por ejemplo), tanto desde el punto de la computabilidad (normalmente $\mathcal{T}$ es una clase no numerable), como desde un punto de vista práctico (es muy útil, como se ha comentado previamente, identificar los conjuntos soporte de los géneros base con un *tipo* del lenguaje de programación utilizado). Por ello, dada una signatura $\Sigma = (S, \Omega)$ consideraremos fijado un *dominio de datos* $D = \{D_s\}_{s \in S}$ para ella. De este modo trataremos de representar los objetos de $\mathcal{T}$ que se pueden construir "sobre" estos datos prefijados. La novedad es que no nos restringiremos a las álgebras para esta signatura que tengan como dominio de datos a D; sino a aquellas que se sustenten en D.

Así, a partir de la signatura Σ con dominio de datos $D = \{D_s\}_{s \in S}$ construimos la signatura con igualdad Σ^{eq}. El dominio de datos que se fija para Σ^{eq} es el dominio de datos que se tiene para Σ junto con el conjunto $D_{bool} = \{true, false\}$. Este nuevo dominio de datos lo seguiremos denotando D (podemos suponer que en la signatura Σ no tenemos un género llamado *bool* con dominio de datos distinto de los booleanos; si así ocurriese bastaría con renombrar este género).

Además, para cada operación $\omega\colon s_1 \dots s_n \to s$ de Ω^{eq}, $n \geq 0$, prefijamos también un conjunto $dom_\omega \subseteq D_{s_1} \times \cdots \times D_{s_n}$, que se llama *dominio de definición* para ω. El conjunto de dominios de definición para las operaciones de una signatura $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$ se llama *dominio de definición* para Σ^{eq} y lo denotamos por $dom^{\Sigma^{eq}}$, es decir, $dom^{\Sigma^{eq}} = \{dom_\omega\}_{\omega \in \Omega^{eq}}$. Denotaremos también este conjunto por dom^{Σ} o bien simplemente por dom si no hay lugar a confusión.

Como resulta natural suponer, el dominio de definición de una signatura va a determinar los dominios de las funciones parciales para sus álgebras. La necesidad de fijar estos dominios de definición viene motivada también por lo que ocurre en el nivel de las implementaciones. Cuando se trabaja en este nivel, para cada operación de la signatura sólo se tiene un código (que implementa esta operación) y un dominio de datos sobre el que el código es sintácticamente correcto; pero no se tiene el dominio real de definición, como ocurre al trabajar con funciones (una función parcial tiene, según su propia definición, incorporado un dominio de definición). Sin embargo, un mismo código puede tener distintos dominios de definición, por lo que en este caso se hace necesario fijar uno de ellos. Como veremos algo más adelante para trabajar en el nivel algebraico no es estrictamente necesario fijar estos dominios de definición. En un álgebra, cada operación viene acompañada de su dominio de definición por lo que está perfectamente definida. No obstante, preferimos comenzar el estudio aproximándonos lo más posible al modelo usado en EAT para la implementación de algunos tipos de estructuras matemáticas y para ello utilizaremos ejemplos directamente obtenidos de dicho programa, tal y como se hizo en [28].

Llegado este punto, tenemos que dada una signatura Σ^{eq} con domino de datos D y dominio de definición *dom* asociados, la categoría ambiente en la que trabajaremos es la categoría $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$ de Σ^{eq}-álgebras parciales con igualdad con conjuntos soportes

D y dominios de definición *dom*. Es decir, dada una signatura $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$, una Σ^{eq}-álgebra A de $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$ verifica que $A_s = D_s$ para cada género $s \in S^{eq}$, en particular $A_{bool} = \{true, false\}$, y para cada operación $\omega\colon s_1 \ldots s_n \to s \in \Omega^{eq}$, $n \geq 0$, w_A es una función parcial con dominio de definición $Def(\omega_A) = dom_\omega$. Además, las relaciones que se obtienen de las funciones parciales que se asocian a las operaciones de igualdad determinan una relación de congruencia parcial.

Los morfismos de la categoría $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$ son los Σ^{eq}-homomorfismos débiles identidad. Nuevamente, los géneros de esta signatura representarán los géneros visibles de un álgebra oculta. Es por ello por lo que las únicas relaciones entre los conjuntos soportes de estas álgebras que vamos a permitir son las identidades. Una mejor notación para este tipo de categorías discretas es posiblemente $P\tilde{A}lg^{D,dom,\{\}}(\Sigma^{eq})$, siguiendo la notación utilizada en [72], de modo que se hace referencia visual explícita a que estamos trabajando con una categoría discreta. Nosotros preferimos no utilizar los símbolos $\{\}$ por no recargar demasiado la notación, insistiendo, cuando sea preciso, que estamos trabajando con una categoría sin morfismos distintos de las identidades.

En general, no nos van a interesar todas las álgebras de la categoría $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$, sino un subcategoría $\mathcal{C}$ que defina un TAD, esto es, que sea cerrada por isomorfismo. En realidad, como los únicos morfismos que tenemos son las identidades, cualquier subcategoría verifica esta condición. Como hemos indicado, en lugar de utilizar un enfoque axiomático, para determinar dichas álgebras utilizaremos una serie de restricciones semánticas, que permitan definir la subcategoría $\mathcal{C}$. Así, una primera restricción semántica es que las álgebras de la categoría verifiquen que las funciones que interpretan las igualdades constituyan una relación de congruencia. En general, deberemos exigir que el álgebra cociente que se define a través de dicha congruencia sea un objeto de $\mathcal{T}$, que es la clase que inicialmente pretendemos representar.

De esta forma, en lugar de buscar, como hacíamos en los capítulos previos, Σ-álgebras totales que representen a los elementos de la clase de objetos $\mathcal{T}$ que se pueden construir sobre un dominio de datos prefijado, utilizaremos Σ^{eq}-álgebras parciales con igualdad A (eso sí con un dominio de datos prefijado D y un dominio de definición para sus operaciones también prefijado *dom*), de modo que el álgebra parcial cociente $\tilde{A}$ que se obtiene a partir de la relación de congruencia parcial definida por las funciones parciales correspondientes a las operaciones de igualdad, represente a un objeto de $\mathcal{T}$.

Debemos observar que la categoría $\mathcal{C}$ que obtenemos supone una generalización respecto a la que definimos en los capítulos previos. Para ello simplemente debemos considerar todas las operaciones totales y las operaciones de igualdad como las igualdades literales en el dominio de datos.

4.4.1 Ejemplos

En esta sección aplicaremos la técnica que hemos desarrollado anteriormente para el modelado de estructuras concretas. En particular especificaremos los grupos finitos

$\mathbb{Z}/n\mathbb{Z}$, los conjuntos simpliciales que pueden construirse sobre un conjunto graduado X de símplices geométricos y los complejos de cadenas que pueden construirse sobre un conjunto graduado X de generadores.

4.4.1.1 Grupos sobre $\mathbb{Z}$

Si tratamos de especificar la familia de grupos finitos $\mathbb{Z}/n\mathbb{Z}$ con $n > 1$ no podemos utilizar la técnica que hemos definido en los capítulos anteriores dentro de la especificación algebraica ecuacional con dominio de datos prefijado. El motivo es que cada grupo de la familia está definido sobre un conjunto de elementos distinto. No obstante, todos esos conjuntos de datos presentan la propiedad de que se pueden definir sobre un soporte común $D = \mathbb{Z}$. Para conseguir cada conjunto particular en cada grupo es suficiente con establecer relaciones de equivalencia en el mismo. Este soporte base está representado en el nivel de implementación por el tipo de datos `integer`.

Realizaremos a continuación la especificación de los grupos que se pueden definir estableciendo identificaciones dentro del conjunto de los números enteros. Estos grupos integran una subcategoría plena de la categoría de los grupos, que denotamos por $\mathcal{T}_{GRP_{\mathbb{Z}}}$, y engloban a los grupos finitos $\mathbb{Z}/n\mathbb{Z}$ con $n > 1$.

La signatura que nos sirve como base sintáctica es $\mathtt{GRP}^{eq}$. Recordamos que esta signatura tiene como conjunto de géneros $\{g, bool\}$ y como conjunto de operaciones $\{prd\colon g\ g \to g, inv\colon g \to g, unt\colon \ \to g, eq^g\colon g\ g \to bool\}$.

El dominio de datos que fijamos para los géneros es $D_g = \mathbb{Z}$ y $D_{bool} = \{true, false\}$. En este ejemplo el dominio de definición *dom* para las operaciones consiste en considerar todas ellas totales. Trabajamos entonces con la categoría de álgebras (totales) con igualdad $\tilde{A}lg^D(\mathtt{GRP}^{eq})$.

Entonces, nos restringimos a la subcategoría $\mathcal{C}^{GRP_{\mathbb{Z}}}$ de $\tilde{A}lg^D(\mathtt{GRP}^{eq})$ tal que cada $\mathtt{GRP}^{eq}$-álgebra A objeto de $\mathcal{C}^{GRP_{\mathbb{Z}}}$ verifica que:

- la función $eq^g_A\colon \mathbb{Z} \times \mathbb{Z} \to \{true, false\}$ junto con la igualdad literal en el género *bool* determinan una relación de congruencia en A,
- la $\mathtt{GRP}$-álgebra cociente $\tilde{A}$ que se obtienen por la relación de congruencia anterior a partir de A verifica que, para cada $[a], [b], [c] \in \tilde{A}_g$,

  · $prd_{\tilde{A}}([a], prd_{\tilde{A}}([b], [c])) = prd_{\tilde{A}}(prd_{\tilde{A}}([a], [b]), [c])$,
  · $prd_{\tilde{A}}([a], unt_{\tilde{A}}) = prd_{\tilde{A}}(unt_{\tilde{A}}, [a]) = [a]$,
  · $prd_{\tilde{A}}([a], inv_{\tilde{A}}([a])) = prd_{\tilde{A}}(inv_{\tilde{A}}([a]), [a]) = unt_{\tilde{A}}$.

  Es decir, $\tilde{A}$ es un grupo.

De este modo, cualquier grupo finito o numerable puede ser modelado sobre $\mathcal{C}^{GRP_{\mathbb{Z}}}$ (y esto cubre cualquier grupo interesante tratable desde el punto de vista del Cálculo Simbólico).

4.4.1.2 Conjuntos simpliciales

Un *conjunto simplicial* K es un conjunto graduado indexado por los enteros no negativos, $K = \{K^n\}_{n\in\mathbb{N}}$, junto con dos familias de funciones

- $\delta_i^n \colon K^n \to K^{n-1}$, para $n > 0$ e $i = 0, \ldots, n$
- $\eta_i^n \colon K^n \to K^{n+1}$, para $n \geq 0$ e $i = 0, \ldots, n$

que satisfacen las siguientes igualdades:

$$
\begin{array}{rclll}
\delta_i^{n-1}\delta_j^n & = & \delta_{j-1}^{n-1}\delta_i^n & \text{si} & i < j, \\
\eta_i^{n+1}\eta_j^n & = & \eta_{j+1}^{n+1}\eta_i^n & \text{si} & i \leq j, \\
\delta_i^{n+1}\eta_j^n & = & \eta_{j-1}^{n-1}\delta_i^n & \text{si} & i < j, \\
\delta_j^{n+1}\eta_j^n & = & \text{identidad} & = & \delta_{j+1}^{n+1}\eta_j^n, \\
\delta_i^{n+1}\eta_j^n & = & \eta_j^{n-1}\delta_{i-1}^n & \text{si} & i > j+1.
\end{array}
$$

A los elementos de K^n se les denomina *símplices de dimensión* n o n*-símplices* de K. A las funciones δ_i^n se les denomina *operadores cara* (un n-símplice tiene $n+1$ caras que son símplices de dimensión $n-1$) y a las η_i^n *operadores degeneración.* Un símplice que se encuentra en la imagen de un operador degeneración se denomina *símplice degenerado* y, en otro caso, *símplice geométrico* o *no degenerado.*

Si K y L son conjuntos simpliciales, un morfismo $f \colon K \to L$ entre ellos es una familia de aplicaciones, $f = \{f^n \colon K^n \to L^n\}_{n\in\mathbb{N}}$, entre los conjuntos correspondientes al mismo grado que conmutan con los operadores cara y degeneración.

Los conjuntos simpliciales junto con los morfismos entre ellos forman una categoría que denominamos categoría de conjuntos simpliciales y que denotamos por $\mathcal{T}_{SS}$.

Referencias en las que podemos encontrar las definiciones anteriores, así como un estudio más detallado de los conjuntos simpliciales son los libros [65], [64].

Nuestro objetivo es modelar la clase de objetos de $\mathcal{T}_{SS}$ y para ello, vamos a definir un TAD que la aproxime. En [57] aparece tratado este ejemplo, y en [72] se desarrolla con detalle. En ambos casos se trabaja con álgebras parciales; ahora aprovechamos además la posibilidad de definir cocientes en el dominio de datos.

Una propiedad que presentan los símplices de un conjunto simplicial y que juega un papel importante en el modo en el que se representan en el sistema EAT es la siguiente.

Propiedad 4.4.1. Cualquier símplice de un conjunto simplicial puede expresarse de forma única como una sucesión de degeneraciones de un símplice no degenerado; es decir, cada n-símplice x admite una expresión de la forma $x = \eta_{j_k}^{n-1} \ldots \eta_{j_1}^{n-k} z$ donde $k \geq 0$, $j_1 < \cdots < j_k$ y z es un $(n-k)$-símplice no degenerado. Si $k = 0$ entonces x es un n-símplice no degenerado.

Debido a la propiedad anterior, es habitual, al trabajar con conjuntos simpliciales, partir de un conjunto graduado $K = \{K^n\}_{n \in \mathbb{N}}$ que no contenga a todos los símplices, sino solamente a los símplices geométricos.

Llamaremos *símplice abstracto* a la representación de un símplice mediante un par formado por una secuencia de degeneraciones aplicables consecutivamente y por un símplice no degenerado. La propiedad anterior nos dice que todo símplice posee representación como símplice abstracto. En la práctica, para representar la secuencia de degeneraciones de un símplice abstracto es suficiente con tener una lista de números que contengan los subíndices de dichas degeneraciones junto con el símplice no degenerado, ya que el superíndice de las mismas viene determinado por la dimensión de este símplice. Por lo tanto, podemos representar un símplice abstracto por un par formado por una lista de enteros no negativos estrictamente decreciente y por un símplice geométrico. La representación de los símplices geométricos tendrá la lista vacía como lista de degeneraciones.

Entonces, si partimos del conjunto de símplices geométricos, tenemos un modo de representación común, un patrón para representar los símplices como símplices abstractos. Por lo tanto, los conjuntos simpliciales son buenos candidatos a ser modelados mediante nuestra técnica de especificación. Nos apoyaremos en este patrón para elegir un dominio de datos para sus elementos.

La signatura Σ^{ss} que definimos a continuación nos va a permitir especificar los conjuntos simpliciales.

signatura Σ^{ss}
  géneros *nat*, *gsm*, *asm*
  operaciones
    $face\colon nat\ nat\ asm \to asm$
    $dgn\colon nat\ nat\ asm \to asm$
finsig

El género *nat* representa al conjunto de los números naturales $\mathbb{N}$, *gsm* a los símplices geométricos y *asm* a los símplices del conjunto simplicial. Las operaciones $face$ y dgn corresponden a los operadores cara y degeneración en el conjunto simplicial, respectivamente.

En este momento son necesarios una serie de comentarios respecto a la signatura anterior. Existe una diferencia fundamental entre el ejemplo de grupo y el ejemplo de complejo simplicial. El primero está compuesto por un único conjunto y por una serie de operaciones sobre este conjunto. Dicho ejemplo se adapta perfectamente a la especificación algebraica clásica. Sin embargo, un conjunto simplicial está formado por un número infinito de conjuntos y por un número también infinito de aplicaciones entre estos conjuntos. Una primera forma de abordar este segundo ejemplo, puede ser el considerar una signatura con un número infinito de géneros y de operaciones totales que representen a cada uno de estos conjuntos y aplicaciones. No obstante, si pensamos en una posible implementación directa de una tal signatura, con una función por cada operación, obtendríamos tuplas infinitas de funciones.

Una segunda opción es considerar la signatura que hemos definido. En ella tenemos un único género que recoge a todos los símplices geométricos en un solo conjunto. Desde un punto de vista matemático, esto significa que estamos más bien representado la unión disjunta de todos los símplices geométricos $\bigsqcup_{n\in\mathbb{N}} K^n$ en lugar del conjunto graduado $\{K^n\}_{n\in\mathbb{N}}$. Esto exige poder determinar qué elemento pertenece a qué conjunto dentro del conjunto graduado. Para ello, podemos representar, por ejemplo, los elementos de $\bigsqcup_{n\in\mathbb{N}} K^n$, a través de parejas $\langle n, x\rangle$ con $n \in \mathbb{N}$ y $x \in K^n$. En EAT los símplices geométricos no tienen esta representación, fundamentalmente por razones de eficiencia. Se realizan muchísimas operaciones sobre estos símplices geométricos y por ello se ha preferido una representación de los mismos más simple a través de un único elemento x que establezca el valor del mismo, sin tener una información explícita del grado al cual pertenece este elemento. Esta decisión implica que el usuario, y también el programa, debe conocer en cada momento cuál es el grado de un elemento. Esto lo podemos representar a través de una función con codominio los naturales, $dim\colon K \to \mathbb{N}$, que permita extraer la dimensión de cada símplice geométrico. La representación del resto de los símplices se realiza a través de su correspondiente expresión como símplice abstracto según la propiedad anterior. Así, el género gsm sin operaciones se utiliza como género auxiliar para poder definir el soporte del género asm. Este género en principio no es necesario para modelar un conjunto simplicial, pero a través de él ponemos de manifiesto, como hace EAT, la distinción entre símplices geométricos y símplices degenerados.

Ahora, gracias a la presencia de parcialidad, podemos recoger en dos únicas operaciones, $face$ y dgn, a todos los operadores cara y degeneración. Estas operaciones contienen como argumentos el índice y la dimensión de la cara o degeneración a calcular, así como un símplice. La idea es que solamente estarán definidas sobre aquellos símplices cuya dimensión coincida con la dimensión de la operación y el índice de la misma esté entre los permitidos.

Si retomamos nuestra técnica de especificación tenemos que no vamos a poder representar todos los conjuntos simpliciales. Únicamente aquéllos que se puedan construir sobre un conjunto de símplices geométricos prefijados $\cup_{n\in\mathbb{N}} K^n$, utilizando equivalencias o parcialidad en ellos, van a poder ser modelados.

Consideramos entonces fijado el siguiente dominio de datos D^{ss} para la signatura anterior:

- $D^{ss}_{nat} := \mathbb{N}$,
- $D^{ss}_{gsm} := \cup_{n\in\mathbb{N}} K^n$,
- $D^{ss}_{asm} := \{\langle (j_k, \ldots, j_1), a\rangle \mid \exists n \in \mathbb{N}$ tal que $a \in K^n$, $k \in \mathbb{N}$, $j_i \in \mathbb{N}$, $\forall i = 1, \ldots, k$ y $0 \le j_1 < \cdots < j_k \le n + k - 1\}$.

Además, asociado al conjunto de datos D^{ss}_{gsm} debe existir una operación $dim^{ss}_{gsm}\colon D^{ss}_{gsm} \to \mathbb{N}$ que permita determinar dado un elemento $a \in D^{ss}_{gsm}$ la *dimensión* o *grado* de a, es decir, el conjunto K^n tal que $a \in K^n$. Esta operación induce otra $dim^{ss}_{asm}\colon D^{ss}_{asm} \to \mathbb{N}$ que permite determinar la dimensión de un símplice abstracto. Esta aplicación, se define de la siguiente manera: para cada símplice abstracto $x = \langle (j_k, \ldots, j_1), a\rangle \in D^{ss}_{asm}$, $dim^{ss}_{asm}(x) := n + k$, si $dim^{ss}_{gsm}(a) = n$. Denotaremos de la misma forma estas dos operaciones cuando no exista riesgo de confusión.

El conjunto de datos D^{ss}_{asm} corresponde con la representación en EAT para los símplices geométricos y depende del conjunto D^{ss}_{gsm}. En realidad, este último será el único conjunto de datos que variaremos de una categoría de álgebras a otra. Es más, el conjunto D^{ss}_{asm} podemos obtenerlo como modelo inicial para la categoría de álgebras asociada a una signatura auxiliar. Esta signatura está formada por las operaciones constantes para los géneros *nat* y *gsm*, una por cada elemento del dominio de datos en estos géneros, junto con la operación $dgn\colon nat\ nat\ asm \to asm$ y una operación de coerción *convert-n-d-gsm*: $gsm \to asm$ (presente en EAT, ver [81], página 41) que transforma un símplice geométrico en el correspondiente símplice abstracto no degenerado (debemos exigir a las álgebras de la categoría que las operaciones anteriores respeten la dimensión de los elementos y que verifiquen la propiedad que permite intercambiar dos operaciones degeneración cuando éstas actúan seguidas). Las operaciones de coerción jugarán un papel destacado en el próximo capítulo.

Construimos entonces la signatura con igualdad $\Sigma^{ss,eq}$ a partir de Σ^{ss}. Esta signatura es:

signatura $\Sigma^{ss,eq}$
  géneros *nat*, *gsm*, *asm*, *bool*
  operaciones
    $face\colon nat\ nat\ asm \to asm$
    $dgn\colon nat\ nat\ asm \to asm$
    $eq^{nat}\colon nat\ nat \to bool$
    $eq^{gsm}\colon gsm\ gsm \to bool$
    $eq^{asm}\colon asm\ asm \to bool$
finsig

Esta signatura contiene tres operaciones de igualdad, una por cada género de Σ^{ss}. Mientras que la operación de igualdad sobre los naturales es una operación total (en realidad exigiremos que sea la igualdad literal), las igualdades sobre los símplices geométricos y símplices abstractos serán operaciones parciales: únicamente interesa comparar dos elementos si tienen la misma dimensión. Entonces, podemos prefijar el siguiente dominio de definición dom^{ss} para las operaciones de la signatura, apoyándonos en las operaciones de dimensión:

- $dom^{ss}_{face} = \{(i,n,x) \in \mathbb{N} \times \mathbb{N} \times D^{ss}_{asm} \mid 0 \le i \le n,\ n > 0,\ dim^{ss}(x) = n\}$,
- $dom^{ss}_{dgn} = \{(i,n,x) \in \mathbb{N} \times \mathbb{N} \times D^{ss}_{asm} \mid 0 \le i \le n,\ n \ge 0,\ dim^{ss}(x) = n\}$,
- $dom^{ss}_{eq^{gsm}} = \{(a,b) \in D^{ss}_{gsm} \times D^{ss}_{gsm} \mid dim^{ss}(a) = dim^{ss}(b)\}$,
- $dom^{ss}_{eq^{asm}} = \{(x,y) \in D^{ss}_{asm} \times D^{ss}_{asm} \mid dim^{ss}(x) = dim^{ss}(y)\}$,

· la operación eq^{nat} es total.

De esta forma tenemos determinado el ambiente en el que situamos nuestras álgebras que es $P\tilde{A}lg^{D^{ss},dom^{ss}}(\Sigma^{ss,eq})$. No obstante, no nos interesan todas las $\Sigma^{ss,eq}$-álgebras de $P\tilde{A}lg^{D^{ss},dom^{ss}}(\Sigma^{ss,eq})$ sólo nos quedaremos con aquéllas que verifiquen que las operaciones de igualdad constituyan una relación de congruencia parcial particular (que sea coherente con el dominio de datos que hemos elegido para representar a los símplices geométricos) y que el álgebra parcial cociente que se obtiene a partir de esta congruencia sea una conjunto simplicial. Es decir, nos restringiremos a una subcategoría $\mathcal{C}^{ss}$ de $P\tilde{A}lg^{D^{ss},dom^{ss}}(\Sigma^{ss,eq})$ tal que cada A objeto de $\mathcal{C}^{ss}$ verifica las siguientes condiciones:

· La función parcial $eq_A^{asm}\colon D_{asm}^{ss} \times D_{asm}^{ss} \to \{true, false\}$, se define de modo natural a partir de la relación de equivalencia parcial definida para los símplices geométricos, es decir, para cada $(\langle(j_k,\ldots,j_1),a\rangle, \langle(l_p,\ldots,l_1),b\rangle) \in dom_{eq^{asm}}^{ss}$, tenemos que $eq_A^{asm}(\langle(j_k,\ldots,j_1),a\rangle, \langle(l_p,\ldots,l_1),b\rangle) = true$ si y solamente si $k = p$, $j_i = l_i$ $\forall i = 1,\ldots,k$ y $eq_A^{gsm}(a,b) = true$.

· La función $eq_A^{nat}\colon \mathbb{N} \times \mathbb{N} \to \{true, false\}$ define la igualdad literal en $\mathbb{N}$, es decir, no identifica parejas de naturales no iguales.

Además, debemos exigir que la Σ^{ss}-álgebra parcial cociente $\tilde{A}$ que se obtiene a partir de la relación de congruencia parcial anterior verifique las igualdades exigidas a los conjuntos simpliciales. Si tenemos en cuenta que la dimensión de un elemento coincide con la de toda su clase de equivalencia, estas igualdades pueden describirse del siguiente modo:

· para cada $[x] \in \tilde{A}_{asm}$ con $dim^{ss}(x) = n$ y cada par $i,j \in \mathbb{N}$ tal que $i,j \leq n$ e $i < j$, se tiene que

$$face_{\tilde{A}}(i, n-1, face_{\tilde{A}}(j, n, [x])) = face_{\tilde{A}}(j-1, n-1, face_{\tilde{A}}(i, n, [x]))$$

· para cada $[x] \in \tilde{A}_{asm}$ con $dim^{ss}(x) = n$ y cada par $i,j \in \mathbb{N}$ tal que $i,j \leq n$ e $i \leq j$, se tiene que

$$dgn_{\tilde{A}}(i, n+1, dgn_{\tilde{A}}(j, n, [x])) = dgn_{\tilde{A}}(j+1, n+1, dgn_{\tilde{A}}(i, n, [x]))$$

· para cada $[x] \in \tilde{A}_{asm}$ con $dim^{ss}(x) = n$ y cada par $i,j \in \mathbb{N}$ tal que $i,j \leq n$ e $i < j$, se tiene que

$$face_{\tilde{A}}(i, n+1, dgn_{\tilde{A}}(j, n, [x])) = dgn_{\tilde{A}}(j-1, n-1, face_{\tilde{A}}(i, n, [x]))$$

· para cada $[x] \in \tilde{A}_{asm}$ con $dim^{ss}(x) = n$ y cada $j \in \mathbb{N}$ tal que $j \leq n$, se tiene que

$$face_{\tilde{A}}(j, n+1, dgn_{\tilde{A}}(j, n, [x])) = face_{\tilde{A}}(j+1, n+1, dgn_{\tilde{A}}(j, n, [x])) = [x]$$

· para cada $[x] \in \tilde{A}_{asm}$ con $dim^{ss}(x) = n$ y cada par $i, j \in \mathbb{N}$ tal que $i \leq n+1$ e $i > j+1$, se tiene que

$$face_{\tilde{A}}(i, n+1, dgn_{\tilde{A}}(j, n, [x])) = dgn_{\tilde{A}}(j, n-1, face_{\tilde{A}}(i-1, n, [x])).$$

Las anteriores igualdades llevan implícita la condición de que las operaciones $face_A$ y dgn_A sean compatibles con la operación dimensión, es decir:

· para cada $(i, n, x) \in dom^{ss}_{face}$, $dim^{ss}(face_A(i, n, x)) = n - 1$,

· para cada $(i, n, x) \in dom^{ss}_{dgn}$, $dim^{ss}(dgn_A(i, n, x)) = n + 1$.

Por último, para ser coherentes con la representación que hemos elegido para los símplices abstractos, debemos exigir la siguiente condición a la operación degeneración: para cada $(i, n, \langle (j_k, \ldots, j_1), a \rangle) \in dom^{ss}_{dgn}$ se verifica que:

$$dgn_A(i, n, \langle (j_k, \ldots, j_1), a \rangle) := \langle (j_k + 1, \ldots, j_l + 1, i, j_{l-1}, \ldots, j_1), a \rangle$$

con l tal que $j_{l-1} < i \leq j_l$.

La propiedad que permite intercambiar las operaciones degeneración, junto con la interpretación de la lista estrictamente decreciente de naturales como la sucesiva aplicación de las operaciones degeneración, es la que posibilita la anterior definición. Tenemos entonces una lista estrictamente decreciente, ya que en el caso de que la lista contenga una pareja (i, j) con $i \leq j$, entonces podemos sustituirla por $(j+1, i)$.

Obtenemos de esta forma a través de la categoría $\mathcal{C}^{ss}$ una representación de cualquier conjunto simplicial que se pueda construir como cociente del conjunto de símplices geométricos $K = \{K^n\}_{n \in \mathbb{N}}$.

4.4.1.3 Complejos de cadenas

Un *complejo de cadenas* $C = \{C_p, d_p\}_{p \in \mathbb{Z}}$ es una familia de $\mathbb{Z}$-módulos libres $\{C_p\}_{p \in \mathbb{Z}}$ junto con una familia de homomorfismos de $\mathbb{Z}$-módulos $\{d_p\}_{p \in \mathbb{Z}}$, $d_p \colon C_p \to C_{p-1}$, llamadas *aplicaciones diferenciales*, tal que $d_{p-1} \circ d_p = 0$, para cada $p \in \mathbb{Z}$ (habitualmente esta propiedad se indica diciendo que $d^2 = 0$).

Si $C = \{C_p, d_p\}_{p \in \mathbb{Z}}$ y $C' = \{C'_p, d'_p\}_{p \in \mathbb{Z}}$ son complejos de cadenas, un *morfismo* $f \colon C \to C'$ entre ellos, es una familia de homomorfismos de módulos, $f = \{f_p \colon C_p \to C'_p\}_{p \in \mathbb{Z}}$, tales que $f_{p-1} \circ d_p = d'_p \circ f_p$, para todo $p \in \mathbb{Z}$.

Un $\mathbb{Z}$-módulo libre es un grupo abeliano que, además es generado, lo que implica que cada elemento del mismo puede expresarse como una combinación lineal sobre un conjunto de generadores. A estos elementos se les denomina por ello *combinaciones* (los elementos de C_p se les llamarán *combinaciones de grado* p). Es importante notar que la definición anterior es sólo un caso particular de la definición de complejo de cadenas que se puede encontrar, por ejemplo, en [64]. En realidad, un complejo de cadenas se define como una familia de R-módulos, con R un anillo cualquiera. No obstante, consideraremos la anterior definición de complejo de cadenas por ser ésta la que utiliza EAT (ver [81], página 1).

Los complejos de cadenas junto con los morfismos entre ellos forman una categoría que denominamos categoría de complejos de cadenas y que denotamos por $\mathcal{T}_{CC}$.

Nuestro objetivo es modelar la clase de objetos de $\mathcal{T}_{CC}$ y para ello vamos a definir un TAD que la aproxime. Nuevamente utilizamos la terminología y las construcciones que utiliza el sistema EAT [81], apoyándonos en el ejemplo desarrollado en [72].

La estructura de complejo de cadenas es similar a la de un conjunto simplicial en cuanto a que su base es un conjunto graduado (en particular un grupo abeliano libre graduado). Debemos observar que, a partir del conjunto de generadores G_p de un $\mathbb{Z}$-módulo libre C_p en un determinado grado p, una combinación de ese grado es una suma formal de *monomios* constituidos por un entero que actúa por multiplicación sobre un generador. En la práctica, para representar una tal combinación es suficiente con tener un conjunto de pares formados por un entero y un generador, junto con el grado de la combinación que puede venir expresado por otro entero, de modo que los generadores de todos los pares deban ser de dicho grado. De esta forma, si partimos del conjunto de generadores, tenemos un patrón común para las combinaciones como pares formados por un entero, que marque el grado de la combinación, y una lista de pares formados por un entero y un generador. Debemos observar que, en esta ocasión, no obtenemos una representación directa de las combinaciones, ya que el orden de las parejas en la lista no debe determinar combinaciones distintas. No obstante, dispondremos posteriormente de una igualdad sobre estos elementos. Esta igualdad será utilizada para identificar dos de estos elementos si se diferencian en el orden de estos pares en la lista.

Entonces, para modelar esta graduación utilizaremos una técnica similar a la utilizada para los conjuntos simpliciales. Consideraremos dos géneros *gnr* y *cmb* en los que se recogerán respectivamente los generadores y las combinaciones de todos los grupos, junto con otro género *int*, que representará al conjunto de los números enteros $\mathbb{Z}$. Es necesario incluir alguna forma de obtener el grado de un generador o de una combinación. Una posible forma de conseguir esto es, como ya comentamos en el caso de los conjuntos simpliciales, reflejar explícitamente en la expresión de los elementos un índice que marque la graduación. EAT codifica de esta forma las combinaciones. Sin embargo, no añade, otra vez por razones de eficiencia, a cada generador el grado que le corresponde. No obstante, nuevamente supone que el usuario, y también el programa, sabe en todo momento el grado de un generador. De modo análogo a lo visto para conjuntos simpliciales, utilizaremos para representar esto una operación que obtenga el grado para cada uno de los generadores.

Una signatura que va a formar parte de nuestra especificación de los complejos de cadenas es Σ^{cc}, que se define por:

signatura Σ^{cc}
 géneros *int*, *gnr*, *cmb*
 operaciones
  $add\colon cmb\ cmb \to cmb$
  $zero\colon int \to cmb$
  $mns\colon cmb \to cmb$
  $mlt\colon int\ cmb \to cmb$
  $df\colon cmb \to cmb$
finsig

Las cuatro primeras operaciones corresponden a la estructura de $\mathbb{Z}$-módulo graduado sobre *cmb*, y la operación *df* a las aplicaciones diferenciales.

Como es de esperar, es necesario prefijar un conjunto de generadores que servirá de base, las combinaciones se obtendrán a partir de ellos. Para ello fijamos el siguiente dominio de datos:

- $D^{cc}_{int} := \mathbb{Z}$,
- $D^{cc}_{gnr} := \cup_{p\in\mathbb{Z}} G_p$,
- $D^{cc}_{cmb} := \{\langle p, [(t_1, a_1), (t_2, a_2), \ldots, (t_m, a_m)]\rangle \mid p \in \mathbb{Z},\ m \in \mathbb{N},\ t_i \in \mathbb{Z},\ \text{y } a_i \in G_p, \forall i = 1, \ldots, m\}$.

El conjunto de datos D^{cc}_{gnr} debe disponer de una operación $dim^{cc}_{gnr}\colon D^{cc}_{gnr} \to \mathbb{Z}$ que permita determinar, dado un elemento $a \in D^{cc}_{gnr}$, la *dimensión* o *grado* de a, es decir, el conjunto G_n tal que $a \in G_n$. En esta ocasión, por la representación elegida para las combinaciones, obtenemos directamente una operación $dim^{cc}_{cmb}\colon D^{cc}_{cmb} \to \mathbb{Z}$ que permite extraer el grado de una combinación: $dim^{cc}_{cmb}(\langle p, [(t_1, a_1), (t_2, a_2), \ldots, (t_m, a_m)]\rangle) := p$. Denotaremos de la misma forma estas dos operaciones cuando no exista riesgo de confusión.

Nuevamente, el conjunto de datos D^{cc}_{cmb} corresponde con la representación que se ha utilizado en EAT para las combinaciones y depende del conjunto D^{cc}_{gnr} que es también el único conjunto de datos que variaremos de una categoría de álgebras a otra. Además, el conjunto D^{cc}_{cmb} también puede obtenerse como modelo inicial para una categoría de álgebras asociadas a una signatura auxiliar. Esta signatura está formada por las operaciones constantes para los géneros *int* y *gnr* junto con las operaciones $zero\colon int \to cmb$, que permite construir la combinación nula, y una operación (parcial) $add\text{-}mnm\text{-}to\text{-}cmbn\colon int\ gnr\ cmb \to cmb$ que añade a una combinación un nuevo monomio (se debe además exigir a las álgebras de la categoría que las operaciones anteriores respeten la dimensión de los elementos). En realidad, en el programa EAT se trabaja con combinaciones denominadas *reducidas*, es decir, que todos sus monomios tengan coeficiente no nulo y sus generadores sean todos distintos. No obstante, en el mismo programa se comenta que los resultados de algunos cálculos intermedios pueden originar combinaciones no reducidas. Nosotros, en lugar de tomar como base para las combinaciones las combinaciones reducidas, consideramos también las no reducidas, de modo que nos va a resultar posteriormente más sencillo dar la definición de algunas de las operaciones que podrán devolver combinaciones no reducidas. Posteriormente, recuperaremos las combinaciones reducidas al actuar la operación de igualdad sobre ellas (simulando el proceso realizado en EAT).

Ahora, construimos la signatura $\Sigma^{cc,eq}$ a partir de Σ^{cc}. Esta signatura es:

signatura $\Sigma^{cc,eq}$
géneros *int*, *gnr*, *cmb*, *bool*
operaciones
$add\colon cmb\ cmb \to cmb$
$zero\colon int \to cmb$
$mns\colon cmb \to cmb$
$mlt\colon int\ cmb \to cmb$
$df\colon cmb \to cmb$
$eq^{int}\colon int\ int \to bool$
$eq^{gnr}\colon gnr\ gnr \to bool$
$eq^{cmb}\colon cmb\ cmb \to bool$
finsig

Nuevamente nos apoyaremos en la parcialidad para conseguir una estructura graduada a partir de estas operaciones: únicamente estarán definidas si actúan sobre elementos de grado correcto. A partir de las anteriores operaciones de dimensión podemos definir el siguiente dominio de definición dom^{cc} para las operaciones de la signatura:

- $dom^{cc}_{add} = \{(x,y) \in D^{cc}_{cmb} \times D^{cc}_{cmb} \mid dim^{cc}(x) = dim^{cc}(y)\}$,
- las operaciones $zero$, mns, mlt, df y eq^{int} son totales,
- $dom^{cc}_{eq^{gnr}} = \{(a,b) \in D^{cc}_{gnr} \times D^{cc}_{gnr} \mid dim^{cc}(a) = dim^{cc}(b)\}$,
- $dom^{cc}_{eq^{cmb}} = \{(x,y) \in D^{cc}_{cmb} \times D^{cc}_{cmb} \mid dim^{cc}(x) = dim^{cc}(y)\}$.

De esta forma tenemos definido el ambiente $P\tilde{A}lg^{D^{cc},dom^{cc}}(\Sigma^{cc,eq})$ en el que situamos nuestras álgebras. Entonces nos restringiremos a una subcategoría $\mathcal{C}^{cc}$ de éstas, compuesta por aquéllas que verifiquen que su cociente por la relación de congruencia definida a través de las operaciones de igualdad sea un complejo de cadenas.

Como hemos hecho en el caso de los conjuntos simpliciales no vamos a incluir cualquier relación de congruencia, ya que, si tenemos en cuenta el dominio de datos que hemos fijado, es natural elegir como equivalencia para los enteros la igualdad literal y para las combinaciones la sólo identifique combinaciones iguales salvo generadores equivalentes. Además, vamos a utilizar esta igualdad sobre combinaciones para dotar de significado, como suma de monomios, a la expresión sintáctica que representa una combinación en D^{cc}_{cmb}. Para ello, identificaremos dos combinaciones si se diferencian en el orden de sus monomios; una combinación que presente un monomio con coeficiente nulo con aquélla que no presente dicho monomio; y por último, una combinación que tenga dos generadores iguales en dos monomios distintos con la combinación que presente un único monomio, con ese mismo generador y coeficiente la suma (entera) de los coeficientes.

Así, nos restringiremos a una subcategoría $\mathcal{C}^{cc}$ de $P\tilde{A}lg^{D^{cc},dom^{cc}}(\Sigma^{cc,eq})$ tal que cada $\Sigma^{cc,eq}$-álgebra parcial con igualdad $A \in \mathcal{C}^{cc}$ verifica las siguientes condiciones:

· La función parcial $eq^{cmb}_A\colon D^{cc}_{cmb} \times D^{cc}_{cmb} \to \{true, false\}$ verifica que

- $eq^{cmb}_A(\langle p, [(t_1, a_1), \ldots, (t_m, a_m)]\rangle, \langle p, [(s_1, b_1), \ldots, (s_n, b_n)]\rangle) = true$ si $m = n$, y para cada $i = 1, \ldots, n$ existe un único $j \in \{1, \ldots, n\}$ tal que $t_i = s_j$ y $eq^{gnr}_A(a_i, b_j) = true$,
- $eq^{cmb}_A(\langle p, [(0, a_1), (t_2, a_2), \ldots, (t_m, a_m)]\rangle, \langle p, [(t_2, a_2), \ldots, (t_m, a_m)]\rangle) = true$,
- $eq^{cmb}_A(\langle p, [(t_1, a_1), (t_2, b_2), (t_3, a_3), \ldots, (t_m, a_m)]\rangle, \langle p, [(t_1 + t_2, a_1), (t_3, a_3), \ldots, (t_m, a_m)]\rangle) = true$ si $eq^{gnr}_A(a_1, b_2) = true$.

de modo que la definición de eq^{cmb}_A se completa para constituir una relación de equivalencia parcial en D^{cc}_{cmb}.

· La función $eq^{int}_A\colon \mathbb{Z} \times \mathbb{Z} \to \{true, false\}$ es la igualdad literal en $\mathbb{Z}$.

Además, $\tilde{A}$ debe ser un complejo de cadenas. No obstante, para ser coherentes con la estructura que se ha dado a las combinaciones, tenemos que algunas de las operaciones de grupo abeliano libre del complejo de cadenas han quedado totalmente determinadas. Estas operaciones son las siguientes:

· $zero_A\colon \mathbb{Z} \to D^{cc}_{cmb}$ es la función total definida por:

$$zero_A(p) := \langle p, [()]\rangle,$$

para cada $p \in \mathbb{Z}$,

· $mns_A\colon D^{cc}_{cmb} \to D^{cc}_{cmb}$ es la función total definida por:

$$mns_A(\langle p, [(t_1, a_1), \ldots, (t_m, a_m)]\rangle) := \langle p, [(-t_1, a_1), \ldots, (-t_m, a_m)]\rangle,$$

para cada $\langle p, [(t_1, a_1), \ldots, (t_m, a_m)]\rangle \in D^{cc}_{cmb}$,

· $mlt_A\colon \mathbb{Z} \times D^{cc}_{cmb} \to D^{cc}_{cmb}$ es la función total definida por:

$$mlt_A(n, \langle p, [(t_1, a_1), \ldots, (t_m, a_m)]\rangle) := \langle p, [(n * t_1, a_1), \ldots, (n * t_m, a_m)]\rangle,$$

para cada $n \in \mathbb{Z}$ y cada $\langle p, [(t_1, a_1), \ldots, (t_m, a_m)]\rangle \in D^{cc}_{cmb}$.

Entonces, para que el álgebra $\tilde{A}$ sea una representación de un $\mathbb{Z}$-módulo libre sobre el conjunto de generadores D^{cc}_{gnr} a través de la representación que se ha elegido para las combinaciones es suficiente que la función parcial $add_A\colon D^{cc}_{cmb} \times D^{cc}_{cmb} \to D^{cc}_{cmb}$ con dominio de definición $\{(x,y) \in D^{cc}_{cmb} \times D^{cc}_{cmb} \mid dim^{cc}(x) = dim^{cc}(y)\}$ esté definida por:

$$add_A(x,y) := \langle p, [(t_1,a_1),\ldots,(t_m,a_m),(s_1,b_1),\ldots,(s_n,b_n)]\rangle,$$

para cada $x = \langle p, [(t_1,a_1),\ldots,(t_m,a_m)]\rangle$, $y = \langle p, [(s_1,b_1),\ldots,(s_n,b_n)]\rangle \in D^{cc}_{cmb}$.

Falta además exigir que la función correspondiente a la diferencial *df* represente a una familia de aplicaciones diferenciales. Para ello, la función $df_{\tilde{A}}$ debe ser compatible con el resto de las operaciones (la restricción a los elementos en cada grado es un homomorfismo de $\mathbb{Z}$-módulos), es decir:

- $dim^{cc}(df_{\tilde{A}}([x])) = dim^{cc}([x]) - 1$, para cada $[x] \in \tilde{A}_{cmb}$,
- $df_{\tilde{A}}(add_{\tilde{A}}([x],[y])) = add_{\tilde{A}}(df_{\tilde{A}}([x]), df_{\tilde{A}}([y]))$, para cada $([x],[y]) \in Def(add_{\tilde{A}})$,
- $df_{\tilde{A}}(zero_{\tilde{A}}(p)) = zero_{\tilde{A}}(p-1)$, para cada $p \in \mathbb{Z}$,
- $df_{\tilde{A}}(mns_{\tilde{A}}([x])) = mns_{\tilde{A}}(df_{\tilde{A}}([x]))$, para cada $[x] \in \tilde{A}_{cmb}$,
- $df_{\tilde{A}}(mlt_{\tilde{A}}(n,[x])) = mlt_{\tilde{A}}(n, df_{\tilde{A}}([x]))$ para cada $n \in \mathbb{Z}$ y cada $[x] \in \tilde{A}_{cmb}$.

La dimensión de una combinación coincide con la de todos los elementos de su clase de equivalencia. Es esta dimensión la que se le asigna a la clase de una combinación en la igualdad anterior.

Por último, debe verificarse la propiedad característica de la familia de homomorfismos de un complejo de cadenas que obliga a que la composición de dos homomorfismos consecutivos sea el elemento neutro, es decir:

$$df_{\tilde{A}}(df_{\tilde{A}}([x])) = zero_{\tilde{A}}(p-2),$$

para cada $[x] \in \tilde{A}_{cmb}$, con $dim^{cc}([x]) = p$.

Obtenemos de esta forma, a través de la categoría $\mathcal{C}^{cc}$, una representación de cualquier complejo de cadenas que se pueda construir sobre el conjunto de generadores $G = \{G_p\}_{p\in\mathbb{Z}}$.

4.5 Adaptación de la operación *imp* para incluir la parcialidad y las operaciones de igualdad

En los capítulos previos hemos definido una operación que modela la posibilidad de EAT para trabajar no con una única estructura sino con familias de estructuras de una misma clase. Esta operación se ha denominado operación *imp*. La limitación que nos viene impuesta es que todas las estructuras de la familia deben tener exactamente el mismo conjunto de datos soporte. En la sección anterior hemos considerado signaturas y modelos que incorporan dos características que nos van a permiten ampliar el tipo de estructuras que podemos especificar, a pesar de continuar trabajando con un conjunto de datos fijado. En concreto trabajamos con funciones parciales para las operaciones y con la posibilidad de definir identificaciones dentro del dominio de datos. En esta sección estudiaremos cuáles son las modificaciones que tenemos que realizar en la operación *imp* para poder recoger familias de estructuras que presenten las anteriores características.

Comenzamos comentando que en el nivel sintáctico la operación *imp* no necesita ninguna variación. Las operaciones de igualdad y el género *bool* presentes en una signatura Σ^{eq} no tienen ningún trato diferenciado con respecto del resto de operaciones y de géneros. Así, dada una signatura $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$ con domino de datos D y dominio de definición *dom* asociados, construimos una nueva signatura $\Sigma^{eq}_{imp} = (S^{eq}_{imp}, \Omega^{eq}_{imp})$. Esta signatura presenta un nuevo género, que denotamos por $imp_{\Sigma^{eq}}$, que se incorpora como primer argumento a cada operación $\omega\colon s_1 \ldots s_n \to s$, $n \geq 0$, de Σ^{eq} (incluidas las operaciones de igualdad) para dar lugar a la operación $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \ldots s_n \to s$ de Σ^{eq}_{imp}.

De los distintos marcos alternativos que estudiamos para situar la signatura resultado de la operación *imp*, comentábamos que posiblemente el más adecuado fuera la teoría oculta. En este capítulo también alojaremos las signaturas Σ^{eq}_{imp} en este marco.

La signatura Σ^{eq}_{imp} se considerará una signatura oculta, con un único genero oculto $imp_{\Sigma^{eq}}$, sobre la signatura visible $V\Sigma^{eq}_{imp} = (S^{eq}, C)$ y el dominio de datos $D = (\{D_s\}_{s \in S^{eq}}, C)$, donde C es un conjunto de constantes, $d\colon \to s$, una por cada $d \in D_s$ y cada género $s \in S^{eq}$. Para que quede bien definida como signatura oculta es preciso incorporar este conjunto de constantes a Σ^{eq}_{imp}.

Si nos centramos ahora en las características semánticas de la construcción *imp*, tenemos que el espacio de salida de esta aplicación es ahora $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$. Pretendemos que la imagen de una de estas álgebras sea una Σ^{eq}_{imp}-álgebra que representa a una familia de Σ^{eq}-álgebras. Lo primero que debemos notar es que la signatura Σ^{eq}_{imp} no es una signatura con igualdad. Debemos pensar que Σ^{eq}_{imp} significa por construcción $(\Sigma^{eq})_{imp}$, lo que no es equivalente a $(\Sigma_{imp})^{eq}$. La diferencia estriba en las operaciones de igualdad. En $(\Sigma^{eq})_{imp}$ no tenemos ninguna operación de igualdad, entendidas como operaciones binarias con género resultado *bool*. Por ello, las Σ^{eq}_{imp}-álgebras que utilizaremos no serán álgebras con igualdad. Sin embargo, con respecto a la parcialidad ocurre algo distinto. Para recoger mediante la operación *imp* una familia de Σ^{eq}-álgebras parciales, vamos a usar una Σ^{eq}_{imp}-álgebra parcial.

La categoría en la que situaremos las Σ^{eq}_{imp}-álgebras que obtenemos es $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$. Esta categoría tiene como objetos Σ^{eq}_{imp}-álgebras *parciales ocultas*. Es decir, Σ^{eq}_{imp}-álgebras parciales cuya restricción a la signatura visible $V\Sigma^{eq}_{imp}$ es el dominio de datos D. Además, el dominio de parcialidad de sus operaciones está definido sobre los géneros visibles a través de *dom* en la correspondiente operación de Σ^{eq}, y se "completa" total sobre el género oculto. Más concretamente, las Σ^{eq}_{imp}-álgebras A de

$HPAlg^{D,dom}(\Sigma^{eq}_{imp})$ verifican que $A_s = D_s$ para cada $s \in S^{eq}$ y $d_A = d$ para cada $d\colon \to s \in C$, y el dominio de definición de cada operación $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \ldots s_n \to s \in \Omega^{eq}_{imp}$, $n \geq 0$, viene dado por $Def(imp\_\omega_A) = A_{imp_{\Sigma^{eq}}} \times dom_\omega$.

Esperamos que la notación que utilizamos para representar los dominios de definición no produzca confusión: mientras que en $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$ el dominio dom coincide con los dominios de definición de las operaciones, en $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$, dom sólo sirve para determinarlos.

Los morfismos de la categoría $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$ son los Σ^{eq}_{imp}-homomorfismos *débiles ocultos*, es decir, Σ^{eq}_{imp}-homomorfismos débiles identidad sobre los géneros visibles.

Si A y B son Σ^{eq}_{imp}-álgebras de $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$, un Σ^{eq}_{imp}-homomorfismo débil oculto, $h\colon A \to B$, está determinado por una función total $h_{imp_{\Sigma^{eq}}}\colon A_{imp_{\Sigma^{eq}}} \to B_{imp_{\Sigma^{eq}}}$ (recordamos que $imp_{\Sigma^{eq}}$ es el único género oculto), tal que para cada operación $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \ldots s_n \to s \in \Omega^{eq}_{imp}$, $n \geq 0$, y para cada $(a, d_1, \ldots, d_n) \in Def(imp\_\omega_A)$ se verifica que

$$imp\_\omega_A(a, d_1, \ldots, d_n) = imp\_\omega_B(h_{imp_{\Sigma^{eq}}}(a), d_1, \ldots, d_n).$$

Debemos notar que para cada $(a, d_1, \ldots, d_n) \in Def(imp\_\omega_A)$ se tiene que $(h_{imp_{\Sigma^{eq}}}(a), d_1, \ldots, d_n) \in Def(imp\_\omega_B)$. Es más, también se obtiene que si $(a, d_1, \ldots, d_n) \notin Def(imp\_\omega_A)$ entonces $(h_{imp_{\Sigma^{eq}}}(a), d_1, \ldots, d_n) \notin Def(imp\_\omega_B)$. Por lo tanto el anterior morfismo, además de débil, es fuerte. El hecho de trabajar sobre dominios de definición prefijados para las operaciones de la signatura Σ^{eq} que constituye la parte visible de la signatura Σ^{eq}_{imp}, junto con la condición de la teoría oculta de exigir operaciones identidades para las operaciones del homomorfismo correspondientes a operaciones visibles, hace que los homomorfismos débiles ocultos actúen como homomorfismos totales si nos restringimos a los elementos de este dominio de definición prefijado. Posteriormente nos situaremos en una categoría menos restrictiva, ya que no fijaremos los dominios de definición de las operaciones y tendrá más morfismos. Aunque desde el punto de vista algebraico la categoría a la que acabamos de referirnos puede resultar más interesante, lo cierto es que $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$ está más próxima al sistema EAT. A nivel de álgebras cada operación viene acompañada de su dominio de definición; sin embargo, esto no ocurre a nivel de implementación (la manera de representar una operación es mediante un código funcional, y un código funcional no determina un único dominio). La idea que subyace en EAT es que el dominio de una operación debe estar prefijado. Por ello preferimos desarrollar primero este caso con el propósito de aproximarnos lo más posible a EAT.

Como ya se ha comentado, lo normal es que no nos interesan todas las álgebras de $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$, sino una subcategoría $\mathcal{C}$, que contiene a las álgebras que representan a las estructuras que pretendemos modelar. La imagen de esta categoría por la operación imp es una subcategoría plena $\mathcal{C}_{imp}$ de $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$. Los objetos de esta subcategoría son las Σ^{eq}_{imp}-álgebras parciales ocultas A tales que, para todo $a \in A_{imp_{\Sigma^{eq}}}$, la Σ^{eq}-álgebra parcial A^a que definimos a continuación es un objeto de $\mathcal{C}$. Para un elemento $a \in A_{imp_{\Sigma^{eq}}}$ se define la Σ^{eq}-álgebra parcial A^a de modo que asocia a cada género

$s \in S^{eq}$ el conjunto $A^a_s = D_s$, y a cada operación $\omega\colon s_1 \dots s_n \to s \in \Omega^{eq}$, $n \geq 0$, la función parcial ω_{A^a}, que tiene dominio de definición $Def(\omega_{A^a}) = dom_\omega$, y está definida por $\omega_{A^a}(d_1, \dots, d_n) := imp\_\omega_A(a, d_1, \dots, d_n)$ para cada $(d_1, \dots, d_n) \in dom_\omega$. Debemos observar que en particular A^a debe ser un álgebra con igualdad.

En estas condiciones la categoría $\mathcal{C}_{imp}$ también va a disponer de álgebra final. Este objeto va a presentar como soporte del género destacado tuplas funcionales (ahora parciales) que representen álgebras de $\mathcal{C}$, reflejando la forma en que han sido implementadas las estructuras de datos de EAT [81]. De esta forma, se entiende la capacidad de EAT para definir estructuras con operaciones parciales y establecer identificaciones dentro de los conjuntos soportes a través de igualdades.

4.5.1 Objeto final

Sean la signatura con igualdad $\Sigma^{eq} = (S^{eq}, \Omega^{eq})$, el dominio de datos D, el dominio de definición dom y la signatura oculta $\Sigma^{eq}_{imp} = (S^{eq} \cup \{imp_{\Sigma^{eq}}\}, \Omega^{eq}_{imp} \cup C)$ tal y como se han definido en la sección anterior. Sea $\mathcal{C}$ subcategoría de $P\tilde{A}lg^{D,dom}(\Sigma^{eq})$ y $\mathcal{C}_{imp}$ su correspondiente subcategoría de $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$. Entonces, definimos un objeto en $\mathcal{C}_{imp}$, que denotamos por A^{can}, de la siguiente manera:

- $A^{can}_s = D_s$, para cada género s de S^{eq},
- $A^{can}_{imp_{\Sigma^{eq}}} = \{(f_\omega)_{\omega \in \Omega^{eq}} \mid \langle D, (f_\omega, dom_\omega)_{\omega \in \Omega^{eq}} \rangle \in \mathcal{C}\}$,
- $d_{A^{can}} = d$, para cada constante $d\colon \to s$ de C,
- $imp\_\omega_{A^{can}}((f_\delta)_{\delta \in \Omega^{eq}}, d_1, \dots, d_n) := f_\omega(d_1, \dots, d_n)$, para todo $(f_\delta)_{\delta \in \Omega^{eq}} \in A^{can}_{imp_{\Sigma^{eq}}}$ y $(d_1, \dots, d_n) \in dom_\omega$, para cada $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \dots s_n \to s \in \Omega^{eq}_{imp}$.

Por definición, se tiene que la Σ^{eq}_{imp}-álgebra A^{can} es un objeto de $\mathcal{C}_{imp}$ (cada dato de $A^{can}_{imp_{\Sigma^{eq}}}$ define un objeto de $\mathcal{C}$). Además es final dentro de esta categoría, como se prueba en la siguiente proposición. Como estamos trabajando sobre dominios de definición prefijados, la demostración de este resultado es muy parecida a la del caso total.

Proposición 4.5.1. *La Σ^{eq}_{imp}-álgebra parcial A^{can} es objeto final de $\mathcal{C}_{imp}$.*

Demostración. Dado un elemento $B \in \mathcal{C}_{imp}$, podemos definir un Σ^{eq}_{imp}-homomorfismo oculto débil a través de la función total $h^{can}\colon B_{imp_{\Sigma^{eq}}} \to A^{can}_{imp_{\Sigma^{eq}}}$ tal que, para cada $b \in B_{imp_{\Sigma^{eq}}}$, $h^{can}(b) := (\omega_{B^b})_{\omega \in \Omega^{eq}}$.

Para cada $imp\_\sigma\colon imp_{\Sigma^{eq}} s_1 \dots s_n \to s \in \Omega^{eq}_{imp}$, y cada $(b, d_1, \dots, d_n) \in Def(imp\_\sigma_B)$ la siguiente secuencia de igualdades prueban la condición de ser h^{can} homomorfismo: $imp\_\sigma_{A^{can}}(h^{can}(b), d_1, \dots, d_n) = imp\_\sigma_{A^{can}}((\omega_{B^b})_{\omega \in \Omega^{eq}}, d_1, \dots, d_n) = \sigma_{B^b}(d_1, \dots, d_n) = imp\_\sigma_B(b, d_1, \dots, d_n)$.

Además el morfismo anterior es único. Si suponemos que existe otro Σ^{eq}_{imp}-homomorfismo oculto débil, en el nivel correspondiente al género distinguido se tiene una función total $h\colon B_{imp_{\Sigma^{eq}}} \to A^{can}_{imp_{\Sigma^{eq}}}$ tal que, para cada $b \in B_{imp_{\Sigma^{eq}}}$, $h(b) := (f_\delta)_{\delta\in\Omega^{eq}}$, con $(f_\delta)_{\delta\in\Omega^{eq}} \in A^{can}_{imp_{\Sigma^{eq}}}$. Entonces, para cada $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \ldots s_n \to s \in \Omega^{eq}_{imp}$ y cada $(b, d_1, \ldots, d_n) \in Def(imp\_\omega_B)$ tenemos, por ser h homomorfismo, la siguiente cadena de igualdades: $\omega_{B^b}(d_1, \ldots, d_n) = imp\_\omega_B(b, d_1, \ldots, d_n) = imp\_\omega_{A^{can}}(h(b), d_1, \ldots, d_n) = imp\_\omega_{A^{can}}((f_\delta)_{\delta\in\Omega^{eq}}, d_1, \ldots, d_n) = f_\omega(d_1, \ldots, d_n)$. Por lo que $\omega_{B^b}(d_1, \ldots, d_n) = f_\omega(d_1, \ldots, d_n)$. Como además ambas funciones comparten el mismo dominio de definición dom_ω, para cada $\omega \in \Omega^{eq}$, se deduce la igualdad $h^{can} = h$. ■

4.5.2 Ejemplos

A continuación detallaremos la imagen por la aplicación *imp* de las categorías $\mathcal{C}^{GRP_{\mathbb{Z}}}$, $\mathcal{C}^{ss}$ y $\mathcal{C}^{cc}$ que hemos propuesto como ejemplos en la sección anterior. Obtenemos de esta forma especificaciones para familias de grupos, conjuntos simpliciales y complejos de cadenas, respectivamente.

4.5.2.1 Familias de grupos sobre $\mathbb{Z}$

Partimos de la subcategoría $\mathcal{C}^{GRP_{\mathbb{Z}}}$ de $\tilde{A}lg^D(\mathtt{GRP}^{eq})$ que hemos definido anteriormente, con GRP la signatura para grupo y $D_g = \mathbb{Z}$. En este ejemplo trabajábamos con álgebras totales. Entonces, la signatura $\mathtt{GRP}^{eq}_{imp}$ es:

signatura $\mathtt{GRP}^{eq}_{imp}$
  géneros g, $bool$, $imp_{\mathtt{GRP}^{eq}}$
  operaciones
    $imp\_prd\colon imp_{\mathtt{GRP}^{eq}}\ g\ g \to g$
    $imp\_inv\colon imp_{\mathtt{GRP}^{eq}}\ g \to g$
    $imp\_unt\colon imp_{\mathtt{GRP}^{eq}} \to g$
    $imp\_eq^g\colon imp_{\mathtt{GRP}^{eq}} g\ g \to bool$
finsig

Esta signatura se considera una signatura oculta con el género $imp_{\mathtt{GRP}^{eq}}$ como único género oculto. Para ello es necesario incorporar a la misma una constante por cada elemento de $D_g = \mathbb{Z}$ y $D_{bool} = \{true, false\}$.

La categoría $\mathcal{C}^{GRP_{\mathbb{Z}}}_{imp}$, imagen de la categoría $\mathcal{C}^{GRP_{\mathbb{Z}}}$ por la operación *imp*, está formada por las $\mathtt{GRP}^{eq}_{imp}$-álgebras A de $HAlg^D(\mathtt{GRP}^{eq}_{imp})$, que verifiquen que para todo $a \in A_{imp_{\mathtt{GRP}^{eq}}}$, la $\mathtt{GRP}^{eq}$-álgebra A^a pertenece a $\mathcal{C}^{GRP_{\mathbb{Z}}}$. En este caso A^a está formada por los conjuntos $\{D_g, D_{bool}\}$ y las funciones $\{imp\_prd_A(a, -), imp\_inv_A(a, -), imp\_unt_A(a, -), imp\_eq^g_A(a, -)\}$. Entonces, $imp\_eq^g_A(a, -)$ define una relación de congruencia para A^a y el álgebra cociente $\tilde{A}^a$ por esta relación de congruencia es un grupo.

Por ejemplo, podemos recoger la familia de grupos finitos $\mathbb{Z}/n\mathbb{Z}$ con $n > 1$, a través del álgebra G de $\mathcal{C}_{imp}^{GRP_{\mathbb{Z}}}$ que definimos a continuación. Como soporte para el género $imp_{\mathtt{GRP}^{eq}}$ se considera $G_{imp_{\mathtt{GRP}^{eq}}} := \{n \in \mathbb{N} \mid n > 1\}$ y como funciones se tiene:

$$\begin{array}{l} imp\_prd_G(n, d1, d2) = d1 + d2 \\ imp\_inv_G(n, d) = -d \\ imp\_unt_G(n) = 0 \\ imp\_eq_G^g(n, d1, d2) = true \quad \text{si y sólo si} \quad n \text{ divide a } d1 - d2 \end{array}$$

Así, para cada $n \in G_{imp_{\mathtt{GRP}^{eq}}}$, se tiene una $\mathtt{GRP}^{eq}$-álgebra $G^n = \langle \mathbb{Z}, (imp\_prd_G(n, -), imp\_inv_G(n, -), imp\_unt_G(n), imp\_eq_G^g(n, -)) \rangle$ que pertenece a $\mathcal{C}^{GRP_{\mathbb{Z}}}$. La función $imp\_eq_G^g(n, -)$ define de modo evidente una relación de congruencia en G^n y el álgebra $\tilde{G^n}$ es una representación del grupo $\mathbb{Z}/n\mathbb{Z}$.

Un segundo ejemplo de familia de grupos que podemos definir sobre $\mathbb{Z}$ es la familia de los subgrupos $\{n\mathbb{Z},\ n \in \mathbb{N}\}$. Para ello consideramos el álgebra H de $\mathcal{C}_{imp}^{GRP_{\mathbb{Z}}}$ que tiene como soporte para el género distinguido $H_{imp_{\mathtt{GRP}^{eq}}} = \mathbb{N}$ y las funciones:

$$\begin{array}{l} imp\_prd_H(n, d1, d2) = d1 + d2 \\ imp\_inv_H(n, d) = -d \\ imp\_unt_H(n) = 0 \\ imp\_eq_H^g(n, d1, d2) = true \quad \text{si y sólo si} \quad d1 = d2 \text{ y } n \text{ divide a } d1 \end{array}$$

De este modo, para cada $n \in H_{imp_{\mathtt{GRP}^{eq}}}$, si fijamos n como primer argumento de las funciones, obtenemos una $\mathtt{GRP}^{eq}$-álgebra H^n tal que el álgebra cociente $\tilde{H^n}$ es una representación del grupo $n\mathbb{Z}$.

Respecto del objeto final dentro de la categoría $\mathcal{C}_{imp}^{GRP_{\mathbb{Z}}}$, según su propia definición, tiene como soporte para el género oculto el conjunto de tuplas funcionales totales ($*\colon \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$, $-\colon \mathbb{Z} \to \mathbb{Z}$, $e\colon\ \to \mathbb{Z}$, $=\colon \mathbb{Z} \times \mathbb{Z} \to \{true, false\}$) que definan un elemento de $\mathcal{C}^{GRP_{\mathbb{Z}}}$. Las interpretaciones de una operación en esta álgebra consiste en aplicar la correspondiente operación de grupo (que forma parte del primer argumento) sobre el resto de argumentos (que representan datos enteros).

4.5.2.2 Familias de conjuntos simpliciales

Partimos en esta ocasión de la subcategoría $\mathcal{C}^{ss}$ de álgebras de $P\tilde{A}lg^{D^{ss}, dom^{ss}}(\Sigma^{ss,eq})$, con Σ^{ss} la signatura para un conjunto simplicial, y D^{ss} y dom^{ss} el dominio datos y el dominio de definición prefijados para esta signatura. La signatura $\Sigma_{imp}^{ss,eq}$ es:

signatura $\Sigma^{ss,eq}_{imp}$
géneros nat, gsm, asm, $bool$, $imp_{\Sigma^{ss,eq}}$
operaciones
$imp\_face\colon imp_{\Sigma^{ss,eq}} nat\ nat\ asm \to asm$
$imp\_dgn\colon imp_{\Sigma^{ss,eq}} nat\ nat\ asm \to asm$
$imp\_eq^{nat}\colon imp_{\Sigma^{ss,eq}} nat\ nat \to bool$
$imp\_eq^{gsm}\colon imp_{\Sigma^{ss,eq}} gsm\ gsm \to bool$
$imp\_eq^{asm}\colon imp_{\Sigma^{ss,eq}} asm\ asm \to bool$
finsig

Consideramos esta signatura como una signatura oculta con el género $imp_{\Sigma^{ss,eq}}$ como único género oculto, por lo que es necesario incorporar a la misma una constante por cada elemento de D^{ss}.

Entonces la subcategoría $\mathcal{C}^{ss}_{imp}$ de $HPAlg^{D^{ss},dom^{ss}}(\Sigma^{ss,eq}_{imp})$ está formada por las $\Sigma^{ss,eq}_{imp}$-álgebras parciales ocultas A, con dominio de definición para cada una de sus funciones parciales el definido a partir de dom^{ss} en la correspondiente operación, y tal que para todo $a \in A_{imp_{\Sigma^{ss,eq}}}$ la $\Sigma^{ss,eq}$-álgebra A^a dada por las operaciones ($imp\_face_A(a,-)$, $imp\_dng_A(a,-)$,$imp\_eq^{nat}_A(a,-)$, $imp\_eq^{gsm}_A(a,-)$, $imp\_eq^{asm}_A(a,-)$) sea un objeto de $\mathcal{C}^{ss}$.

Una familia muy importante de conjuntos simpliciales que aparece explícitamente como ejemplo en EAT son los llamados símplices estándar. El *símplice estándar* de dimensión m, denotado habitualmente en la literatura por Δ^m, tiene como símplices de dimensión k las listas ordenadas con exactamente $k+1$ naturales entre 0 y m. El operador degeneración i-ésimo sobre una de estas listas duplica el elemento que ocupa la posición $i+1$ en la lista. El operador cara i-ésimo elimina el elemento que ocupa la posición $i+1$.

En este ejemplo podemos observar fácilmente la Propiedad 4.4.1 que permite representar un símplice de forma única como símplice abstracto: la repetición de un número dentro de la lista de naturales que forman un símplice puede expresarse como la aplicación del operador degeneración de índice el correspondiente a la posición de ese elemento sobre el símplice que no tiene dicho elemento. De este modo obtenemos que las listas de naturales estrictamente decrecientes corresponden con los símplices geométricos.

A través de nuestra técnica podemos modelar la familia de los símplices estándar. En este caso, para tener determinado el dominio de datos es suficiente con fijar

$$D^{ss}_{gsm} := \{(a_1,\ldots,a_k) \mid k \in \mathbb{N}, k \geq 1, a_i \in \mathbb{N}, \forall i = 1,\ldots,k \text{ y } a_i < a_j \text{ si } i < j\}$$

de forma que la dimensión de una lista de longitud k es $k-1$, es decir, $dim^{ss}_{gsm}((a_1,\ldots,a_k)) = k-1$.

El dominio de datos para el género asm está entonces formado por parejas de listas de naturales, la primera estrictamente decreciente y la segunda estrictamente creciente.

Los dominios de definición de las operaciones están prefijados, a partir de la operación de dimensión para los símplices geométricos.

Una $\Sigma_{imp}^{ss,eq}$-álgebra de $HPAlg^{D^{ss},dom^{ss}}(\Sigma_{imp}^{ss,eq})$ que modela la familia $\{\Delta^m\}_{m\in\mathbb{N}}$ es el álgebra $\Delta^{\mathbb{N}}$ que definimos a continuación. El soporte para el género destacado es $\Delta^{\mathbb{N}}_{imp_{\Sigma^{ss,eq}}} := \mathbb{N}$. Con respecto a las operaciones tenemos que:

· La función parcial $imp\_dgn_{\Delta^{\mathbb{N}}}$ está definida como la operación degeneración predefinida en $\mathcal{C}^{ss}$ independientemente del argumento $m \in \Delta^{\mathbb{N}}_{imp_{\Sigma^{ss,eq}}}$,

· La función parcial $imp\_face_{\Delta^{\mathbb{N}}}$ tiene dominio de definición: $Def(imp\_face_{\Delta^{\mathbb{N}}}) = \{(m, i, n, \langle(j_l, \ldots, j_1), (a_1, \ldots, a_k)\rangle) \in \mathbb{N} \times \mathbb{N} \times \mathbb{N} \times D^{ss}_{asm} \mid 0 \leq i \leq n,\ l \geq 0,\ k \geq 1,\ n > 0,\ n = l + k - 1\}$. Esta función se define sobre los símplices geométricos del dominio por:

$$imp\_face_{\Delta^{\mathbb{N}}}(m, i, k-1, \langle(\ ), (a_1, \ldots, a_k)\rangle) := \langle(\ ), (a_1, \ldots, a_i, a_{i+2}, \ldots, a_k)\rangle,$$

y se extiende la definición a todos los símplices abstractos a través de las propiedades que permiten intercambiar los operadores cara y degeneración.

· La función parcial $imp\_eq^{gsm}_{\Delta^{\mathbb{N}}}$ tiene dominio de definición: $Def(imp\_eq^{gsm}_{\Delta^{\mathbb{N}}}) = \{(m, (a_1, \ldots, a_k), (b_1, \ldots, b_l)) \in \mathbb{N} \times D^{ss}_{gsm} \times D^{ss}_{gsm} \mid k = l\}$. Esta función se define sobre estos elementos del dominio por:

$$imp\_eq^{gsm}_{\Delta^{\mathbb{N}}}(m, (a_1, \ldots, a_k), (b_1, \ldots, b_k)) := true \text{ sii } a_i = b_i, \forall i = 1, \ldots, k \text{ y } a_k \leq m.$$

· La función parcial $imp\_eq^{asm}_{\Delta^{\mathbb{N}}}$ para un elemento de $\Delta^{\mathbb{N}}_{imp_{\Sigma^{ss,eq}}}$ se define a partir de la correspondiente aplicación $imp\_eq^{gsm}_{\Delta^{\mathbb{N}}}$ para ese elemento tal y como se ha exigido en la categoría $\mathcal{C}^{ss}$.

· Por último, la función total $imp\_eq^{nat}_{\Delta^{\mathbb{N}}}$ se define como la igualdad literal de naturales, independientemente del argumento $\Delta^{\mathbb{N}}_{imp_{\Sigma^{ss,eq}}}$.

La notación elegida puede resultar confusa. El nombre $\Delta^{\mathbb{N}}$ que hemos elegido para el álgebra anterior se utiliza habitualmente para denotar al complejo simplicial libremente generado por los enteros positivos. Este conjunto simplicial tiene como símplices geométricos exactamente a D^{ss}_{gsm}, y como operador cara al definido anteriormente (independientemente del primer género) sobre los símplices que se obtienen a partir de estos símplices geométricos. De este modo, $\Delta^m \subset \Delta^{\mathbb{N}}$, para todo $m \in \mathbb{N}$. Entonces podemos obtener los símplices geométricos de Δ^m a partir de D^{ss}_{gsm} a través de la anterior relación de equivalencia parcial cuyo dominio es exactamente este conjunto de símplices geométricos.

El objeto final en este caso tiene como soporte para el género oculto tuplas de funciones parciales $(fc, dg, =_{nat}, =_{gsm}, =_{asm})$ tal que $\langle D^{ss}, ((fc, dom^{ss}_{face}), (dg, dom^{ss}_{dgn}), (=_{nat}, dom^{ss}_{eq^{nat}}), (=_{gsm}, dom^{ss}_{eq^{gsm}}), (=_{asm}, dom^{ss}_{eq^{asm}}))\rangle \in \mathcal{C}^{ss}$.

Es importante señalar que la implementación que en sistema EAT se hace de los conjuntos simpliciales no incluye dentro de la tupla funcional todas las componentes anteriores. Por ejemplo, no están presentes las operaciones de igualdad en los géneros *nat* y *asm* ni tampoco la función degeneración. En realidad, estas funciones no son necesarias para determinar un conjunto simplicial de $\mathcal{C}^{ss}$. La función parcial $=_{asm}$ se obtiene a partir de la función parcial $=_{gsm}$ y la igualdad $=_{nat}$ está prefijada: es la igualdad literal. Además, la operación degeneración viene determinada por la representación que se ha fijado para las combinaciones.

Así, una tupla formada por la pareja de funciones parciales $(fc, =_{gsm})$ es suficiente para servir de índice de un objeto de $\mathcal{C}^{ss}$. Podemos obtener entonces una expresión equivalente del objeto final en la que el soporte en el género destacado esté formado por parejas de funciones del tipo la anterior.

No obstante, la representación elegida por EAT todavía es algo más sencilla. En lugar de aparecer en la tupla una función parcial (fc, dom^{ss}_{face}), que representa al operador cara sobre los símplices abstractos, aparece una función parcial $(fc', Def(fc'))$ que representa al operador cara exclusivamente sobre los símplices geométricos del dominio de definición. Esta función, usando las propiedades que permiten intercambiar las operaciones cara y degeneración, es suficiente para obtener la definición de la operación cara sobre todos los símplices. De esta forma, una tupla tan sencilla como $(fc', =_{gsm})$ es utilizada en EAT para servir de índice de un objeto de $\mathcal{C}^{ss}$.

4.5.2.3 Familias de complejos de cadenas

Por último, y de forma más resumida, abordaremos el ejemplo de las familias de complejos de cadenas. Partimos de la subcategoría $\mathcal{C}^{cc}$ de álgebras de $P\tilde{A}lg^{D^{cc},dom^{cc}}(\Sigma^{cc,eq})$, con Σ^{cc} la signatura para un complejo de cadenas y D^{cc}, dom^{cc}, el dominio datos y el dominio de definición prefijados para esta signatura. La signatura $\Sigma^{cc,eq}_{imp}$ es:

signatura $\Sigma^{cc,eq}_{imp}$
géneros *int*, *gnr*, *cmb*, *bool*, $imp_{\Sigma^{cc,eq}}$
operaciones
$imp\_add\colon imp_{\Sigma^{cc,eq}} cmb\ cmb \to cmb$
$imp\_zero\colon imp_{\Sigma^{cc,eq}} int \to cmb$
$imp\_mns\colon imp_{\Sigma^{cc,eq}} cmb \to cmb$
$imp\_mlt\colon imp_{\Sigma^{cc,eq}} int\ cmb \to cmb$
$imp\_df\colon imp_{\Sigma^{cc,eq}} cmb \to cmb$
$imp\_eq^{int}\colon imp_{\Sigma^{cc,eq}} int\ int \to bool$
$imp\_eq^{gnr}\colon imp_{\Sigma^{cc,eq}} gnr\ gnr \to bool$
$imp\_eq^{cmb}\colon imp_{\Sigma^{cc,eq}} cmb\ cmb \to bool$
finsig

Esta signatura es una signatura oculta con el género $imp_{\Sigma^{cc,eq}}$ como único género oculto, por lo que es necesario incorporar a la misma una constante por cada elemento de D^{cc}.

Entonces, la subcategoría $\mathcal{C}^{cc}_{imp}$ de $HPAlg^{D^{cc},dom^{cc}}(\Sigma^{cc,eq}_{imp})$ está formada por las $\Sigma^{cc,eq}_{imp}$-álgebras parciales ocultas A, cuyo dominio de definición para sus operaciones está definido a partir del respectivo dominio en dom^{cc} (siendo total sobre el género oculto), y tal que, para cada elemento del soporte $a \in A_{imp_{\Sigma^{cc,eq}}}$, la $\Sigma^{cc,eq}$-álgebra que se obtiene fijando el elemento a como primer argumento define un álgebra de $\mathcal{C}^{cc}$.

El objeto canónico dentro de esta categoría tiene como soporte para el género oculto un conjunto de tuplas de funciones parciales $(+, 0, -, *, d, =_{int}, =_{gnr}, =_{cmb})$ con dominio de definición dom^{cc}, que constituyen la interpretación de las operaciones en un álgebra de $\mathcal{C}^{cc}$.

Si nos fijamos en el sistema EAT podemos observar que, nuevamente, la implementación concreta que se usa para trabajar con complejos de cadenas no incluye todas esas funciones. Esta vez no están presentes las cuatro operaciones que determinan la estructura de $\mathbb{Z}$-módulo, ni tampoco las igualdades correspondientes a los enteros y a las combinaciones. En realidad, estas funciones no son necesarias para determinar un complejo de cadenas de $\mathcal{C}^{cc}$. Las funciones 0, $-$, $*$ quedan fijadas para ser coherentes con la estructura elegida para las combinaciones y la forma de sumar combinaciones también queda determinada a partir de la estructura de éstas (junto con la igualdad). La igualdad para los enteros es la igualdad literal y la igualdad para las combinaciones se obtiene a partir de la igualdad sobre los generadores.

De esta forma, una tupla formada por la pareja de funciones $(d, =_{gnr})$, diferencial e igualdad entre generadores, es suficiente para servir de índice de un objeto de $\mathcal{C}^{cc}$. Podemos obtener entonces una expresión equivalente del objeto final en la que el soporte en el género destacado esté formado por parejas de funciones como la anterior.

EAT implementa una representación todavía más sencilla de un complejo de cadenas. En realidad, para tener completamente definida la operación diferencial d sobre las combinaciones es suficiente con disponer de una operación d' que defina la diferencial sobre las combinaciones formadas por un único monomio con coeficiente la unidad (éstas vienen en realidad a representar a los generadores). Podemos recuperar la diferencial simplemente extendiendo por linealidad. EAT utiliza una tupla de la forma $(d', =_{gnr})$ para servir de índice de un objeto de $\mathcal{C}^{ss}$.

4.5.3 Dominios de definición no prefijados

Hemos comentado previamente que la necesidad de fijar un dominio de definición para las operaciones de la signatura Σ^{eq} viene motivado por lo que ocurre a nivel de implementaciones. A nivel de algebras no es necesario prefijar este dominio de definición, ya que cada operación lleva consigo el dominio sobre el que está definida. Si trabajamos sobre dominios de definición no prefijados, obtenemos una categoría más rica en objetos, en la que los homomorfismos débiles ocultos ya no tienen por qué ser fuertes. Sin embargo, para obtener en tal caso un objeto final debemos trabajar con homomorfismos fuertes ya que, en caso contrario, perdemos la propiedad de unicidad (puede haber más de un morfismo con codominio el objeto final).

4.5.3.1 Trabajar con homomorfismos fuertes

Podemos repetir el estudio que hemos realizado para modelar los objetos de una categoría $\mathcal{T}$, pero ahora sin fijar los dominios de definición de las funciones en los modelos. Para ello consideramos una signatura Σ y un dominio de datos D para Σ. Trabajaremos en la categoría $P\tilde{A}lg^D(\Sigma^{eq})$ de Σ^{eq}-álgebras parciales con igualdad con conjuntos soporte prefijados por D. Los morfismos de esta categoría seguirán siendo los Σ^{eq}-homomorfismos débiles identidad, que obviamente también son fuertes. Es decir, en este caso tenemos que $P\tilde{A}lg^D(\Sigma^{eq})$ coincide con $P_f\tilde{A}lg^D(\Sigma^{eq})$. Lo mismo que antes, no necesariamente nos interesa toda esta categoría, sino un subcategoría propia, que denotamos por $\mathcal{C}_f$, que verifique condiciones semánticas que permitan modelar los objetos de $\mathcal{T}$. En esta ocasión vamos a permitir, para cada operación, un dominio de definición distinto en cada álgebra.

Si ampliamos ahora la definición de la operación *imp* para que tenga como espacio de salida $P_f\tilde{A}lg^D(\Sigma^{eq})$, entonces nos vemos obligados a ampliar también el espacio de llegada. En este caso, la categoría en la que se situarán las Σ^{eq}_{imp}-álgebras imagen por esta operación es $HP_fAlg^D(\Sigma^{eq}_{imp})$, categoría de las Σ^{eq}_{imp}-álgebras parciales ocultas con dominio de datos D. Los morfismos de la categoría $HP_fAlg^D(\Sigma^{eq}_{imp})$ son los Σ^{eq}_{imp}-homomorfismos fuertes ocultos.

Si A y B son Σ^{eq}_{imp}-álgebras de $HP_fAlg^D(\Sigma^{eq}_{imp})$ un Σ^{eq}_{imp}-homomorfismo débil oculto, $h\colon A \to B$, está determinado por una función total $h_{imp_{\Sigma^{eq}}}\colon A_{imp_{\Sigma^{eq}}} \to B_{imp_{\Sigma^{eq}}}$, tal que para cada operación $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \ldots s_n \to s \in \Omega^{eq}_{imp}$, se tiene:

si $(a, d_1, \ldots, d_n) \in Def(imp\_\omega_A)$ entonces $(h_{imp_{\Sigma^{eq}}}(a), d_1, \ldots, d_n) \in Def(imp\_\omega_B)$, y en ese caso

$$imp\_\omega_A(a, d_1, \ldots, d_n) = imp\_\omega_B(h_{imp_{\Sigma^{eq}}}(a), d_1, \ldots, d_n).$$

El homomorfismo anterior es fuerte si además se verifica que:

si $(a, d_1, \ldots, d_n) \notin Def(imp\_\omega_A)$ entonces $(h_{imp_{\Sigma^{eq}}}(a), d_1, \ldots, d_n) \notin Def(imp\_\omega_B)$.

En este caso, al no estar prefijado los dominios de definición, obviamente pueden existir Σ^{eq}_{imp}-homomorfismos débiles que no sean fuertes.

Entonces, la operación *imp* define a partir de una subcategoría $\mathcal{C}_f$ de $P_f\tilde{A}lg^D(\Sigma^{eq})$, una subcategoría plena $\mathcal{C}_{fimp}$ de $HP_fAlg^D(\Sigma^{eq}_{imp})$. Ahora los objetos de esta categoría son las Σ^{eq}_{imp}-álgebras parciales A, tales que, para todo $a \in A_{imp_{\Sigma^{eq}}}$, la Σ^{eq}-álgebra parcial A^a dada por los conjuntos $A^a_s = D_s$, para cada $s \in S^{eq}$, y las funciones parciales $\omega_{A^a}(d_1, \ldots, d_n) := imp\_\omega_A(a, d_1, \ldots, d_n)$ cuyo dominio de definición es $Def(\omega_{A^a}) := \{(d_1, \ldots, d_n) \in D_1 \times \cdots \times D_n \mid (a, d_1, \ldots, d_n) \in Def(imp\_\omega_A)\}$, para cada $\omega\colon s_1 \ldots s_n \to s \in \Omega^{eq}$, es un objeto de $\mathcal{C}_f$.

En este caso, la categoría $\mathcal{C}_{fimp}$ también posee objeto final. Este objeto final, que denotaremos por A_f^{can} puede describirse por:

- $A_{fs}^{can} = D_s$, para cada $s \in S^{eq}$,
- $A_{fimp_{\Sigma^{eq}}}^{can} = \{(f_\omega, Def(f_\omega))_{\omega\in\Omega^{eq}} \mid \langle D, (f_\omega, Def(f_\omega))_{\omega\in\Omega^{eq}}\rangle \in \mathcal{C}_f\}$,
- $d_{A_f^{can}} = d$, para cada constante $d\colon \to s$ de C,
- para cada $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \dots s_n \to s \in \Omega_{imp}^{eq}$, $n \geq 0$, $Def(imp\_\omega_{A_f^{can}}) := \{\big((f_\delta, Def(f_\delta))_{\delta\in\Omega^{eq}}, d_1, \dots, d_n\big) \in A_{fimp_{\Sigma^{eq}}}^{can} \times D_{s_1} \times \cdots \times D_{s_n} \mid (d_1, \dots, d_n) \in Def(f_\omega)\}$ y sobre este dominio de definición se define $imp\_\omega_{A_f^{can}}\big((f_\delta, Def(f_\delta))_{\delta\in\Omega^{eq}}, d_1, \dots, d_n\big) := f_\omega(d_1, \dots, d_n)$.

Obtenemos la siguiente proposición. En su demostración únicamente nos detendremos en las diferencias de la misma respecto a la dada para el caso de dominios prefijados.

Proposición 4.5.2. *La Σ_{imp}^{eq}-álgebra parcial oculta A_f^{can} es objeto final de $\mathcal{C}_{fimp}$*

Demostración. Sea un álgebra $B \in \mathcal{C}_{fimp}$. Entonces, por definición de $\mathcal{C}_{fimp}$, para cada $b \in B_{imp_{\Sigma^{eq}}}$ las funciones parciales definidas por $\omega_{B^b}(d_1, \dots, d_n) := imp\_\omega_B(b, d_1, \dots, d_n)$, con dominio de definición $Def(\omega_{B^b}) := \{(d_1, \dots, d_n) \in D_1 \times \cdots \times D_n \mid (b, d_1, \dots, d_n) \in Def(imp\_\omega_B)\}$, para cada operación $\omega : s_1 \dots s_n \to s \in \Omega^{eq}$, $n \geq 0$, constituyen, junto con D, un objeto de $\mathcal{C}_f$.

Ahora, la función total $h^{can}\colon B_{imp_{\Sigma^{eq}}} \to A_{fimp_{\Sigma^{eq}}}^{can}$ que se define para cada $b \in B_{imp_{\Sigma^{eq}}}$, por $h^{can}(b) := (\omega_{B^b}, Def(\omega_{B^b}))_{\omega\in\Omega^{eq}}$, determina un Σ_{imp}^{eq}-homomorfismo oculto fuerte. Para cada $imp\_\sigma\colon imp_{\Sigma^{eq}} s_1 \dots s_n \to s \in \Omega_{imp}^{eq}$, tenemos que $(b, d_1, \dots, d_n) \in Def(imp\_\sigma_B)$ si y sólo si $(d_1, \dots, d_n) \in Def(\sigma_{B^b})$ lo que es equivalente a que $(h^{can}(b), d_1, \dots, d_n) \in Def(imp\_\sigma_{A_f^{can}})$.

Además, este morfismo es único. Si existe otro Σ_{imp}^{eq}-homomorfismo oculto fuerte entre B y A_f^{can}, esté quedará determinado por una función total $h\colon B_{imp_{\Sigma^{eq}}} \to A_{fimp_{\Sigma^{eq}}}^{can}$ tal que, para cada $b \in B_{imp_{\Sigma^{eq}}}$, $h(b) := (f_\delta, Def(f_\delta))_{\delta\in\Omega^{eq}} \in A_{fimp_{\Sigma^{eq}}}^{can}$. Entonces, para cada operación $imp\_\omega\colon imp_{\Sigma^{eq}} s_1 \dots s_n \to s \in \Omega_{imp}^{eq}$, tenemos que $(b, d_1, \dots, d_n) \in Def(imp\_\omega_B)$ si y sólo si $((f_\delta, Def(f_\delta))_{\delta\in\Omega^{eq}}, d_1, \dots, d_n) \in Def(imp\_\omega_{A_f^{can}})$ por ser h homomorfismo fuerte. Por tanto, $(d_1, \dots, d_n) \in Def(\omega_{B^b})$ si y sólo si $(d_1, \dots, d_n) \in Def(f_\omega)$ y concluimos que los dominios de definición de ambas tuplas de funciones son iguales. ■

Debemos observar que el caso en el que los dominios de definición de las operaciones están prefijados es un caso particular de éste. No obstante, si no prefijamos los dominios de definición podemos obtener ejemplos en los que los miembros de la familia correspondan con estructuras definidas parcialmente sobre conjuntos distintos. En el siguiente ejemplo presentamos una de estas familias.

Ejemplo 4.5.3. Suponemos que queremos especificar la categoría discreta $\mathcal{T}$ formada por sólo dos objetos que correspondan al conjunto de los números naturales con la operación siguiente y al mismo conjunto con la operación predecesor. Entonces, una signatura Σ que permite especificar los elementos de la anterior categoría viene dada por un único género s y una única operación $\omega\colon s \to s$. El dominio de datos que fijamos para esta signatura es $D_s = \mathbb{N}$. En este caso no vamos a establecer igualdades sobre este dominio (exigiremos que la igualdad sea la literal) por lo que no necesitamos construir la signatura Σ^{eq}. De esta forma, trabajamos con la categoría de álgebras $PAlg^D(\Sigma)$. De toda esta categoría nos restringimos a la subcategoría $\mathcal{C}$ con dos únicas álgebras: A_1 que está definida por $Def(\omega_{A_1}) = \mathbb{N}$ y $\omega_{A_1}(n) = n+1$, para todo $n \in \mathbb{N}$; y A_2 que está definida por $Def(\omega_{A_2}) = \mathbb{N}^*$ y $\omega_{A_2}(n) = n-1$, para todo $n \in \mathbb{N}^*$.

Ahora, si construimos la signatura oculta Σ_{imp} y la correspondiente subcategoría $\mathcal{C}_{imp}$ de $HPAlg^D(\Sigma_{imp})$, podemos considerar un álgebra B de $\mathcal{C}_{imp}$ que recoge la familia de dos álgebras que forman A_1 y A_2 de la siguiente manera. El soporte sobre el género destacado es el conjunto formado por dos elementos $\{+,-\}$, y la única función parcial tiene como dominio de definición $(\{+\}\times\mathbb{N})\cup(\{-\}\times\mathbb{N}^*)$ y está definida por $imp\_\omega_B(+,n) := n+1$, para todo $n \in \mathbb{N}$, e $imp\_\omega_B(-,n) := n-1$, para todo $n \in \mathbb{N}^*$. $\square$

La posibilidad de disponer de dominios de definición distintos en cada álgebra permite pensar en modelados alternativos para las estructuras que hemos presentado anteriormente, sin necesidad de recurrir a las álgebras con igualdad.

Ejemplo 4.5.4. Podemos obtener un álgebra parcial que represente a la familia de los subgrupos de $\mathbb{Z}$ si trabajamos sobre la categoría $PAlg^D(\mathtt{GRP})$, con $\mathtt{GRP}$ la signatura para grupo y $\mathbb{Z}$ como dominio de datos para el único género. Esta álgebra es una $\mathtt{GRP}_{imp}$-álgebra G que esta definida por el soporte $\mathbb{N}$ para el género destacado y las operaciones:

- $imp\_prd_G(n, d1, d2) = d1 + d2$ con $Def(imp\_prd_G) = \{(n, d_1, d_2) \mid n \in \mathbb{N},\ d_1, d_2 \in n\mathbb{Z}\}$,
- $imp\_inv_G(n, d) = -d$ con $Def(imp\_inv_G) = \{(n, d) \mid n \in \mathbb{N},\ d \in n\mathbb{Z}\}$,
- $imp\_unt_G(n) = 0$.

Para cada $n \in \mathbb{N}$ fijado, obtenemos una $\mathtt{GRP}$-álgebra equivalente a $n\mathbb{Z}$ sin más que restringirnos al dominio de definición de las operaciones. No obstante, debemos observar que no obtenemos directamente una representación del grupo $n\mathbb{Z}$, ya que el conjunto soporte del álgebra es todo $\mathbb{Z}$ a diferencia de lo que conseguíamos con las álgebras con igualdad, donde el soporte del álgebra es $n\mathbb{Z}$. $\square$

4.5.3.2 Trabajar con homomorfismos débiles

Si no fijamos los dominios de definición y consideramos nuestros modelos dentro de la categoría $HPAlg^D(\Sigma^{eq}_{imp})$, cuyos morfismos son los homomorfismos débiles ocultos, entonces el objeto canónico que hemos definido en la subcategoría plena $\mathcal{C}_{imp}$ deja de ser objeto final. Esto es debido a que el homomorfismo que utilizamos para probar la finalidad de este objeto no es único, como demuestra el siguiente contraejemplo.

Ejemplo 4.5.5. Tomamos como categoría de partida la categoría discreta formada por dos objetos: el conjunto de los números naturales con la aplicación identidad y este mismo conjunto con la identidad restringida a $\mathbb{N}^*$. Es decir, $\mathcal{T} = \{\langle \mathbb{N}, (id_{\mathbb{N}} \colon \mathbb{N} \to \mathbb{N}, Def(id_{\mathbb{N}}) = \mathbb{N})\rangle, \langle \mathbb{N}, (id_{\mathbb{N}^*} \colon \mathbb{N} \to \mathbb{N}, Def(id_{\mathbb{N}^*}) = \mathbb{N}^*)\rangle\}$.

Una signatura adecuada para especificar los objetos de esta categoría es la misma signatura Σ y el mismo dominio de datos que hemos utilizado en el Ejemplo 4.5.3. Tampoco necesitamos en esta ocasión construir la signatura con igualdad correspondiente (la única igualdad considerada es la literal). Ahora nos restringimos a la subcategoría $\mathcal{C}$ de $PAlg^D(\Sigma)$ con dos únicos objetos que corresponden a las dos álgebras de $\mathcal{T}$.

Nos entretenemos un momento en citar las características tanto de la signatura Σ_{imp} como de la categoría $\mathcal{C}_{imp}$. La signatura oculta Σ_{imp} tiene dos géneros s, imp_{Σ}, y una única operación $imp\_\omega \colon imp_{\Sigma} s \to s$, además de las constantes visibles. Si continuamos con nuestra construcción tomando una subcategoría $\mathcal{C}_{imp}$ plena de $HPAlg^D(\Sigma_{imp})$ (categoría cuyos morfismos son homomorfismos débiles), tenemos que $\mathcal{C}_{imp}$ está formada por las álgebras B de $HPAlg^D(\Sigma_{imp})$ tal que para todo $b \in B_{imp_{\Sigma}}$, la Σ-álgebra B^b dada por $B^b_s = \mathbb{N}$ y $\omega_{B^b}(x) := \omega_B(b, x)$, con $Def(\omega_{B^b}) = \{x \in \mathbb{N} \mid (b, x) \in Def(\omega_B)\}$, es un objeto de $\mathcal{C}$. Es decir ω_{B^b} es $id_{\mathbb{N}}$ o $id_{\mathbb{N}^*}$.

Ahora, el objeto canónico A^{can} en $\mathcal{C}_{imp}$ es la Σ_{imp}-álgebra cuyo soporte sobre el género destacado es $A^{can}_{imp_{\Sigma}} = \{(f, Def(f)) \mid \langle \mathbb{N}, (f, Def(f))\rangle \in \mathcal{C}\}$; es decir, en este caso, $A^{can}_{imp_{\Sigma}} = \{(id_{\mathbb{N}}, \mathbb{N}), (id_{\mathbb{N}^*}, \mathbb{N}^*)\}$. La única operación de A^{can} está definida de la siguiente forma: $Def(imp\_\omega_{A^{can}}) = \{((f, Def(f)), x) \in A^{can}_{imp_{\Sigma}} \times \mathbb{N} \mid x \in Def(f)\}$, es decir, $Def(imp\_\omega_{A^{can}}) = \{(id_{\mathbb{N}}, \mathbb{N}) \times \mathbb{N}, (id_{\mathbb{N}^*}, \mathbb{N}^*) \times \mathbb{N}^*\}$, y para estos elementos $imp\_\omega_{A^{can}}((f, Def(f)), x) = f(x)$.

Sin embargo, vamos a comprobar que el objeto anterior no es final en $\mathcal{C}_{imp}$. En efecto, podemos definir dos morfismos de A^{can} en sí mismo. Por un lado tenemos el morfismo identidad. Por otro lado, definimos el Σ_{imp}-homomorfismo oculto débil h dado por la aplicación $h_{imp_{\Sigma}} \colon A^{can}_{imp_{\Sigma}} \to A^{can}_{imp_{\Sigma}}$ tal que $h_{imp_{\Sigma}}((id_{\mathbb{N}}, \mathbb{N})) = (id_{\mathbb{N}}, \mathbb{N})$ y $h_{imp_{\Sigma}}((id_{\mathbb{N}^*}, \mathbb{N}^*)) = (id_{\mathbb{N}}, \mathbb{N})$.

Esta aplicación realmente determina un Σ_{imp}-homomorfismo oculto débil. Para la única operación oculta $imp\_\omega \colon imp_{\Sigma} g \to g$, se verifica la condición sobre los dominios de definición. Como $Def(imp\_\omega_{A^{can}}) = \{(id_{\mathbb{N}}, \mathbb{N}) \times \mathbb{N}, (id_{\mathbb{N}^*}, \mathbb{N}^*) \times \mathbb{N}^*\}$. Dado $((id_{\mathbb{N}}, \mathbb{N}), x)$, con $x \in \mathbb{N}$, se tiene que $(h_{imp_{\Sigma}}((id_{\mathbb{N}}, \mathbb{N})), x) = ((id_{\mathbb{N}}, \mathbb{N}), x)$, o dado $((id_{\mathbb{N}^*}, \mathbb{N}^*), x)$, con $x \in \mathbb{N}^*$, entonces $(h_{imp_{\Sigma}}((id_{\mathbb{N}^*}, \mathbb{N}^*)), x) = ((id_{\mathbb{N}}, \mathbb{N}), x)$. Ambos casos pertenecen a $Def(imp\_\omega_{A^{can}})$. Además, también se verifica la condición de homomorfismo. Para cada $((f, Def(f)), x) \in Def(imp\_\omega_{A^{can}})$, $imp\_\omega_{A^{can}}(h_{imp_{\Sigma}}((f, Def(f)), x) = imp\_\omega_{A^{can}}((id_{\mathbb{N}}, \mathbb{N}), x) = id_{\mathbb{N}}(x) = x = imp\_\omega_{A^{can}}((f, Def(f)), x)$, así $h_{imp_{\Sigma}}$ conmuta con la única operación de A^{can}. $\square$

La idea subyacente es que, cuando trabajamos con homomorfismos ocultos débiles (sin prefijar los dominios de definición), perdemos una propiedad que estaba presente en todo nuestro estudio anterior y que no ha sido mencionada hasta ahora. Esta propiedad dice que la restricción de un homomorfismo oculto a los género visibles es el homomorfismo identidad para el dominio de datos. Esta condición sobre los morfismos puede expresarse en general en nuestra construcción *imp* de una manera bastante natural:

> Si A y B son Σ^{eq}_{imp}-álgebras de $\mathcal{C}_{imp}$, un Σ^{eq}_{imp}-homomorfismo oculto débil, $f\colon A \to B$, $f = (f_{imp_{\Sigma^{eq}}}, (id_{D_s})_{s\in S^{eq}})$, de $HPAlg^{D,dom}(\Sigma^{eq}_{imp})$ es un morfismo de $\mathcal{C}_{imp}$ si para todo $a \in A_{imp_{\Sigma^{eq}}}$, $(id_{D_s})_{s\in S^{eq}}\colon A^a \to B^{f_{imp_{\Sigma^{eq}}}(a)}$ es un morfismo de $\mathcal{C}$.

Esta condición no se satisface en el ejemplo anterior. Observamos que nuestra categoría $\mathcal{C}$ es una categoría discreta, por lo que realmente sólo se dispone del homomorfismo identidad y únicamente podemos garantizar que los dominios de definición de las operaciones de A^a coincidan con los de las operaciones de $B^{f_{imp_{\Sigma^{eq}}}(a)}$ si trabajamos con homomorfismos fuertes.

Si imponemos esta condición sobre los morfismos de $\mathcal{C}_{imp}$, que obviamente dejará de ser una subcategoría plena, recuperamos la propiedad de ser objeto final para el caso en el que trabajemos con homomorfismos ocultos débiles.

4.6 Modelado de relaciones entre diferentes estructuras algebraicas

Hasta ahora sólo nos hemos preocupado de especificar por separado las estructuras de datos presentes en EAT. No obstante, en EAT estas estructuras no viven solas, sino que existen operadores que permiten establecer relaciones entre ellas. En particular, estudiaremos un tipo de operadores muy importantes, aquéllos que permiten construir una estructura a partir de otra. Un ejemplo de este tipo es el operador que construye el complejo de cadenas asociado a un conjunto simplicial. Las relaciones entre estructuras algebraicas corresponden con la restricción a objetos de funtores entre las correspondientes categorías de estructuras. En esta sección trataremos de extender nuestros resultados previos de tal modo que estos funtores entre categorías puedan ser también modelados. Dicho modelado constituye el bloque central del artículo [28].

Partimos de dos categorías $\mathcal{T}^1$ y $\mathcal{T}^2$, y un funtor entre ellas $F\colon \mathcal{T}^1 \to \mathcal{T}^2$ (la construcción que pretendemos modelar). Suponemos que ambas categorías admiten una modelización siguiendo la técnica descrita en las secciones anteriores. Para ello retomamos el hecho de tener un dominio de definición prefijado, tal y como se hace en [28], ya que es suficiente para nuestros propósitos. No obstante, podemos hacer un desarrollo paralelo sin tener estos dominios prefijados trabajando con homomorfismos fuertes o bien con homomorfismos débiles incluyendo condiciones parecidas a las presentes en el comentario final de la sección anterior. Así, disponemos, para $i = 1, 2$, de una signatura

$\Sigma^i = (S^i, \Omega^i)$, un dominio de datos D^i junto con un dominio de definición dom^i, y una subcategoría $\mathcal{C}^i$ de $P\tilde{A}lg^{D^i,dom^i}(\Sigma^{i,eq})$, tal que cada A objeto de $\mathcal{C}^i$ verifica que el cociente $A/_{eq^i_A}$ define un objeto de $\mathcal{T}^i$.

Además, exigimos que el funtor F cumpla que, para cada $\Sigma^{1,eq}$-álgebra A de $\mathcal{C}^1$, el $\mathcal{T}^2$-objeto $F(A/_{eq^1_A})$ pueda ser expresado como $B/_{eq^2_B}$ para alguna $\Sigma^{2,eq}$-álgebra B de $\mathcal{C}^2$. En otras palabras, estamos asumiendo que el funtor $F\colon \mathcal{T}^1 \to \mathcal{T}^2$ pueda elevarse al nivel de la representación dando lugar a una aplicación $\tilde{F}\colon Obj(\mathcal{C}^1) \to Obj(\mathcal{C}^2)$ que hace conmutativo el siguiente diagrama sobre objetos:

$$\begin{array}{ccc}
Obj(\mathcal{C}^1) & \xrightarrow{\tilde{F}} & Obj(\mathcal{C}^2) \\
{\scriptstyle /_{eq^1}}\big\downarrow & & \big\downarrow{\scriptstyle /_{eq^2}} \\
\mathcal{T}^1 & \xrightarrow[F]{} & \mathcal{T}^2
\end{array}$$

En general, esta condición dará lugar a una restricción sobre el dominio de datos D^2, como veremos posteriormente en los ejemplos.

De la misma forma que para realizar el modelado de los objetos de una categoría $\mathcal{T}$, nos preocupábamos de modelar los objetos de una categoría intermediaria $\mathcal{C}$, de modo que los objetos de la categoría inicial se obtienen por paso al cociente, el modelado de la aplicación sobre objetos del funtor F se realizará a partir del modelado de una aplicación auxiliar $\tilde{F}$, que actúe sobre los objetos de las categorías auxiliares. Nuevamente, el funtor F se obtendrá tras un posterior paso al cociente.

Bajo las condiciones anteriores definimos una signatura Σ que es la unión de $\Sigma^{1,eq}_{imp}$ y $\Sigma^{2,eq}_{imp}$, (signaturas construidas a partir de $\Sigma^{1,eq}$ y $\Sigma^{2,eq}$ a través de la operación imp), junto con una operación $\rho\colon imp_{\Sigma^{1,eq}} \to imp_{\Sigma^{2,eq}}$ entre los géneros distinguidos de ambas signaturas. La signatura Σ se define como una signatura oculta con dos géneros ocultos $imp_{\Sigma^{1,eq}}$ y $imp_{\Sigma^{2,eq}}$. El álgebra de datos está formada por las constantes que provienen del dominio de datos $D = D^1 \cup D^2$.

Debemos notar que las dos signaturas anteriores no son disjuntas: por ejemplo coinciden en el género *bool*. Para evitar entrar en conflicto con los nombres de los géneros de ambas signaturas, renombraremos un género si aparece en las dos signaturas con dominio de definición distinto. Además, consideraremos que los géneros ocultos son siempre distintos (renombraremos en caso contrario), por lo que no se produce conflictos entre las operaciones de las dos signaturas, ya que todos ellos presentan a uno u otro de estos géneros ocultos. No obstante, no pretendemos entrar por ahora en problemas como el polimorfismo de operadores, que por otro lado no está presente en EAT. Por lo tanto, también renombraremos una operación si aparece con el mismo nombre en ambas signaturas.

Esta situación es nueva con respecto a lo que hemos presentado en los capítulos previos y también respecto al trabajo realizado en [72]. En esta signatura aparecen por primera vez dos géneros ocultos y también operaciones constructoras. Naturalmente, este tipo de signaturas están incluidas en la teoría general de especificaciones ocultas [42], aunque no en el caso particular de signatura obtenida a partir de la construcción *imp* que en este trabajo hemos tratado.

El ambiente en el que nos situamos a partir de la signatura Σ es la categoría $HPAlg^{D,dom}(\Sigma)$. Esta categoría tiene como objetos las Σ-álgebras parciales ocultas definidas sobre el dominio de datos $D = D^1 \cup D^2$. El dominio de definición para las operaciones deconstructoras viene definido como antes a partir de $dom = dom^1 \cup dom^2$, mientras que la operación ρ es considerada total. Los morfismos de esta categoría son los Σ-homomorfismos débiles ocultos. No obstante, al trabajar sobre dominios prefijados y ser ρ total son también fuertes.

Un morfismo $h\colon A \to B$ de $HPAlg^{D,dom}(\Sigma)$ entre dos Σ-álgebras A y B de esta categoría, está determinado por dos aplicaciones totales:

$$h_{imp_{\Sigma^{i,eq}}} : A_{imp_{\Sigma^{i,eq}}} \to B_{imp_{\Sigma^{i,eq}}}, i = 1, 2,$$

una para cada género oculto, tal que verifique:

- para cada operación $imp\_\omega\colon imp_{\Sigma^{i,eq}} s_1 \ldots s_n \to s$, con $i = 1, 2$ y cada tupla $(a, d_1, \ldots, d_n) \in Def(imp\_\omega_A)$

$$imp\_\omega_A(a, d_1, \ldots, d_n) = imp\_\omega_B(h_{imp_{\Sigma^{i,eq}}}(a), d_1, \ldots, d_n),$$

- y para cada $a \in A_{imp_{\Sigma^{1,eq}}}$

$$h_{imp_{\Sigma^{2,eq}}}(\rho_A(a)) = \rho_B(h_{imp_{\Sigma^{1,eq}}}(a)).$$

Como es de esperar, no nos interesan todas las álgebras de $HPAlg^{D,dom}(\Sigma)$. Consideramos una subcategoría plena $\mathcal{C}$ de $HPAlg^{D,dom}(\Sigma)$ tal que dada un álgebra A de $\mathcal{C}$ se tiene que:

- Las restricciones (siendo precisos, los reductos para las inclusiones de signaturas) de A a $\Sigma^{1,eq}_{imp}$ y a $\Sigma^{2,eq}_{imp}$ son objetos de $\mathcal{C}^1_{imp}$ y $\mathcal{C}^2_{imp}$ respectivamente; es decir, $A|\Sigma^{i,eq} \in \mathcal{C}^i_{imp}$, $i = 1, 2$. Denotaremos $A|\Sigma^{i,eq}$ por $A(i)$, $i = 1, 2$.
- Para cada elemento $a \in A_{imp_{\Sigma^{1,eq}}}$, $A(2)^{\rho_A(a)} = \tilde{F}(A(1)^a)$.

Recordamos que, para $b \in A_{imp_{\Sigma^{i,eq}}}$, $A(i)^b$ es la $\Sigma^{i,eq}$-álgebra parcial de $\mathcal{C}^i$ que definimos por las funciones parciales $\omega_{A(i)^b}(d_1, \ldots, d_n) := imp\_\omega_{A(i)}(b, d_1, \ldots, d_n)$ y $Def(\omega_{A(i)^b}) = dom_\omega$, para cada $\omega\colon s_1 \ldots s_n \to s$ de $\Sigma^{i,eq}$. Entonces, la segunda condición exige que la interpretación de la operación ρ sea exactamente la prestada de $\tilde{F}$. Es decir, el elemento $\rho_A(a) \in A_{imp_{\Sigma^{2,eq}}}$ determina el objeto de $\mathcal{C}^2$ que se obtiene al aplicar $\tilde{F}$ al álgebra de $\mathcal{C}^1$ obtenida a partir de a.

Si A y B son Σ-álgebras de $\mathcal{C}$, un Σ-homomorfismo oculto débil $h\colon A \to B$, dado por $h = (h_{imp_{\Sigma^{1,eq}}}, h_{imp_{\Sigma^{2,eq}}}, (id_s)_{s\in S\setminus\{imp_{\Sigma^{1,eq}}, imp_{\Sigma^{2,eq}}\}})$, obviamente verifica que $h^i = (h_{imp_{\Sigma^{i,eq}}}, (id_s)_{s\in S^{i,eq}})$ es un $\Sigma^{i,eq}_{imp}$-homomorfismo de $\mathcal{C}^i_{imp}$, $i = 1, 2$. Así, todos los morfismos de la categoría cumplen esta condición esperable y podemos considerar nuestra subcategoría plena.

En estas condiciones podemos de nuevo definir un objeto canónico en $\mathcal{C}$, a partir de la unión de los objetos finales de $\mathcal{C}^1_{imp}$ y $\mathcal{C}^2_{imp}$. Dicho objeto resultará ser final en $\mathcal{C}$ y, desde el punto de vista de la implementación, lo importante es que corresponde con la forma en que este tipo de trasformaciones entre estructuras de datos fueron implementados en EAT.

4.6.1 Objeto final

Sean la signatura $\Sigma = (S, \Omega)$, el dominio de datos D y el dominio de definición *dom* tal y como han sido definidos en la sección anterior, así como la subcategoría $\mathcal{C}$ de $HPAlg^{D,dom}(\Sigma)$. Sea $A^{i,can}$ el objeto final que obtuvimos para la categoría $\mathcal{C}^i_{imp}$, $i = 1, 2$. Entonces, podemos definir un objeto de $\mathcal{C}$, que denotamos de nuevo por A^{can}, a partir de estos dos objetos finales, junto con una interpretación de ρ que hace corresponder a cada tupla de funciones que representa a un objeto O de $\mathcal{C}^1$, la tupla de funciones que representa al objeto $\tilde{F}(O)$ de $\mathcal{C}^2$. De forma más precisa, definimos el álgebra A^{can} como la Σ-álgebra que verifica:

- $A^{can}|\Sigma^{i,eq}_{imp} = A^{i,can}$, $i = 1, 2$,
- para cada $(f_{\omega^1})_{\omega^1\in\Omega^{1,eq}} \in A^{can}_{imp_{\Sigma^{1,eq}}}$, se define $\rho_{A^{can}}((f_{\omega^1})_{\omega^1\in\Omega^{1,eq}}) = (g_{\omega^2})_{\omega^2\in\Omega^{2,eq}}$, siendo $\langle D^2, (g_{\omega^2})_{\omega^2\in\Omega^{2,eq}}\rangle = \tilde{F}(\langle D^1, (f_{\omega^1})_{\omega^1\in\Omega^{1,eq}}\rangle)$.

De la propia definición se obtiene que este objeto pertenece a la categoría $\mathcal{C}$. El siguiente resultado demuestra que A^{can} es objeto final de $\mathcal{C}$.

Teorema 4.6.1. *La Σ-álgebra A^{can}, cuyos reductos sobre $\Sigma^{1,eq}_{imp}$ y $\Sigma^{2,eq}_{imp}$ corresponden con los objetos finales en $\mathcal{C}^1_{imp}$ y $\mathcal{C}^2_{imp}$, es objeto final en $\mathcal{C}$.*

Demostración. Sea B un objeto de $\mathcal{C}$. Entonces, por definición de la subcategoría $\mathcal{C}$ se tiene que:

· $B|\Sigma^{i,eq}_{imp} \equiv B(i) \in \mathcal{C}^i_{imp}$, para $i = 1, 2$,

· para cada $b \in B_{imp_{\Sigma^{1,eq}}}$, $B(2)^{\rho_B(b)} = \tilde{F}(B(1)^b)$.

Como $B(i) \in \mathcal{C}^i_{imp}$ y $A^{i,can}$ es el objeto final de esta categoría, existe un único $\Sigma^{i,eq}_{imp}$-homomorfismo oculto débil de $\mathcal{C}^i_{imp}$ entre $B(i)$ y $A^{i,can}$, $i = 1, 2$. Suponemos que este homomorfismo está determinado por la función total en el género oculto $h^{i,can}\colon B(i)_{imp_{\Sigma^i,eq}} \to A^{i,can}_{imp_{\Sigma^i,eq}}$, con $i = 1, 2$.

Podemos definir entonces el único morfismo entre B y A^{can} en $\mathcal{C}$ que determinado por la pareja de funciones totales $(h^{1,can}, h^{2,can})$.

Para demostrar que este morfismo verifica la propiedad de homomorfismo débil es suficiente con comprobar la propiedad de homomorfismo sobre la operación ρ. Para el resto de las operaciones se obtiene de forma inmediata por ser $h^{i,can}$, $i = 1, 2$ un homomorfismo débil en la correspondiente categoría. Es decir, debemos probar si para cada $b \in B_{imp_{\Sigma^1,eq}}$

$$h^{2,can}(\rho_B(b)) = \rho_{A^{can}}(h^{1,can}(b)).$$

Por un lado tenemos que $h^{1,can}(b) = (\omega^1_{B(1)^b})_{\omega^1 \in \Omega^{1,eq}}$, por definición de $h^{1,can}$, y, por definición de A^{can}, $\rho_{A^{can}}((\omega^1_{B(1)^b})_{\omega^1 \in \Omega^{1,eq}}) = (f_{\omega^2})_{\omega^2 \in \Omega^{2,eq}}$, tal que $\langle D^2, (f_{\omega^2})_{\omega^2 \in \Omega^{2,eq}} \rangle = \tilde{F}(\langle D^1, (\omega^1_{B(1)^b})_{\omega^1 \in \Omega^{1,eq}} \rangle)$.

Por otro lado tenemos que $h^{2,can}(\rho_B(b)) = (\omega^2_{B(2)^{\rho_B(b)}})_{\omega^2 \in \Omega^{2,eq}}$, por definición de $h^{2,can}$, y $\langle D^2, (\omega^2_{B(2)^{\rho_B(b)}})_{\omega^2 \in \Omega^{2,eq}} \rangle = \tilde{F}(\langle D^1, (\omega^1_{B(1)^b})_{\omega^1 \in \Omega^{1,eq}} \rangle)$, por definición de B. Entonces $(f_{\omega^2})_{\omega^2 \in \Omega^{2,eq}} = (\omega^2_{B(2)^{\rho_B(b)}})_{\omega^2 \in \Omega^{2,eq}}$, por lo que obtenemos la igualdad buscada.

La unicidad del morfismo se demuestra a partir de la unicidad de los morfismos finales $h^{i,can}$ en las categorías $\mathcal{C}^i_{imp}$, $i = 1, 2$. Sea otro morfismo entre B y A^{can} en $\mathcal{C}$ determinado por la pareja de funciones totales (h^1, h^2), con $h^i\colon B(i)_{imp_{\Sigma^i,eq}} \to A^{i,can}_{imp_{\Sigma^i,eq}}$, $i = 1, 2$. Entonces, en particular h^i es morfismo en $\mathcal{C}^i_{imp}$. Pero, $h^{i,can}$ es final en esta categoría, por lo que $h^{i,can} = h^i$, $i = 1, 2$. ■

4.6.2 Ejemplos

Para ilustrar la potencia práctica de esta última construcción teórica, en esta sección incluimos tres ejemplos de su aplicación. En el primero desarrollamos el funtor olvido entre grupos y semigrupos. En este ejemplo nos situaremos en un contexto “rígido” (en el que trabajaremos con álgebras totales y no permitiremos definir equivalencias en sus soportes). En el segundo, estudiamos la construcción anillo de grupo. En esta ocasión permitiremos igualdades distintas de las literales aunque las álgebras seguirán siendo totales. El tercer ejemplo lo dedicaremos a representar la construcción que define el complejo de cadenas asociado a un conjunto simplicial. Esta construcción está presente en EAT y para describirla formalmente necesitamos incorporar la parcialidad a nuestros modelos y la posibilidad de establecer cocientes en los mismos.

4.6.2.1 Semigrupo de un grupo

Sean $SGRP$ la categoría de los semigrupos, GRP la categoría de los grupos y $F\colon GRP \to SGRP$ el funtor olvido entre ambas que asigna a cada grupo el semigrupo que se obtiene prescindiendo de la operación inversa y el elemento neutro del grupo.

Comenzamos recordando la modelización de ambas categorías por separado. Para ello utilizamos la técnica presentada en las secciones anteriores.

Para GRP rescatamos la signatura de grupo $\mathtt{GRP}$ y fijamos un dominio de datos $D_g^{GRP} = X$, con X un conjunto cualquiera. Así, debemos restringimos a los grupos que se pueden construir sobre este conjunto X, es decir, nos quedamos con la subcategoría $\mathcal{C}^{GRP}$ de $Alg^{D^{GRP}}(\mathtt{GRP})$ cuyos objetos están en GRP.

Para modelar los semigrupos definimos la signatura $\mathtt{SGRP}$ con un único género sg y una única operación $+\colon sg \times sg \to sg$. Para ella es necesario fijar un dominio de datos $D_{sg}^{SGRP} = Y$, con Y un conjunto cualquiera. Nos restringimos entonces a una subcategoría $\mathcal{C}^{SGRP}$ de $Alg^{D^{SGRP}}(\mathtt{SGRP})$ cuyos objetos están en $SGRP$.

En este caso, como no establecemos equivalencias dentro de los soportes, tenemos que:

$$\tilde{F} := F|_{\mathcal{C}^{GRP}} \colon Obj(\mathcal{C}^{GRP}) \to Obj(\mathcal{C}^{SGRP}).$$

Entonces, para que el funtor F verifique la condición que hemos exigido a los funtores que vamos a tratar es necesario y suficiente que $D_{sg}^{SGRP} = D_g^{GRP}$, es decir, el grupo y el semigrupo deben compartir el mismo soporte.

Si aplicamos la operación imp a esas signaturas y esas categorías de álgebras, obtenemos una categoría C_{imp}^{GRP} de $\mathtt{GRP}_{imp}$-álgebras, y una categoría C_{imp}^{SGRP} de $\mathtt{SGRP}_{imp}$-álgebras. Ahora, para especificar el funtor F, construimos la signatura Σ que resulta de la unión de las signaturas $\mathtt{GRP}_{imp}$ y $\mathtt{SGRP}_{imp}$ junto con una operación $\rho\colon imp_{\mathtt{GRP}} \to imp_{\mathtt{SGRP}}$ entre los géneros destacados de ambas signaturas. Esta signatura es una signatura oculta con los anteriores géneros como únicos géneros ocultos. El dominio de datos D está formado por $D_g = D_{sg} = X$.

Entonces, consideramos la subcategoría $\mathcal{C}$ de $HAlg^D(\Sigma)$ tal que cada A objeto de $\mathcal{C}$, verifique que:

- para cada $a \in A_{imp_{\mathtt{GRP}}}$, la $\mathtt{GRP}$-álgebra $A(1)^a = \langle X, (imp\_prd_A(a,-), imp\_inv(a,-), imp\_unt(a))\rangle$ esté en $\mathcal{C}^{GRP}$, y para cada $b \in A_{imp_{\mathtt{SGRP}}}$, la $\mathtt{SGRP}$-álgebra $A(2)^b = \langle X, imp\_+_A(b,-)\rangle$ esté en $\mathcal{C}^{sgrp}$,
- para cada $a \in A_{imp_{\mathtt{GRP}}}$, $A(2)^{\rho_A(a)} = F(A(1)^a)$, es decir:

$$imp\_+_A(\rho_A(a),-) := imp\_prd_A(a,-)$$

Si aplicamos el teorema anterior a este caso particular obtenemos que el objeto canónico consta de tuplas para los géneros ocultos. En concreto, tuplas de tres funciones para el género destacado $imp_{\texttt{GRP}}$ que determinen grupos sobre X y tuplas de una única función para el género destacado $imp_{\texttt{SGRP}}$ que determinen semigrupos sobre X. La función que se asocia a ρ asigna a un elemento del conjunto para $imp_{\texttt{GRP}}$, la primera función correspondiente a la operación producto en la tupla. Nuestra construcción hace *explícito* que esta operación de proyección es un "olvido" de estructura.

Por ejemplo, si $X = \mathbb{Z}$, podemos representar cualquier grupo *numerable* provisto del semigrupo que se obtiene al olvidar las operaciones inversa y elemento neutro. Sin embargo, no es posible, en este contexto "rígido", codificar por ejemplo grupos finitos del mismo modo, si mantenemos $X = \mathbb{Z}$.

En el siguiente ejemplo abordamos una construcción algebraica más interesante como es la del funtor que asocia a un grupo, el anillo de grupo correspondiente.

4.6.2.2 Anillo de grupo

La siguiente definición de anillo de grupo ha sido extraída de [77].

Definición 4.6.2 (Anillo de grupo). Sean $\mathbb{Z}$ el anillo de los enteros y G un grupo. El *anillo de grupo* $\mathbb{Z}[G]$ se define como el conjunto de todas las sumas formales $\sum_{x\in G} r_x x$ con $r_x \in \mathbb{Z}$ y $r_x = 0$ para todo $x \in G$ salvo en un número finito de excepciones, junto con las operaciones de suma y multiplicación dadas por

$$\left(\sum_{x\in G} r_x x\right) + \left(\sum_{x\in G} r'_x x\right) = \sum_{x\in G}(r_x + r'_x)x$$

y

$$\left(\sum_{x\in G} r_x x\right)\left(\sum_{y\in G} r'_y y\right) = \sum_{z\in G}\left(\sum_{xy=z} r_x r'_y\right)z.$$

Por analogía con estructuras anteriores, en ocasiones nos referiremos a los elementos de un anillo de grupo como combinaciones y a los del grupo como generadores.

Como es habitual en la definición anterior estamos denotando de la misma forma al grupo G y al conjunto sobre el que está definido el grupo, aunque debemos tener presente que son cosas distintas. Resulta redundante señalar que si G es un grupo, $\mathbb{Z}[G]$ es un anillo.

En este ejemplo tratamos de especificar el funtor que asocia a cada grupo G el anillo de grupo $\mathbb{Z}[G]$. Este funtor se define sobre morfismos de forma natural (a partir de las imágenes de los generadores se obtienen las imágenes de las combinaciones).

Podemos trabajar con una definición de anillo de grupo más general. A partir de cualquier anillo con unidad R y cualquier grupo G se construye el anillo de grupo $R[G]$. Así, fijado un anillo con unidad R definimos análogamente el funtor que asocia a cada grupo G el anillo de grupo $R[G]$.

Sean GRP la categoría de los grupos, RNG la categoría de los anillos y $F\colon GRP \to RNG$ el funtor que asigna a cada grupo su anillo de grupo asociado.

Debemos comenzar dando modelos formales para grupos y anillos. En esta ocasión incluimos la posibilidad de realizar identificaciones sobre los soportes pero no permitiremos parcialidad. Por ello, no es necesario prefijar un dominio de definición.

Para la primera categoría necesitamos nuevamente la signatura de grupo $\mathtt{GRP}$, así como un dominio de datos $D_g^{GRP} = X$. Definimos la subcategoría $\mathcal{C}^{GRP^{eq}}$ de $\tilde{A}lg^{D^{GRP}}(\mathtt{GRP}^{eq})$ que está formada por las $\mathtt{GRP}^{eq}$-álgebras A tal que la $\mathtt{GRP}$-álgebra $\tilde{A}$ es un grupo con elementos tomados en X.

Similarmente, para modelizar los anillos, utilizamos la signatura $\mathtt{RNG}$ que tiene un único género r y las operaciones $+\colon r\times r \to r$, $-\colon r \to r$, $e\colon\ \to r$, $*\colon r\times r \to r$, junto con un conjunto de datos $D_r^{RNG} = Y$. Definimos la subcategoría $\mathcal{C}^{RNG^{eq}}$ de $\tilde{A}lg^{D^{RNG}}(\mathtt{RNG}^{eq})$ que está formada por las $\mathtt{RNG}^{eq}$-álgebras B tal que la $\mathtt{GRP}$-álgebra $\tilde{B}$ es un anillo con soporte de elementos en Y.

Ahora, para poder especificar la aplicación entre objetos dada por el funtor $F\colon GRP \to RNG$, debe verificarse que para cada álgebra A de $Obj(\mathcal{C}^{GRP^{eq}})$, el objeto $F(\tilde{A})$ de RNG pueda ser modelado a través de un álgebra B de $Obj(\mathcal{C}^{RNG^{eq}})$. Como X es el soporte del álgebra A para el género g, entonces el conjunto sobre el que se define el grupo $\tilde{A}$ es el cociente $X/_{eq_A^g}$. Así, el conjunto sobre el que se define el anillo $F(\tilde{A})$ es $\mathbb{Z}[X/_{eq_A^g}]$. Entonces, para que se verifique la anterior condición es suficiente considerar como soporte para la signatura de anillo el conjunto $D_r^{RNG} = \mathbb{Z}[X]$, es decir, el grupo abeliano libre sobre el conjunto X, de modo que al actuar la igualdad eq_B^r sobre este conjunto recuperemos el soporte anterior para el anillo. Con idea de aproximarnos a una posible implementación, tomaremos como soporte D_r^{RNG} el conjunto formado por listas de parejas de elementos, similar al que elegimos para las combinaciones de un complejo de cadenas en un determinado grado pero prescindiendo del elemento que marcaba el grado. Es decir, consideramos

$$D_r^{RNG} = \{\langle (r_1, x_1), \ldots, (r_n, x_n)\rangle \mid n \geq 0, r_i \in \mathbb{Z}, x_i \in X, i = 1, \ldots, n\},$$

de modo que se recoge un número finito de parejas que debe contener aquéllas en las que el coeficiente entero no es nulo. No se exige que todos los coeficientes sean no nulos, ni que los elementos de X sean distintos para poder definir más cómodamente las operaciones del anillo. Posteriormente la igualdad deberá realizar identificaciones que permitan recuperar la suma formal.

Si aplicamos ahora la operación imp a dichas signaturas y a sus correspondientes categorías de álgebras, obtenemos una categoría $C_{imp}^{GRP^{eq}}$ de $\mathtt{GRP}_{imp}^{eq}$-álgebras y una categoría $C_{imp}^{RNG^{eq}}$ de $\mathtt{RNG}_{imp}^{eq}$-álgebras. Entonces, para especificar el funtor F, consideramos la signatura Σ que es la unión de las signaturas $\mathtt{GRP}_{imp}^{eq}$ y $\mathtt{RNG}_{imp}^{eq}$ junto con una operación $\rho\colon imp_{\mathtt{GRP}^{eq}} \to imp_{\mathtt{RNG}^{eq}}$ entre los géneros destacados de ambas signaturas. Esta signatura es una signatura oculta con los dos anteriores como únicos géneros ocultos. El dominio de datos D está formado por los dominios de datos fijados en las dos signaturas.

Entonces, consideramos la subcategoría $\mathcal{C}$ de $HAlg^D(\Sigma)$ tal que cada A objeto de $\mathcal{C}$, verifique que:

- para cada $a \in A_{imp_{\mathtt{GRP}^{eq}}}$, la $\mathtt{GRP}^{eq}$-álgebra $A(1)^a = \langle X, \{imp\_prd_A(a,-), imp\_inv_A(a,-), imp\_unt_A(a), imp\_eq^g_A(a,-)\}\rangle$ esté en $\mathcal{C}^{GRP^{eq}}$, y para cada $b \in A_{imp_{\mathtt{RNG}^{eq}}}$, la $\mathtt{RNG}^{eq}$-álgebra $A(2)^b = \langle \mathbb{Z}[X], \{imp\_+_A(b,-), imp\_-_A(b,-), imp\_e_A(b), imp\_*_A(b,-), imp\_eq^r_A(b,-)\}\rangle$ esté en $\mathcal{C}^{RNG^{eq}}$,
- para cada $a \in A_{imp_{\mathtt{GRP}}}$, $A(2)^{\rho_A(a)} = \tilde{F}(A(1)^a)$.

Para que se dé la segunda condición anterior es suficiente exigir que las operaciones imp_+_A, imp_-_A y imp_e_A para el elemento $\rho_A(a)$ fijado, se definan como las operaciones correspondientes al grupo abeliano libre del complejo de cadenas, aprovechando la estructura que tienen los elementos de D_r^{RNG}. La operación de igualdad $imp\_eq^r_A$ para el elemento $\rho_A(a)$ fijado, se define similarmente a partir de la operación $imp\_eq^g_A$ para el elemento a fijado, esto es, se tienen igualdades entre combinaciones que se definan a partir de igualdades sobre generadores. Además, se utiliza esta igualdad para recuperar la suma formal a partir de la representación anterior de modo que se identifican dos combinaciones si se diferencian en el orden de sus monomios, una combinación que presente un elemento con coeficiente nulo con aquélla que no la presente y por último una combinación con dos monomios que comparten el mismo generador con la combinación que presente un único monomio con ese generador y coeficiente la suma (entera) de los coeficientes. Por último, la operación imp_*_A sobre el elemento $\rho_A(a)$ se define de la siguiente manera:

$$imp\_*_A(\rho_A(a), \langle (r_{x_1}, x_1), \ldots, (r_{x_n}, x_n)\rangle, \langle (r_{y_1}, y_1), \ldots, (r_{y_m}, y_m)\rangle) := \langle (r_{x_i} r_{y_j}, imp\_prd_A(a, x_i, y_j))_{i=1,\ldots,n, j=1,\ldots,m}\rangle.$$

Si aplicamos el teorema de existencia de objeto final, en este caso particular obtenemos un modo "universal" de representar la construcción anillo de grupo (sobre un conjunto X). Ésta consta de tuplas de cuatro funciones (incluida la igualdad) para el género destacado $imp_{\mathtt{GRP}^{eq}}$ que determinen un grupo sobre X, y de tuplas de cinco funciones para el género destacado $imp_{\mathtt{RNG}^{eq}}$ que determinen anillos sobre $\mathbb{Z}[X]$. La función correspondiente a ρ asigna a una tupla de funciones en $imp_{\mathtt{GRP}^{eq}}$ la tupla del anillo de grupo que se construye sobre el anterior grupo en $imp_{\mathtt{RNG}^{eq}}$.

Por ejemplo, si $X = \mathbb{Z}$, podemos representar cualquier grupo finito o numerable provisto de su correspondiente anillo de grupo.

4.6.2.3 Complejo de cadenas asociado a un conjunto simplicial

Como último ejemplo abordamos la representación del funtor que aplica a cada conjunto simplicial el complejo de cadenas que canónicamente tiene asociado. La siguiente definición ha sido tomada de [65].

Definición 4.6.3 (Complejo de cadenas asociado a un conjunto simplicial). Sea $K = \{K_n\}_{n\in\mathbb{N}}$ un conjunto simplicial con operaciones cara y degeneración $\{\delta_i^n\}_{n>0,i=1,\ldots,n}$, $\{\eta_i^n\}_{n\geq 0,i=1,\ldots,n}$. El *complejo de cadenas asociado a* K, que denotamos por $C(K)$, es el complejo de cadenas dado por $\{C(K)_n, d_n\}_{n\in\mathbb{Z}}$ donde, si $n \geq 0$, $C(K)_n = \mathbb{Z}[K_n]$ siendo $\mathbb{Z}[K_n]$ el $\mathbb{Z}$-módulo libre generado por K_n, y si $n < 0$, $C(K)_n = \emptyset$; la operación diferencial, para $n > 0$ se define a partir de la siguiente aplicación sobre los generadores

$$d_n := \sum_{i=0}^{n}(-1)^i\delta_i^n$$

(se completa al extenderse por linealidad para los elementos de $\mathbb{Z}[K_n]$), y para $n \leq 0$ se toma el homomorfismo nulo.

Sobre los morfismos, es claro como cada morfismo entre conjuntos simpliciales da lugar a un morfismo entre los correspondientes complejos de cadenas (otra vez basta extender a combinaciones una aplicación entre generadores).

Sean SS la categoría de los conjuntos simpliciales, CC la categoría de los complejos de cadenas y $F\colon SS \to CC$ el funtor que construye el complejo de cadenas asociado a un conjunto simplicial. Antes de tratar de modelar la aplicación a nivel de objetos correspondiente a este funtor, necesitamos construir representaciones para los conjuntos simpliciales y los complejos de cadenas. Estas representaciones han sido propuestas como ejemplos en las secciones anteriores. Para conseguirlas ha sido necesario utilizar tanto equivalencias dentro de los dominios como parcialidad en los modelos. Así, definimos dos signaturas $\Sigma^{ss,eq}$ y $\Sigma^{cc,eq}$, junto con los dominios de datos D^{ss}, D^{cc} y los dominios de definición dom^{ss} y dom^{cc}. Entonces, consideramos categorías $\mathcal{C}^{ss}$ y $\mathcal{C}^{cc}$ que recogen los conjuntos simpliciales y complejos de cadenas que se pueden definir sobre los dominios de definición a través de relaciones de equivalencia en los mismos.

Nuestro objetivo es dar una especificación de la restricción a objetos del funtor F, a través de una aplicación $\tilde{F}\colon Obj(\mathcal{C}^{ss}) \to Obj(\mathcal{C}^{cc})$. Para ello es necesario que se verifique que para cada álgebra A de $Obj(\mathcal{C}^{ss})$, el objeto $F(\tilde{A}) \in CC$ pueda ser modelado a través de un álgebra B de $Obj(\mathcal{C}^{cc})$. Debemos observar que la familia de conjuntos de generadores del complejo de cadenas $F(\tilde{A})$ viene dada por $D^{ss}_{asm}/_{eq^{asm}_A}$. Así, es suficiente con fijar como soporte para el género gnr en la signatura $\Sigma^{cc,eq}$ el conjunto de datos $D^{cc}_{gnr} := D^{ss}_{asm}$. Entonces, si definimos la igualdad sobre los generadores del complejo de cadenas como la igualdad sobre los símplices del conjunto simplicial, recuperamos el anterior soporte para los generadores del complejo de cadenas.

Ahora, definimos la signatura Σ que es la unión de las signaturas $\Sigma^{ss,eq}_{imp}$ y $\Sigma^{cc,eq}_{imp}$ junto con una operación $\rho\colon imp_{\Sigma^{ss,eq}} \to imp_{\Sigma^{cc,eq}}$ entre los géneros destacados de ambas signaturas. Esta signatura es una signatura oculta con los anteriores géneros como únicos géneros ocultos. El dominio de datos D para ella se construye a partir del conjunto de símplices geométricos $D^{ss}_{gsm} = K$ y el conjunto de generadores $D^{cc}_{gnr} = D^{ss}_{asm}$. La operación de grado para los generadores del complejo de cadenas se obtiene a partir de la operación de dimensión para los símplices abstractos en el conjunto simplicial y a partir de éstas se obtienen los dominio de definición de las operaciones. Debemos observar que, como consecuencia, los conjuntos de generadores para índices negativos son vacíos.

Entonces, consideramos la subcategoría $\mathcal{C}$ de $HPAlg^{D,dom}(\Sigma)$ tal que cada A objeto de $\mathcal{C}$, verifique que cada elemento de $A_{imp_{\Sigma^{ss,eq}}}$ sirva de índice para un objeto de $\mathcal{C}^{ss}$ y cada elemento de $A_{imp_{\Sigma^{cc,eq}}}$ sirva de índice para un objeto de $\mathcal{C}^{cc}$. Además, el elemento $\rho_A(a)$ debe indexar al objeto $\tilde{F}(A(1)^a)$ para cada $a \in A_{imp_{\Sigma^{ss,eq}}}$.

Para que se verifique la segunda condición anterior basta con definir la igualdad para los generadores en el elemento $\rho_A(a)$ como la igualdad para los símplices abstractos en el elemento a y, además, que la diferencial sobre una combinación en el elemento $\rho_A(a)$ esté definida a través de las operaciones caras en el elemento a según la fórmula $d_n := \sum_{i=0}^{n}(-1)^i \delta_i^n$. El resto de las operaciones para $\rho_A(a)$ están definidas a partir de éstas. Podemos expresar estas condiciones de un modo más explícito:

- $imp\_eq_A^{gnr}(\rho_A(a), x, y) := imp\_eq_A^{asm}(a, x, y)$, para todo $x, y \in dom^{cc}_{eq^{gnr}}$,
- $imp\_df_A(\rho_A(a), \langle n, [(1, x)]\rangle) :=$
  $imp\_add_A(\rho_A(a), imp\_mlt_A(\rho_A(a), (-1)^0, imp\_face_A(a, 0, n, x)),$
  $imp\_add_A(\rho_A(a), imp\_mlt_A(\rho_A(a), (-1)^1, imp\_face_A(a, 1, n, x)), \ldots,$
  $imp\_add_A(\rho_A(a), imp\_mlt_A(\rho_A(a), (-1)^n, imp\_face_A(a, n, n, x)) \ldots)),$
  para todo $x \in D^{cc}_{gnr}$ con $dim(x) = n$, $n \geq 0$, y se extiende por linealidad para el resto de elementos de D^{cc}_{cmb}.

EAT trabaja, por razones de eficiencia, con el complejo de cadenas normalizado $C^N(K)$ asociado a un conjunto simplicial K. Este complejo de cadenas tiene como generadores en cada grado exclusivamente a los símplices geométricos. Las operaciones diferenciales se definen a través de la suma formal de caras que se tiene en $C(K)$, excepto que se excluye de dicha suma los elementos en los que el resultado de la cara sea un símplice degenerado. Estos dos complejos de cadenas son equivalentes. Podemos encontrar una demostración de este resultado en [65].

Esta versión del complejo de cadenas normalizado asociado a un conjunto simplicial puede ser obtenida como una adaptación del modelo anterior. Para ello, el conjunto de datos asociado a D^{cc}_{gnr} debe ser D^{ss}_{gsm} y la igualdad sobre los generadores tiene que venir dada por la igualdad sobre los símplices geométricos. Entonces, simplemente debemos modificar la condición sobre la diferencial de modo que la suma únicamente incluya las caras que originen elementos no degenerados.

Si aplicamos de nuevo el teorema de finalidad en este caso particular tenemos un modo "universal" de representar la construcción complejo de cadenas normalizado asociado a un conjunto simplicial (sobre un conjunto de símplices geométricos K). Ésta consta de tuplas funciones para el género destacado $imp_{\Sigma^{ss,eq}}$ que determinen un conjunto simplicial sobre K, y de tuplas de funciones para el género destacado $imp_{\Sigma^{cc,eq}}$ que determinen un complejo de cadenas sobre $\mathbb{Z}[K]$. La función correspondiente a ρ asigna a una tupla de funciones en $imp_{\Sigma^{ss,eq}}$ la tupla del complejo de cadenas asociado en $imp_{\Sigma^{cc,eq}}$.

Recordamos que para tener determinado un conjunto simplicial es suficiente con tener una pareja de funciones $(fc', =_{gsm})$ que representen la operación cara y la igualdad, ambas sobre los símplices geométricos. Igualmente para tener determinado un complejo de cadenas es suficiente con definir una pareja de funciones $(d', =_{gnr})$ que representen la operación diferencial y la igualdad, ambas sobre los generadores. Así, podemos definir una nueva álgebra coherente con estos nuevos modelos, que presente como conjuntos soporte para los géneros destacados tuplas de funciones como las anteriores. La función que esta álgebra asocia a la aplicación ρ hace corresponder a una tupla $(fc', =_{gsm})$ la tupla $(d', =_{gnr})$ de modo que $=_{gnr}$ es exactamente $=_{gsm}$ y d' se define de la forma conocida a través de fc'. No es difícil entonces probar que esta álgebra es también objeto final dentro de la categoría $\mathcal{C}$. Esta álgebra corresponde exactamente con el modo en el que el operador que permite construir un complejo de cadenas a partir de un conjunto simplicial fue implementado en EAT (ver [81], página 42).

Capítulo 5

De EAT a Kenzo

5.1 Introducción

En este capítulo trataremos de realizar un estudio de las estructuras de datos presentes en el sistema Kenzo, de la misma forma que hemos hecho con EAT. El sistema Kenzo [34] es el descendiente de EAT, y ambos están dedicados al cálculo en Topología Algebraica. En particular los dos calculan grupos de homología de espacios de lazos iterados. Una diferencia importante entre ambos sistemas, aparte de que en Kenzo se utilizan algoritmos más eficientes, es que para implementar Kenzo se ha utilizado CLOS (Common Lisp Object System) que es la componente orientada a objetos de Common Lisp [93]. Podemos encontrar un estudio detallado del sistema Kenzo en [33] y el manual de usuario en [34]. Nosotros nos centraremos en el tratamiento que este programa da a sus estructuras de datos.

El enfoque orientado a objetos de Kenzo permite construir estructuras eficientes aprovechando las posibilidades que nos ofrecen herramientas como la herencia y el polimorfismo. La herencia es un concepto bastante escurridizo [100, 94, 66], abordable desde diferentes perspectivas (ver, por ejemplo, [37, 50, 52, 67, 55, 44, 12, 16]), lo que hace que su formalización sea una tarea complicada. En este capítulo nos centraremos en la parte de especificación de la herencia [92], continuando con la filosofía de los capítulos previos, sin entrar en cuestiones de implementación de la misma. Nuestra aproximación se basará en el marco conocido como *especificación con orden en los géneros* ("order-sorted specification" [44]), que es la técnica que habitualmente se ha utilizado en especificación oculta para modelar la herencia [41, 40]. No obstante, no nos ceñiremos a esta teoría, de modo que interpretaremos la herencia como un tipo de *coerción explícita* [100, 13, 10, 35, 37], en lugar de utilizar la interpretación habitual de la teoría oculta como inclusión entre los conjuntos soportes.

Además de la herencia, otro concepto importante en este capítulo es el de *polimorfismo* en las operaciones, que en una concepción amplia significa que un mismo identificador puede hacer referencia a operaciones diferentes dependiendo del contexto. No obstante, en la literatura este concepto ha recibido diversas interpretaciones. Nosotros nos quedaremos con las distintas clases de polimorfismo que observaron Cardelli y Wegner en [17]. Estos autores distinguen dos clases de polimorfismo: el polimorfismo *universal* y el polimorfismo *ad-hoc*. El primer grupo trabaja normalmente sobre un número infinito de tipos, los cuales deben tener una estructura común, mientras que las operaciones polimórficas ad-hoc trabajan solamente sobre un conjunto finito de tipos diferentes que, en principio, pueden no estar relacionados. Reciben el apellido de ad-hoc ya que podemos entenderlas como un conjunto pequeño de operaciones *monomórficas* (no polimórficas). En términos de implementación, un operador polimórfico universal ejecuta el mismo código para argumentos de cualquier tipo admisible, mientras que un operador polimórfico ad-hoc puede ejecutar diferentes códigos para cada tipo de argumento.

Dentro del polimorfismo universal se distinguen dos tipos distintos de polimorfismo. Por un lado el polimorfismo *paramétrico* y por otro el polimorfismo de *inclusión*. En el polimorfismo paramétrico una operación tiene un parámetro implícito o explícito que determina el tipo de argumento al cual puede aplicarse esta operación. En el polimorfismo inclusión un objeto puede ser visto como perteneciente a muchas clases diferentes. Este último polimorfismo es el que se utiliza para especificar los subtipos y la herencia. Se apellida inclusión ya que se refiere a clases no disjuntas en las que puede existir una relación de contenido entre las mismas.

Dentro del polimorfismo ad-hoc se distinguen otros dos tipos de polimorfismo: la *sobrecarga* de operaciones y las *coerciones*. Con sobrecarga de una operación se suele expresar que un mismo nombre se utiliza para denotar operaciones conceptualmente diferentes. En este caso, el contexto se utiliza para decidir qué operación debe actuar en cada momento. Una coerción es una operación sintáctica que convierte un argumento al tipo esperado por una función, de modo que al actuar esta coerción evita un error de tipo.

En general, cuando hablemos de polimorfismo a lo largo de este capítulo nos referiremos al polimorfismo de inclusión, que es el que obtendremos a partir de la herencia de estructuras. No obstante, como veremos, no vamos a tener un contenido entre las clases y nos apoyaremos en coerciones para formalizar este tipo de polimorfismo presente en Kenzo.

Este capítulo está organizado de la siguiente forma. En la siguiente sección damos una breve presentación del sistema Kenzo[1]. En ella observaremos como Kenzo utiliza, a diferencia de EAT, características de la programación orientada a objetos. En particular nos interesa cómo se hace uso del polimorfismo y de la herencia. Así, en la Sección 5.3 introducimos de modo informal posibles ideas para especificar estas características y en la Sección 5.5 abordamos la especificación de la herencia y del polimorfismo que ésta genera. Previamente dedicamos la Sección 5.4 a analizar brevemente algunas operaciones

[1] No nos ha parecido oportuno hacer una presentación similar de EAT pues ésta está ya publicada en [72], por ejemplo; por otra parte, nos ha parecido imprescindible incluir esta sección debido a ciertos aspectos orientados a objetos de Kenzo que pueden ser considerados “no estándar”, como la relación entre complejos de cadenas y conjuntos simpliciales.

sobrecargadas en Kenzo. Las Secciones 5.6 y 5.7 contienen algunos ejemplos particulares. Terminamos el capítulo con un apartado de problemas pendientes de resolver. Este capítulo puede considerarse una ampliación de los artículos [31] y [32].

5.2 Presentación de Kenzo

El propósito de esta sección es presentar el sistema Kenzo a través de ejemplos que nos permitan estudiar algunas de sus características. Pretendemos con ella examinar algunas de las semejanzas y diferencias de este programa en relación a EAT en cuanto a la codificación de sus estructuras (para una descripción de EAT consultarse [72], por ejemplo).

Imaginemos que queremos calcular el grupo de homología $H_5(\Omega^2 S^3)$, es decir, el quinto grupo de homología del segundo espacio de lazos de la esfera S^3. (El *espacio de lazos de un espacio* es el conjunto de funciones continuas de la circunferencia en el espacio que preservan el punto base, dotado de la topología compacto-abierta.) Con Kenzo, construimos la esfera S^3 a través de la siguiente sentencia[2]:

```
>(setf s3 (sphere 3)) ✠
[K1 Simplicial-Set]
```

El programa devuelve el objeto de Kenzo `K1` que es un conjunto simplicial, es decir, una versión combinatoria de la esfera S^3, y este objeto es asociado al símbolo `s3`.

El siguiente paso es construir el segundo espacio de lazos de esta esfera:

```
> (setf l2s3 (loop-space s3 2)) ✠
[K18 Simplicial-Group]
```

Kenzo construye una versión combinatoria de este espacio. Debemos observar que la versión combinatoria de ese espacio de funciones continuas es de naturaleza infinita. No obstante, se obtiene una codificación de la misma a través de un pequeño conjunto de funciones que integran un objeto grupo simplicial. Un grupo simplicial es un conjunto simplicial provisto de una estructura de grupo compatible con la estructura simplicial (los operadores simpliciales son homomorfismos).

Finalmente, para calcular el quinto grupo de homología de la esfera introducimos la instrucción:

[2] Las sentencias que mostramos en esta sección se han ejecutado en el sistema Kenzo bajo `Allegro Common Lisp`. El símbolo `>` representa al prompt de Lisp y la cruz de malta ✠ corresponde a la tecla `<Return>`, que da paso a la evaluación de la sentencia.

```
> (homology l2s3 5) ✠
homology in dimension 5:
Component Z/3Z
Component Z/2Z
---done---
```

y el sistema devuelve la descripción del grupo de homología $H_5(\Omega^2 S^3) = \mathbb{Z}_2 \oplus \mathbb{Z}_3$. Este resultado se ha obtenido en un segundo con un PC a 400 MHz.

Podemos analizar las estructuras de datos anteriores viendo cómo son almacenadas en la memoria del ordenador. Para ello utilizamos la primitiva Common Lisp `inspect` [93].

```
> (inspect s3) ✠
```

La evaluación de esa expresión Lisp hace que se muestre la ventana de la Figura 5.1.

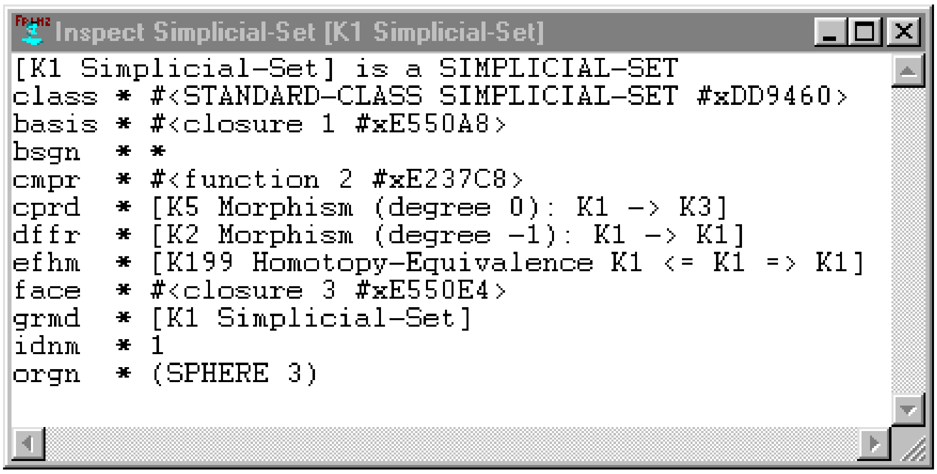

Figura 5.1: Inspect de `s3`

En la Figura 5.1 vemos un ejemplo de cómo `inspect` muestra un objeto CLOS. Utiliza una línea para cada campo del objeto e incluye el nombre del campo delante del símbolo `*`.

Vamos a explicar los campos de un objeto conjunto simplicial como el anterior: el campo `basis` contiene la información del conjunto de símplices, en este caso esta información viene dada por un algoritmo que asocia a cada número natural n una lista con los símplices de dimensión n; `bsgn` almacena el punto base; `cmpr` contiene un test de comparación entre símplices y el campo `face` codifica los operadores cara. Los campos como `grmd, idnm, orgn` corresponden a la organización interna del software y no son relevantes desde el punto de vista de la especificación.

El resto de los campos aparecen por *herencia* a partir de otras clases. Concretamente, los campos `dffr` y `efhm` proceden de la clase complejo de cadenas, y `cprd` de la clase coálgebra. Son precisamente estos campos adicionales la diferencia que tiene la estructura de conjunto simplicial codificada en Kenzo en relación a la estructura codificada en EAT (ver [81, 72]). Aprovechamos para comentar que los campos que determinan la estructura anterior son (`basis, cmpr, face`), todos ellos de naturaleza funcional, bien funciones primitivas (`#<function ...>`) o bien cierres léxicos definidos por el programa (`#<closure ...>`). Este hecho queda reflejado en el objeto final que obteníamos en el capítulo anterior para modelizar esta estructura, en el que el soporte para el género destacado está formado por tuplas funcionales. Concretamente, estas tuplas están compuestas por dos funciones que representan a la operación de comparación y a las operaciones cara sobre un conjunto de símplices prefijado (conjunto al que hace referencia el campo `basis`). En este sentido puede decirse que el modo de implementación usado en EAT y Kenzo para estructuras matemáticas es, en ciento sentido, universal.

Inspeccionamos ahora la estructura `l2s3`.

> `(inspect l2s3)` ✠

La evaluación de la anterior expresión hace que se muestre la ventana de la Figura 5.2.

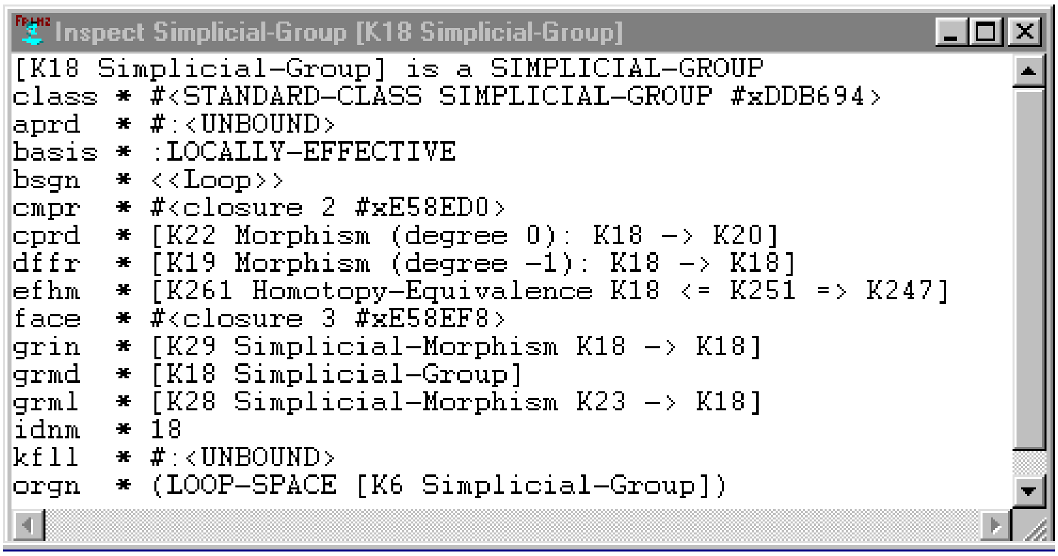

Figura 5.2: Inspect de `l2s3`

Si comparamos las Figuras 5.1 y 5.2, podemos observar como la estructura `l2s3` contiene todos los campos de la anterior estructura de conjunto simplicial. Esto se debe al hecho de que en Kenzo la clase grupo simplicial ha sido definida como una subclase de la clase conjunto simplicial. En términos de Teoría de Categorías podemos decir que existe un *funtor olvido* de la categoría de los grupos simpliciales a la categoría de los conjuntos simpliciales. En nuestro marco algebraico, esto corresponde a considerar que la relación de herencia *"olvida"* alguno de los campos.

Los campos nuevos en esta clase de grupo simplicial son `grml` y `grin`, que definen la operación de multiplicación y la inversa del grupo, respectivamente. Los campos `aprd` y `kfll` pertenecen a esta clase porque, además, la clase de grupo simplicial hereda de otras clases, en concreto de la clase álgebra (`aprd` codifica el producto del álgebra) y la clase Kan (`kfll` codifica el "sombrero" de Kan (ver [65], página 3)).

Una diferencia esencial entre las estructuras `s3` y `l2s3` es que la primera de ellas es una estructura finita, *efectiva* en terminología de Sergeraert [79]. Ésta es una implementación directa de un conjunto simplicial finito. Concretamente, es una versión combinatoria del espacio topológico S^3. Por su parte, `l2s3` es una implementación de un espacio infinito. Sin embargo, dicha implementación recoge toda la información necesaria para realizar ciertos cálculos sobre el espacio topológico al que representa. Esta información es siempre de naturaleza *local*, es decir, relacionada con un único elemento o un conjunto finito de elementos que estén relacionados, en algún sentido algebraico o geométrico. Así, estas estructuras se llaman *localmente efectivas* [79]. El campo `basis` refleja esta característica. En el caso en el que trabajemos con un objeto efectivo, como por ejemplo S^3, este campo almacena una aplicación que asigna a cada dimensión la lista de símplices geométricos del conjunto simplicial. Podemos observar en la Figura 5.1 como el campo `basis` es un cierre léxico. En caso en el que el conjunto simplicial sea infinito en alguna de sus dimensiones o el sistema no contenga información sobre su cardinalidad, entonces se almacena el símbolo `:LOCALLY-EFFECTIVE` en este campo (ver Figura 5.2). Así, en el primer caso, podemos preguntar por la lista de símplices geométricos de `s3`, por ejemplo en dimensión 3:

```
> (basis s3 3) ✠
(S3)
```

La lista está formada por un único símplice, el símplice "fundamental" de S^3 que recibe el nombre de `S3` también. Sin embargo, en el caso del objeto `l2s3` no podemos obtener información similar ya que la "lista" de símplices geométricos en dimensión 3 del espacio codificado sería infinita:

```
> (basis l2s3 3) ✠
;; Error:  The object [K18 Simplicial-Group] is locally-effective.
```

Sin embargo, podemos trabajar con símplices particulares de este grupo simplicial, podemos comparar si dos símplices son iguales, calcular sus caras, etc. Por ejemplo:

```
> (face l2s3 0 3 (loop3 1 (loop3 1 's3 1) 1)) ✠
<AbSm - <<Loop[<<Loop[0 s3]>>]>>>
```

aplica la cara δ_0^3 del grupo simplicial a un símplice de `l2s3`.

Esta distinción entre estructuras finitas y no finitas es compartida con su predecesor EAT. Debemos comentar que en los capítulos previos no hemos considerado dentro de nuestra signatura para los conjuntos simpliciales una operación que recoja el operador `basis`. El motivo es que no hemos tenido en consideración la distinción entre objetos efectivos y localmente efectivos, de modo que la base era recogida para ambos casos por un dominio prefijado, sin preocuparnos de la cardinalidad del mismo.

Otro aspecto importante en Kenzo es que los operadores que se utilizan pueden estar *sobrecargados*, lo que es una de las ventajas que incorpora el uso de la programación orientada a objetos. Por ejemplo, para calcular la cara δ_0^3 del símplice fundamental del conjunto simplicial que representa a S^3 podemos utilizar la misma función:

```
> (face s3 0 3 's3) ✠
<AbSm 1-0 *>
```

Vamos a aprovechar la evaluación anterior para comentar el papel que en el operador juega el primero de sus argumentos, ya que es diferente al desempeñado por el resto de argumentos. El primer argumento determina el "espacio ambiente" en el que se realizan los cálculos. Los datos para este argumento son más bien "ocultos" ya que aunque su estructura pueda explorarse explícitamente por medio de inspect (ver Figuras 5.1 y 5.2), su composición interna no puede mostrarse: los valores de los campos son cierres léxicos. La conclusión de esto es que los objetos asociados a s3 y l2s3 sólo tienen *comportamiento*, son *exclusivamente* de observación, sobre una serie de argumentos para estos cierres léxicos. Por el contrario, el resto de argumentos son "visibles", en el sentido de que su estructura es completamente conocida. Esta característica la poseen también los resultados obtenidos al aplicar las operaciones. Por ejemplo, el símplice abstracto <AbSm 1-0 *> es la degeneración $\eta_1\eta_0$ del punto base de la esfera (denotado por *). Su estructura es completamente "visible" a partir de su representación en la máquina. Debemos observar que los datos "visibles" pueden ser de dos tipos: constantes, como los enteros 0 y 3, o generados como <AbSm 1-0 *>. Esta distinción dentro de los géneros visibles fue detectada en los ejemplos para EAT del capítulo anterior. De este modo no es preciso incluir todas las constantes para el dominio de datos dentro de cada álgebra, sino que es suficiente con disponer de operaciones que permitan generar un elemento cuando éste sea necesario.

Como hemos comentado, el objetivo de los programas EAT y Kenzo es el cálculo de la homología de ciertos espacios topológicos. Como herramienta para calcular la homología de espacios de naturaleza infinita Sergeraert introdujo la noción de *objeto con homología efectiva* [88]. Un objeto con homología efectiva es un objeto "mixto" que presenta características efectivas y localmente efectivas, permitiendo aprovechar las ventajas que proporcionan ambas: la codificación localmente efectiva de un objeto de naturaleza infinita permite resolver las dificultades que deriven de la no finitud, mientras que la codificación efectiva permite obtener información de naturaleza global, como es la homología del objeto. Los dos tipos de codificación están relacionadas a través de objetos híbridos, esencialmente equivalencias de homotopía entre complejos de cadenas efectivos y complejos de cadenas localmente efectivos [80]. Estas equivalencias de homotopía, que permiten trabajar con objetos localmente efectivos, están almacenadas en el campo efhm de las estructuras anteriores. Podemos ver un ejemplo de objeto híbrido en este campo del objeto l2s3 (ver Figura 5.2), entre los complejos de cadenas localmente efectivos K18 y K251 y el complejo de cadenas efectivo K247. Sin entrar en detalles, esta equivalencia de homotopía es una herramienta que nos permite calcular la homología de un objeto localmente efectivo, en este caso `K18` (`12s3`), construyendo un objeto efectivo `K247` con la misma homología (este objeto puede entenderse como una descripción de la homología del anterior objeto), a través de un objeto intermediario `K251`. Podemos encontrar en [90, 89] una descripción detallada del anterior proceso. Si analizamos el tipo de datos de este objeto auxiliar, comprobamos que en efecto se trata de un complejo de cadenas:

```
> (K 251) ✠
[K251 Chain-Complex]
```

Si ahora intentamos calcular su quinto grupo de homología:

```
> (homology (K 251) 5) ✠
;; Error:  I don't know how to determine the effective homology of:
[K251 Chain-Complex] (Origin:  (BICONE [K111 Reduction K95⇒K33]
[K235 Reduction K233⇒K33]).
```

El sistema falla ya que no tiene información suficiente para calcular la homología de este complejo de cadenas localmente efectivo. Vamos a observar con más detenimiento este objeto. Para ello nos ayudamos otra vez de la primitiva **`inspect`** cuyo resultado se recoge en la Figura 5.3.

```
> (inspect (K 251)) ✠
```

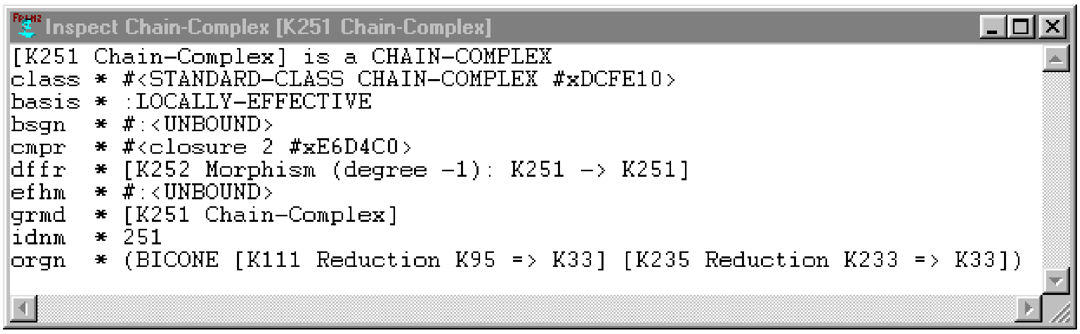

Figura 5.3: Inspect de `K251`

La clase complejo de cadenas es una de la más elementales en Kenzo. Sus campos principales (en el caso localmente efectivo) son **`cmpr`**, que codifica una igualdad entre generadores, y **`dffr`**, que codifica las aplicaciones diferenciales. Ambos campos son funcionales, y son también los campos fundamentales para codificar un complejo de cadenas en EAT (ver [81, 72]). Este hecho también ha quedado patente en el objeto final que obtuvimos en el capítulo anterior para modelizar esta estructura, en el que el soporte para el género destacado estaba formado por tuplas de dos funciones que representan a las anteriores operaciones.

La clase complejo de cadenas es heredada por la clase conjunto simplicial (podemos observar como cada campo de la Figura 5.3 también aparece en las Figuras 5.1 y 5.2). En concreto, la clase conjunto simplicial se construye por herencia de la clase coálgebra e incluye un nuevo campo `face` que recoge las operaciones cara; a su vez la clase coálgebra se construye por herencia de la clase complejo de cadenas e incluye un nuevo campo `cprd` que corresponde al coproducto de la coálgebra (ver [34]).

La herencia entre grupo simplicial y conjunto simplicial parece bastante natural ya que se admite de forma general que un grupo simplicial *es* (en particular) un conjunto simplicial. Esta herencia entre ambas corresponde a una relación *is_a* que es el "modelo estándar" de entender la herencia orientada a objetos (ver [66], por ejemplo). Sin embargo, en Topología Algebraica se dice de manera usual que un conjunto simplicial *tiene* asociado canónicamente un complejo de cadenas, y no se considera que un conjunto simplicial *es* un complejo de cadenas. Así, en este caso se utiliza herencia para implementar una relación *has_a* (ver nuevamente [66]). Esto muestra que la noción de herencia es, conceptualmente, complicada de entender (podemos referirnos de nuevo a [66], [94] o [100]).

No obstante, debemos observar que en términos de la teoría de categorías, existe un funtor F de la categoría de los conjuntos simpliciales a la categoría de los complejos de cadenas. Así, la situación no es tan diferente del caso entre los grupos simpliciales y los conjuntos simpliciales en la que se establece la relación entre ambas a través del funtor *olvido*. En una situación como ésta en la que tenemos dos categorías $\mathcal{C}$ y $\mathcal{D}$ y un funtor entre ambas $F\colon \mathcal{C} \to \mathcal{D}$, siempre es posible *enriquecer* la categoría origen $\mathcal{C}$ para obtener para cada objeto X de $\mathcal{C}$ una pareja $[X, F(X)]$. Obviamente existe un funtor de esta nueva categoría a $\mathcal{D}$, que representa al funtor F. Esta idea corresponde con lo realizado por Eilenberg y MacLane, quienes reemplazaron la noción de conjunto simplicial por la de *FD-complejo* [65]. Esta estructura ha sido implementada en Kenzo para codificar los conjuntos simpliciales y nosotros trataremos de describirla formalmente a lo largo de este capítulo.

No queremos terminar esta sección sin mostrar al menos uno de los cálculos interesantes realizados por Kenzo. En este momento, el sistema Kenzo ha conseguido obtener grupos de homología que son muy complicados de conseguir por los métodos tradicionales o que incluso no han sido obtenidos a través de la Topología Algebraica "clásica". Por ejemplo, vamos a calcular el grupo de homología $H_5(\Omega^3\mathrm{Moore}(\mathbb{Z}/2\mathbb{Z}, 4))$, donde el espacio $\mathrm{Moore}(\mathbb{Z}/2\mathbb{Z}, 4)$ es el espacio conexo "canónico" que tiene como única homología no trivial en dimensión 4 a $\mathbb{Z}/2\mathbb{Z}$. Este grupo de homología es "en principio" alcanzable a través de métodos tradicionales, ver [18], pero la experiencia muestra que incluso los mejores topólogos encuentran dificultades para determinarlo. Con la ayuda del programa Kenzo, primero construimos el tercer espacio de lazos del espacio de Moore:

```
> (setf l3m4 (loop-space (moore 2 4) 3)) ✠
[K291 Simplicial-Set]
```

y después calculamos su quinto grupo de homología:

```
> (homology l3m4 5) ✠
homology in dimension 5:
Component Z/2Z
```

```
Component Z/2Z
Component Z/2Z
Component Z/2Z
Component Z/2Z
---done---
```

Obtenemos como resultado que $H_5(\Omega^3\mathrm{Moore}(\mathbb{Z}/2\mathbb{Z},4)) = (\mathbb{Z}/2\mathbb{Z})^5$. Este cálculo ha sido realizado en 1 minuto y 30 segundos en un PC a 400 MHz.

La cuestión que se plantea es la fiabilidad del resultado obtenido, ya que no parece muy razonable confirmarlo ni refutarlo directamente. Por este motivo, consideramos que todos los esfuerzos encaminados al análisis de la corrección del sistema son de gran importancia. Como paso previo resulta imprescindible la modelización del sistema, por medio de la especificación de sus estructuras de datos y de sus piezas algorítmicas fundamentales.

5.3 Idea intuitiva sobre cómo modelar la herencia entre complejos de cadenas y conjuntos simpliciales

En este sección trataremos de estudiar la herencia que se presenta en Kenzo entre dos de sus estructuras básicas, como son los complejos de cadenas y los conjuntos simpliciales. Como hemos comentado en la sección previa la formalización de la herencia en general es una tarea complicada que escapa a los propósitos de este trabajo. Por ello nos restringiremos exclusivamente a tratar de razonar sobre la herencia presente entre estas dos estructuras, como ejemplo paradigmático para tratar de obtener un tipo de especificación que sea extrapolable al resto del programa Kenzo.

Como hemos mencionado previamente, Kenzo considera un conjunto simplicial como un *FD-complejo de Eilenberg-MacLane* (ver [33, 34]). Esta estructura consiste en dotar a un conjunto simplicial de una estructura de complejo de cadenas: el complejo de cadenas normalizado canónicamente asociado a este conjunto simplicial, que hemos detallado en la Sección 4.6.2.3. Recordamos que esta construcción consiste en que un conjunto simplicial (K, δ, η) se completa con una estructura diferencial $d_n \colon C(K)_n \to C(K)_{n-1}$, donde $C(K)_n$ es el $\mathbb{Z}$-módulo libre generado por los símplices geométricos de dimensión n de K, y cada aplicación d_n se define por

$$d_n := \sum_{i=0}^{n} (-1)^i \delta_i^n,$$

considerando en la anterior suma los elementos en los que el resultado de la cara sea un símplice no degenerado. En [65] podemos encontrar una definición precisa de este concepto así como el desarrollo teórico correspondiente.

Así, mientras el sistema EAT suministra una función constructora que obtiene este complejo de cadenas a partir de un conjunto simplicial (cuando esto sea necesario), Kenzo incorpora a la estructura de conjunto simplicial este complejo de cadenas particular. Para ello el sistema usa la técnica de herencia que le proporciona la programación orientada a objetos. Se tiene de este modo, para esta construcción central en ambos programas, la reutilización de los elementos comunes así como algunas funciones polimórficas importantes, como es, por ejemplo, el test de igualdad sobre los elementos generadores (ahora compartidos) para ambas estructuras. Este ejemplo ilustra los beneficios de la programación orientada a objetos desde el punto de vista de la ingeniería del software.

Ahora, si pretendemos modelar esta nueva situación, un primer intento es tratar de utilizar especificación con orden en los géneros (*order-sorted specification* [44]). Básicamente, este tipo de especificaciones consiste en incorporar una relación de orden entre los géneros de la signatura. Posteriormente, esta relación de orden entre los géneros se debe transformar en una relación de contenido en los soportes de sus modelos. Este tipo de especificación es el que generalmente se ha incorporado a la especificación oculta para modelar la herencia (ver por ejemplo [41, 40]): a la signatura para la estructura "madre" se le añaden algunas operaciones y algunos géneros junto con un orden entre uno de los géneros de la "madre" y uno de los que hemos añadido, que marque la relación de herencia entre ambas estructuras (remitimos nuevamente a [44] para una exposición detallada sobre este punto).

Si utilizamos en nuestro caso esta técnica, podemos construir una nueva signatura que se obtiene al añadir a la signatura para los complejos de cadenas $\Sigma_{imp}^{cc,eq}$, las operaciones y géneros de la signatura para los conjuntos simpliciales de $\Sigma_{imp}^{ss,eq}$ y declaramos el siguiente orden entre los géneros ocultos: $imp_{\Sigma^{ss,eq}} < imp_{\Sigma^{cc,eq}}$.

Sin embargo, este enfoque no es conveniente por una razón de naturaleza semántica. La declaración sintáctica $imp_{\Sigma^{ss,eq}} < imp_{\Sigma^{cc,eq}}$ conlleva a nivel semántico que para cada álgebra A de nuestra signatura se verifique que $A_{imp_{\Sigma^{ss,eq}}} \subseteq A_{imp_{\Sigma^{cc,eq}}}$. Pero la implementación que Kenzo hace de estas estructuras, que hemos analizado en la sección anterior, así como los objetos finales que obteníamos para las mismas en los capítulos previos, ilustran el hecho bien conocido de que la herencia, en el contexto del *álgebra universal*, es más bien una cuestión de *olvido*, más que de inclusión. Por ejemplo, la relación que existe entre los grupos y los semigrupos, no se entiende como que un grupo esté contenido en un semigrupo, sino más bien que a partir de un grupo se obtiene un semigrupo si nos olvidamos de las operaciones de elemento neutro e inverso.

El hecho de que existan relaciones entre estructuras que no pueden ser modeladas a partir de la idea original de interpretar el orden entre los géneros como inclusiones ha sido puesto de manifiesto por muchos autores. Por ejemplo en [69], en el contexto del lenguaje de especificación algebraica CASL [98], se definen dos relaciones de orden $\leq^1$ y $\leq^2$. La primera está incluida dentro del lenguaje CASL y pretende representar la interpretación usual de relación entre géneros como inclusión. La segunda, que proponen incorporarla a dicho lenguaje, está más cercana a la traducción de esta relación a través de una *coerción.* La idea de interpretar una relación entre géneros como una coercion no es original de [69]. Ésta ha sido propuesta previamente por otros autores en [100, 13]. Todos ellos proponen coerciones para modelar la *sobrecarga* de operaciones que aparecen explícitamente con el mismo nombre dentro de una signatura. Nosotros utilizaremos también este tipo de coerciones en la siguiente sección para especificar la sobrecarga de algunos operadores particulares presentes en Kenzo. Por ejemplo, podemos incluir una operación de coerción entre generadores y combinaciones que nos permita especificar el operador sobrecargado que corresponde a la diferencial, el cual puede actuar bien sobre combinaciones o bien sobre generadores junto con la dimensión de la mismos. No obstante este uso de las coerciones parece ajeno a la herencia.

El uso que nosotros hacemos de las coerciones en la herencia es ligeramente distinto. A través de una coerción pretendemos especificar el *polimorfismo* por reutilización de operaciones que se da en la herencia. En la especificación *order-sorted* se supone que un elemento para el soporte en un determinado género es también del soporte de cualquier *supergénero* del mismo. De este modo, se produce en este elemento una *coerción implícita*, debida a este contenido, que se traduce en polimorfismo en las operaciones: cualquier operación de un género puede actuar sobre cualquier elemento para un *subgénero.* En nuestro caso, no tendremos dicho contenido entre los conjuntos soporte y preferimos utilizar una *coerción explícita* entre subgénero y supergénero, de modo que cualquier operación con dominio el subgénero pueda ser utilizada sobre un elemento del supergénero tras actuar la coerción. De este modo, el modelo de la herencia entre estructuras no produce la repetición de dos operaciones con el mismo nombre y distintos géneros (sobrecarga) en una misma signatura, sino que una misma operación puede utilizarse sobre elementos de género distinto tras actuar la operación de coerción sobre ellos (polimorfismo). En [37] se utiliza un sistema de coerciones implícito para tratar de construir una jerarquía algebraica constructiva en Coq [24]. En este caso utilizan este sistema como paso previo para realizar la implementación de estas estructuras algebraicas. Curiosamente en este sistema no utilizan las coerciones para especificar la sobrecarga de operaciones ya que éstas no están permitidas. Nosotros estamos en una situación diferente, en la que partimos de un sistema ya implementado y en funcionamiento y pretendemos realizar un modelado del mismo.

Antes de pasar a tratar de formalizar la anterior relación de herencia, dedicaremos la siguiente sección a estudiar algunos ejemplos de operaciones sobrecargadas que aparecen en Kenzo, en los que podemos aplicar la técnica de coerción comentada anteriormente. En particular, estas operaciones son la diferencial en los complejos de cadenas y la operación cara en los conjuntos simpliciales.

5.4 Algunas operaciones sobrecargadas en Kenzo

A pesar de que la representación que utiliza Kenzo para un complejo de cadenas es básicamente la misma que la que usa EAT [34, 81], salvando las distancias entre *clase* de CLOS y *estructura* de Lisp [93], podemos destacar algunas diferencias. Por ejemplo, la operación de comparación que se ha implementado para los generadores en Kenzo en lugar de ser una igualdad, establece una relación de orden total entre los mismos. Entonces, se puede obtener una representación de las combinaciones en las que los monomios están ordenados atendiendo a este orden de los generadores. El objetivo de este orden es obtener algoritmos que manejan combinaciones de modo más eficaz en cuanto a la rapidez de cálculo. En este trabajo no tendremos en cuenta este orden, que es importante desde el punto de vista de la implementación pero no tanto para la especificación de la estructura. Así, mantenemos la misma igualdad sobre generadores y el mismo dominio de datos para las combinaciones que definimos en el capítulo anterior para los complejos de cadenas.

Otra diferencia que nos resulta más interesante entre ambos programas es la sobrecarga en Kenzo de algunas operaciones. Por ejemplo, en Kenzo aparecen dos operaciones diferenciales que comparten el nombre. Una de ellas se corresponde con la operación diferencial habitual sobre combinaciones $imp\_df\colon imp_{\Sigma^{cc,eq}} cmb \rightarrow cmb$. Una segunda operación viene dada por $imp\_df\colon imp_{\Sigma^{cc,eq}} int\ gnr \rightarrow cmb$. Ésta representa a la diferencial sobre los elementos generadores, donde el género *int* abstrae el grado de dicho generador (ver [34], página 12).

Para especificar la sobrecarga de esta operación incluimos en la signatura oculta $\Sigma^{cc,eq}_{imp}$, que ha sido utilizada en el capítulo anterior para especificar familias de complejos de cadenas, una nueva operación

$$imp\_df\colon imp_{\Sigma^{cc,eq}} int\ gnr \rightarrow cmb,$$

con dominio de definición determinado a partir de $dom^{cc}_{df} := \{(p,a) \in \mathbb{Z} \times D^{cc}_{gnr} \mid dim^{cc}(a) = p\}$, junto con la siguiente operación de coerción:

$$coer^{int \times gnr}_{cmb}\colon int \times gnr \rightarrow cmb.$$

En el enfoque *order-sorted* si consideramos una relación de orden entre los géneros visibles *gnr* y *cmb*, entonces debe verificarse que existe un contenido entre los dominios de datos D^{cc}_{gnr} y D^{cc}_{cmb}. Esto no se cumple en nuestro caso, por lo que este tipo de especificaciones no son adecuadas para tratar de especificar la sobrecarga de esta operación. Resulta más adecuado la visión por medio de una operación de coerción.

La anterior operación de coerción es considerada visible, ya que todos sus géneros son visibles. Asociamos a esta operación una función parcial con dominio de definición determinado por el dominio de definición $dom^{cc}_{coer^{int\times gnr}_{cmb}} := dom^{cc}_{df}$ y que se define para estos elementos por

$$coer^{int \times gnr}_{cmb}(p,a) := \langle p, [(1,a)] \rangle.$$

Entonces, se asume que la nueva operación diferencial sobre los elementos generadores está canónicamente definida por

$$df(cc,p,a) := df(cc, coer^{gnr}_{cmb}(p,a)).$$

La anterior operación es una coerción "2 a 1" (es decir, que combina dos datos de dos tipos para obtener un dato de un tipo nuevo) y la sobrecarga que se produce a través de ella en el operador diferencial no está permitida en *ningún* lenguaje de programación, ya que siempre deben ser "1 a 1" (en Kenzo se implementa por medio de *macros*, ver [93], página 193).

Entre estas dos operaciones se produce un hecho curioso. Tenemos que la operación diferencial sobre generadores también determina canónicamente a la operación diferencial sobre combinaciones. Una vez que tenemos una definición de la diferencial sobre los generadores, obtenemos por linealidad la de cualquier combinación. Éste es el motivo de que aparezca este operador sobrecargado en Kenzo (en EAT se refleja con dos operadores con distinto nombre), ya que para tener representada la diferencial es suficiente con definirla sobre los generadores. Este hecho también quedó reflejado en el objeto final que teníamos para los complejos de cadenas en el capítulo anterior. En particular, el soporte para el género destacado de este objeto estaba formado por tuplas funcionales que determinaban un complejo de cadenas en las que la función correspondiente a la diferencial estaba definida simplemente sobre los generadores. De este modo la implementación elegida en Kenzo (y también en EAT), además de ser la "más general" posible, es eficiente, en el sentido de que busca ser "mínima" entre todos los posibles objetos finales isomorfos.

Por su parte, la implementación de los conjuntos simpliciales en Kenzo es totalmente distinta a la que se tiene en EAT, ya que como hemos comentado esta estructura se define por herencia a través de los complejos de cadenas. Posteriormente trataremos de estudiar esta relación de herencia. Ahora nos fijaremos en otras diferencias en la implementación de esta estructura. Por ejemplo, como comentamos en el capítulo anterior, EAT utiliza una lista decreciente de enteros para marcar las degeneraciones, si es que las tiene, del símplice. Sin embargo, por ser esta lista estrictamente decreciente puede ser representada de una manera ingeniosa por un único número no negativo. Por ejemplo, la lista $(5, 2, 0)$ puede ser codificada por el número 37, sabiendo que $37 = 2^5 + 2^2 + 2^0$. A través de esta nueva codificación se obtienen implementaciones muy eficientes de las operaciones del conjunto simplicial. Otra vez este tipo de diferencias son cuestiones de implementación que no nos interesan a nivel de modelado, de modo que preferimos quedarnos con la representación más intuitiva y más cercana la codificación en topología simplicial (ver por ejemplo [65]) de las degeneraciones como lista decreciente de enteros.

Otra diferencia más interesante en la implementación de esta estructura es que Kenzo también presenta operadores sobrecargados para los conjuntos simpliciales. En particular aparecen dos operaciones cara que comparten el nombre. Por un lado tenemos la operación $imp\_face\colon imp_{\Sigma^{ss,eq}} nat\ nat\ asm \rightarrow asm$ que calcula la cara en un conjunto simplicial sobre símplices abstractos y por otro la operación $imp\_face\colon imp_{\Sigma^{ss,eq}} nat\ nat\ gsm \rightarrow asm$ que actúa sobre símplices geométricos.

Si suponemos que estamos trabajando con la signatura que hemos definido anteriormente para especificar los conjuntos simpliciales $\Sigma^{ss,eq}_{imp}$ podemos nuevamente especificar la sobrecarga de esta operación si incluimos en esta signatura una operación

$$imp\_face\colon imp_{\Sigma^{ss,eq}} nat\ nat\ gsm \rightarrow asm$$

con dominio de definición determinado a partir de $dom^{ss}_{face} := \{(i, n, a) \in \mathbb{N} \times \mathbb{N} \times D^{ss}_{gsm} \mid 0 \leq i \leq n, n > 0, dim^{ss}(a) = n\}$. Debemos observar que también hemos sobrecargado el símbolo utilizado para denotar el dominio de definición. Los dominios de definición para este caso particular son disjuntos de modo que el contexto permite distinguirlos. En otro caso, podemos por ejemplo incorporar a la notación de estos conjuntos la aridad de la operación de modo que rompamos la sobrecarga de este símbolo.

Entonces, añadimos a la signatura una operación de coerción

$$coer^{gsm}_{asm}\colon gsm \rightarrow asm$$

que corresponde al orden que podemos establecer entre los géneros visibles gsm y asm. Éste es un ejemplo más de soportes en los que no es válida la teoría *order-sorted* ya que obviamente $D^{ss}_{gsm} \not\subseteq D^{ss}_{asm}$. Esta operación de coerción es de nuevo una operación visible y le asociamos la función total entre D^{ss}_{gsm} y D^{ss}_{asm} que se define por $coer^{gsm}_{asm}(a) := \langle (\ ), a\rangle$ para todo $a \in D^{ss}_{gsm}$.

Entonces, se asume que la operación imp_face que añadimos está canónicamente definida por

$$imp\_face(ss, i, n, a) := imp\_face(ss, i, n, coer^{gsm}_{asm}(a)).$$

Nuevamente, la idea de esta sobrecarga de operaciones en Kenzo es que la operación cara sobre los símplices geométricos determina canónicamente la operación cara sobre los símplices abstractos, a partir de las propiedades que permiten conmutar las operaciones cara y degeneración. Este hecho quedó nuevamente reflejado en el objeto final que obtuvimos para los conjuntos simpliciales en el capítulo anterior. El soporte para el género destacado de este objeto estaba formado por tuplas funcionales que determinaban un conjunto simplicial en las que la función correspondiente a las operaciones cara estaba definida únicamente sobre los símplices geométricos.

5.5 Especificación de la herencia de Kenzo a través de coerciones

En esta sección trataremos de especificar parte de la herencia que aparece en Kenzo, que se puede explicar a través de relaciones de olvido entre las distintas estructuras algebraicas con las que trabaja. La técnica que utilizaremos no está muy distante de la que teníamos para especificar relaciones entre distintas estructuras de EAT. No obstante, introducimos ahora los beneficios que se obtienen de la programación orientada a objetos, como son la posibilidad de reutilizar estructuras para construir otras y el polimorfismo de operaciones.

Partimos de dos categorías $\mathcal{T}^1$ y $\mathcal{T}^2$, que están relacionadas a través de una relación de olvido dada por un funtor $F\colon \mathcal{T}^2 \to \mathcal{T}^1$. Entonces, suponemos que la categoría $\mathcal{T}^1$ admite una modelización siguiendo la técnica presente en el capítulo anterior. Básicamente, disponemos de una signatura $\Sigma^1 = (S^1, \Omega^1)$, un dominio de datos D^1 junto con un dominio de definición dom^1, y definimos una subcategoría plena $\mathcal{C}^1$ de $P\tilde{A}lg^{D^1,dom^1}(\Sigma^{1,eq})$, tal que cada A objeto de $\mathcal{C}^1$ verifica que el álgebra parcial cociente $A/_{eq^1_A}$ que se obtiene a partir de la relación de congruencia parcial determinada por las funciones de igualdad define un objeto de $\mathcal{T}^1$.

Entonces, para especificar familias de objetos de $\mathcal{T}^1$, construimos una signatura oculta $\Sigma^{1,eq}_{imp}$ y una subcategoría plena $\mathcal{C}^1_{imp}$ de $HPAlg^{D^1,dom^1}(\Sigma^{1,eq}_{imp})$ tal que cada álgebra A de esta categoría verifica que para cada $a \in A_{imp_{\Sigma^{1,eq}}}$, la $\Sigma^{1,eq}$-álgebra A^a es un objeto de $\mathcal{C}^1$.

Para modelar la segunda categoría y la relación de olvido entre ambas, suponemos que tenemos definida una signatura $\Sigma^2 = (S^2, \Omega^2)$, tal que $\Sigma^1 \subseteq \Sigma^2$, junto con un dominio de datos D^2 y un dominio de definición dom^2, tal que $D^1 \subseteq D^2$ y $dom^1 \subseteq dom^2$. Además, podemos definir una subcategoría $\mathcal{C}^2$ de $P\tilde{A}lg^{D^2,dom^2}(\Sigma^{2,eq})$, formada por las álgebras A de esta categoría que verifican que el cociente $A/_{eq^2_A}$, que se obtiene a partir de la relación de congruencia parcial eq^2_A en A, define un objeto de $\mathcal{T}^2$ tal que si le aplicamos el funtor F obtenemos el objeto de $\mathcal{T}^1$ dado por $(A|\Sigma^{1,eq})/_{eq^1_A}$. Con eq^1_A nos referimos a la relación de congruencia parcial en $A|\Sigma^{1,eq}$ que se obtiene a partir de eq^2_A al restringirse a las operaciones de igualdad que provienen de géneros en Σ^1 .

Estas condiciones pueden parecer muy exigentes, pero, como veremos en los ejemplos, se dan con total naturalidad cuando tratamos de especificar la relación de olvido entre estructuras algebraicas y en particular con las estructuras que maneja Kenzo. Las anteriores condiciones son similares a las que exigimos en el capítulo anterior centrándonos ahora en que el funtor representa una relación de olvido.

Ahora, para especificar familias de objetos de $\mathcal{T}^2$ construimos una signatura oculta $\Sigma^{2,eq}_{imp}$ que contenga los géneros y operaciones de $\Sigma^{1,eq}_{imp}$ junto con los géneros $S^2 \setminus S^1 \cup \{imp_{\Sigma^{2,eq}}\}$ y las operaciones $\{imp\_\omega\colon imp_{\Sigma^{2,eq}} s_1 \ldots s_n \to s\}_{\omega\colon s_1 \ldots s_n \to s \in \Omega^2 \setminus \Omega^1} \cup \{coer^{imp_{\Sigma^{2,eq}}}_{imp_{\Sigma^{1,eq}}}\colon imp_{\Sigma^{2,eq}} \to imp_{\Sigma^{1,eq}}\}$. A través de esta signatura pretendemos especificar además la relación de olvido que existe entre las dos estructuras anteriores.

Reflejamos en la anterior signatura una reutilización a nivel sintáctico, ya que construimos la signatura para la estructura "hija" aprovechando la signatura para la estructura "madre". Además obtendremos polimorfismo en las operaciones, ya que vamos a poder utilizar, gracias a la operación de coerción, operaciones de la "hija" sobre elementos de la "madre". La diferencia con la especificación adoptada en el capítulo anterior consiste en que antes construíamos la signatura $\Sigma^{2,eq}_{imp}$ añadiendo el género $imp_{\Sigma^{2,eq}}$ como primer argumento a todas las operaciones de $\Sigma^{2,eq}$ sin preocuparnos de aprovechar las definidas en $\Sigma^{1,eq}_{imp}$.

A nivel de modelos, debemos restringirnos a una subcategoría de álgebras para la signatura $\Sigma^{2,eq}_{imp}$ de modo que cada álgebra de esta categoría represente a familias de objetos de $\mathcal{T}^2$ apoyándonos en la operación de coerción para utilizar las operaciones de $\Sigma^{1,eq}_{imp}$ sobre elementos del soporte correspondiente a $imp_{\Sigma^{2,eq}}$. Además, obtenemos que si en una de estas álgebras olvidamos la estructura que corresponde a la parte de signatura que se añade para definir la signatura "hija", obtenemos un álgebra de la categoría fijada para la signatura "madre". Esto es, cada álgebra A de nuestra categoría debe verificar que para cada elemento fijado $a \in A_{imp_{\Sigma^{2,eq}}}$, la $\Sigma^{2,eq}$-álgebra que se define tras fijar en las operaciones de A el primer argumento bien en a si su género corresponde a $imp_{\Sigma^{1,eq}}$ o bien en $(coer^{imp_{\Sigma^{2,eq}}}_{imp_{\Sigma^{1,eq}}})_A(a)$ si su género corresponde a $imp_{\Sigma^{2,eq}}$, determine una $\Sigma^{2,eq}$-álgebra de $\mathcal{C}^2$. De este modo conseguimos dotar a la operación de coerción de una interpretación como relación olvido entre ambas categorías, ya que $(coer^{imp_{\Sigma^{2,eq}}}_{imp_{\Sigma^{1,eq}}})_A(a)$ representará al objeto de $\mathcal{T}^1$ que se obtiene por olvido a través de la anterior álgebra.

Para que la signatura $\Sigma^{2,eq}_{imp}$ quede completamente definida como signatura oculta debemos todavía decidir qué parte de la misma es oculta y qué parte es visible, así como fijar el dominio de datos para esta última parte. Para ello, debemos comenzar por determinar cuáles de sus géneros son ocultos. Mientras que parece claro que el género $imp_{\Sigma^{2,eq}}$ debe ser oculto, la naturaleza de $imp_{\Sigma^{1,eq}}$ es más controvertida. Si las estructuras $\mathcal{T}^1$ y $\mathcal{T}^2$ se consideran "al mismo nivel" entonces tanto $imp_{\Sigma^{2,eq}}$ como $imp_{\Sigma^{1,eq}}$ deberían ser declarados ocultos. Por el contrario, si $\mathcal{T}^1$ se considera como una estructura de datos previa y auxiliar, entonces $imp_{\Sigma^{1,eq}}$ debe declararse visible. En las dos próximas secciones estudiaremos por separado ambas alternativas.

5.5.1 Signatura oculta con dos géneros ocultos

Si declaramos tanto $imp_{\Sigma^{1,eq}}$ como $imp_{\Sigma^{2,eq}}$ géneros ocultos en la signatura $\Sigma^{2,eq}_{imp}$, entonces se genera en dicha signatura una operación constructora, que corresponde a la operación de coerción, mientras que el resto son operaciones deconstructoras con género oculto uno de los dos anteriores. Para esta signatura oculta fijamos el dominio de datos que viene dado por D^2 e incluimos en la misma una constante por cada elemento de este dominio. Además prefijamos, como hemos venido haciendo, un dominio de definición para sus operaciones a partir de dom^2, y consideramos la operación de coerción como una operación total. Continuaremos trabajando con la categoría de álgebras parciales ocultas para esta signatura que definimos en el capítulo anterior y que denotamos por $HPAlg^{D^2,dom^2}(\Sigma^{2,eq}_{imp})$.

Entonces, nos restringimos a una subcategoría plena $\mathcal{C}^2_{imp}$ de $HPAlg^{D^2,dom^2}(\Sigma^{2,eq}_{imp})$ tal que cada $\Sigma^{2,eq}_{imp}$-álgebra A que sea objeto de $\mathcal{C}^2_{imp}$ verifique las siguientes condiciones:

- La restricción de A a $\Sigma^{1,eq}_{imp}$ es un objeto de $\mathcal{C}^1_{imp}$; es decir, $A|\Sigma^{1,eq}_{imp} \in \mathcal{C}^1_{imp}$.
- Para cada $a \in A_{imp_{\Sigma^{2,eq}}}$ definimos una $\Sigma^{2,eq}$-álgebra parcial A^a dada por: a cada género $s \in S^{2,eq}$ se le asocia como soporte el conjunto $A^a_s = D^2_s$; cada operación $\omega\colon s_1 \dots s_n \to s \in \Omega^{2,eq} \setminus \Omega^{1,eq}$, $n \geq 0$, se interpreta en A^a mediante la función parcial ω_{A^a} con dominio de definición $Def(\omega_{A^a}) = dom^2_\omega$ y dada por

$\omega_{A^a}(d_1, \ldots, d_n) := imp\_\omega_A(a, d_1, \ldots, d_n)$, para cada $(d_1, \ldots, d_n) \in dom^2_\omega$; por último, cada operación $\sigma\colon s_1 \ldots s_n \to s \in \Omega^{1,eq}$, $n \geq 0$, se interpreta por la función parcial σ_{A^a} con dominio de definición $Def(\sigma_{A^a}) = dom^1_\sigma$ y dada por $\sigma_{A^a}(d_1, \ldots, d_n) := imp\_\sigma_A((coer^{imp_{\Sigma^{2,eq}}}_{imp_{\Sigma^{1,eq}}})_A(a), d_1, \ldots, d_n)$, para cada $(d_1, \ldots, d_n) \in dom^1_\sigma$. Entonces, se exige que A^a pertenezca a $\mathcal{C}^2$.

Lo importante es que bajo estas condiciones podemos definir un objeto canónico en $\mathcal{C}^2_{imp}$. Éste se va a obtener a partir del objeto canónico de $\mathcal{C}^1_{imp}$, que como hemos visto se caracteriza por tener como soporte para el género oculto $imp_{\Sigma^{1,eq}}$ un conjunto de tuplas funcionales (parciales) que determinan objetos de $\mathcal{C}^1$. El soporte para el género oculto $imp_{\Sigma^{2,eq}}$ en este objeto también será un conjunto de tuplas funcionales (parciales), que determinan ahora objetos de $\mathcal{C}^2$. La función correspondiente a la operación de coerción aplicará a cada tupla de funciones de $imp_{\Sigma^{2,eq}}$ la tupla que se obtiene *olvidando* las funciones correspondientes a las operaciones de Ω^2 que no estén en Ω^1.

5.5.2 Objeto final

Sean la signatura $\Sigma^2 = (S^2, \Omega^2)$, el dominio de datos D^2 y el dominio de definición dom^2 tal y como los hemos definido en el apartado anterior a partir de una signatura Σ^1, con dominio de datos D^1 y dominio de definición dom^1. Sean $\mathcal{C}^2_{imp}$ y $\mathcal{C}^1_{imp}$ las subcategorías de $HPAlg^{D^2,dom^2}(\Sigma^{2,eq}_{imp})$ y $HPAlg^{D^1,dom^1}(\Sigma^{1,eq}_{imp})$ que también hemos considerado en dicho apartado. Sea por último $A^{1,can}$ el objeto final que obtuvimos para la categoría $\mathcal{C}^1_{imp}$. Entonces, podemos definir un objeto de $\mathcal{C}^2_{imp}$, que denotamos por $A^{2,can}$ de la siguiente forma:

- $A^{2,can}|\Sigma^{1,eq}_{imp} = A^{1,can}$,
- $A^{2,can}_s = D^2_s$, para cada género $s \in S^2 \setminus S^1$, y $d_{A^{2,can}} = d$, para cada constante $d\colon \to s$ de $D^2 \setminus D^1$,
- $A^{2,can}_{imp_{\Sigma^{2,eq}}} = \{(f_\omega)_{\omega \in \Omega^{2,eq}} \mid \langle D^2, (f_\omega, dom^2_\omega)_{\omega \in \Omega^{2,eq}} \rangle \in \mathcal{C}^2\}$,
- $imp\_\omega_{A^{2,can}}((f_\delta)_{\delta \in \Omega^{2,eq}}, d_1, \ldots, d_n) = f_\omega(d_1, \ldots, d_n)$, para todo $(f_\delta)_{\delta \in \Omega^{2,eq}} \in A^{2,can}_{imp_{\Sigma^{2,eq}}}$ y $(d_1, \ldots, d_n) \in dom^2_\omega$, para cada $imp\_\omega\colon imp_{\Sigma^{2,eq}} s_1 \ldots s_n \to s \in \Omega^{2,eq}_{imp} \setminus \Omega^{1,eq}_{imp}$, $n \geq 0$.
- $(coer^{imp_{\Sigma^{2,eq}}}_{imp_{\Sigma^{1,eq}}})_{A^{2,can}}((f_\delta)_{\delta \in \Omega^{2,eq}}) = (f_\delta)_{\delta \in \Omega^{1,eq}}$, para cada $(f_\delta)_{\delta \in \Omega^{2,eq}} \in A^{can}_{imp_{\Sigma^{2,eq}}}$.

De la propia definición se obtiene que este objeto pertenece a la categoría $\mathcal{C}^2_{imp}$. Es más, esta álgebra es final dentro de dicha categoría. Recogemos esta afirmación en el siguiente resultado, cuya demostración es similar a la demostración que obtuvimos en el capítulo anterior cuando consideramos un funtor entre dos categorías.

Teorema 5.5.1. *La $\Sigma_{imp}^{2,eq}$-álgebra $A^{2,can}$, cuyo reducto sobre $\Sigma_{imp}^{1,eq}$ corresponde al objeto final en $\mathcal{C}_{imp}^1$, es objeto final en $\mathcal{C}_{imp}^2$.*

Debemos observar el paralelismo entre este objeto final y el que obteníamos en el capítulo anterior para representar un funtor entre dos categorías. No obstante, en esta ocasión el objeto final incorpora características de la programación orientada a objetos como es la reutilización de una estructura para construir otra. Además, se tiene polimorfismo en las operaciones, de modo que las funciones de la estructura "madre" pueden utilizarse sobre la estructura "hija", simplemente olvidando las operaciones que se incorporan. Este objeto refleja fielmente la forma en que han sido implementadas las estructuras de datos en Kenzo (ver [34] o [33]).

5.5.3 Signatura oculta con un único género oculto

Si decidimos declarar $imp_{\Sigma^{1,eq}}$ como género visible, entonces las cosas son más sencillas. En este caso la signatura $\Sigma_{imp}^{2,eq}$ es una signatura deconstructora y podemos aplicarle los resultados generales para este tipo particular de signaturas de [58, 56] que hemos adaptado en el capítulo anterior para trabajar con signaturas con igualdad. No obstante, en este caso debemos completar el dominio de datos con un nuevo conjunto $D^2_{imp_{\Sigma^{1,eq}}}$. Los elementos de $D^2_{imp_{\Sigma^{1,eq}}}$ pueden interpretarse como los elementos base sobre los que construir estructuras de $\mathcal{C}_{imp}^2$. Si tenemos en mente esta interpretación, es claro que un buen candidato puede ser el conjunto soporte para el género $imp_{\Sigma^{1,eq}}$ del objeto final $A^{1,can}$ que obteníamos en $\mathcal{C}_{imp}^1$. El hecho de que habitualmente se definan soportes para géneros visibles a través de modelos iniciales (por ejemplo, las combinaciones para los complejos de cadenas o los símplices abstractos para los conjuntos simpliciales), y para este elemento se utilice un modelo final refleja la distinta naturaleza entre unos y otros géneros visibles. Mientras que los primeros especifican *elementos* de una estructura y por ello es conveniente tener sólo las representaciones de datos que sean generables desde la signatura, el soporte para el género $imp_{\Sigma^{1,eq}}$ especifica *familias de estructuras* que podemos construir sobre estos elementos, y por ello necesitamos tener una representación tan general como sea posible (ver [84] o [54]).

En esta signatura las operaciones que tienen como primer género $imp_{\Sigma^{1,eq}}$ son visibles y las funciones que prefijamos para ellas son las correspondientes a la mismas en el objeto final $A^{1,can}$.

Además, ahora la operación de coerción es una operación deconstructora. Esta operación debe asociar a cada elemento $a \in A_{imp_{\Sigma^{2,eq}}}$, la tupla de funciones de $A^{1,can}_{imp_{\Sigma^{1,eq}}}$ que define, junto con las funciones que se obtienen a partir del resto de funciones ocultas de A con primer argumento fijado en el elemento a, un álgebra de $\mathcal{C}^2$.

Podemos definir nuevamente un objeto particular de la categoría $\mathcal{C}_{imp}^2$ como antes, si consideramos la misma definición del álgebra $A^{2,can}$ de modo que ahora el soporte para el género $imp_{\Sigma^{1,eq}}$ está prefijado por ser visible. Obtenemos entonces el siguiente teorema.

Teorema 5.5.2. *La categoría $\mathcal{C}^2_{imp}$ sobre la signatura $\Sigma^{2,eq}_{imp}$ con un único género oculto $imp_{\Sigma^{2,eq}}$ tiene objeto final.*

Debemos observar que los objetos finales que hemos obtenido en los dos últimos teoremas son exactamente iguales si se consideran como álgebras estándar (no ocultas). Obviamente los morfismos finales para ambas categorías son distintos. Sin embargo, la interpretación semántica que podemos hacer de ambas situaciones presenta matices diferentes. En la primera tenemos dos géneros ocultos y podemos interpretar la relación entre ellos como una relación de *reuso*. En ella tenemos dos estructuras que están situadas al mismo nivel semántico y una se utiliza para construir la otra. En la segunda se produce más bien una relación de *uso*. Una estructura ya construida y fijada sirve como soporte de datos para añadirle componentes y formar la segunda estructura. Podemos encontrar una discusión entre las diferencias entre *uso* y *reuso* en [66]. Si queremos respetar la herencia que ha sido utilizada en Kenzo para construir este tipo de estructuras, debemos optar por la interpretación de reuso, más próxima a la *herencia*, que por la de uso, más próxima a una relación de *cliente*. No obstante, como se cita en [66], ambas pueden considerarse como interpretaciones de la herencia.

5.6 Semigrupos, monoides, grupos, grupos abelianos...

Para ilustrar la construcción anterior vamos a utilizar estructuras sencillas y ampliamente conocidas como pueden ser los semigrupos, monoides, grupos y grupos abelianos. En [91], Sergeraert utiliza ejemplos parecidos para tratar de explicar la implementación del diagrama de herencia de Kenzo. En realidad la relación que se tiene entre estas estructuras es la misma que se da entre las estructuras de Kenzo, y el uso de la herencia se justifica de la misma forma para ambas situaciones.

Partimos entonces de las categorías $SGRP$, MND, GRP y $AGRP$, que corresponden respectivamente a las categorías de semigrupos, monoides, grupos y grupos abelianos.

Una signatura `SGRP` que nos permita especificar los semigrupos está formada por un único género g, y una única operación $prd\colon g\ g \to g$. Una signatura `MND` para los monoides consiste en la signatura `SGRP` junto con una nueva operación $unt\colon \ \to g$. Una signatura `GRP` para los grupos consiste en la signatura `MND` junto con una nueva operación $inv\colon g \to g$. Por último, una signatura `AGRP` que permite especificar los grupos abelianos es la propia signatura `GRP`.

Como siempre a lo largo de esta memoria, no pretendemos especificar todos los semigrupos, sino que debemos limitarnos a aquéllos que se puedan construir sobre un conjunto de datos $D = \{D_g\}$ prefijado. Para concentrarnos en el proceso de herencia entre estas estructuras vamos a limitarnos a considerar álgebras totales para estas signaturas. Además, tampoco permitiremos en esta ocasión establecer equivalencias en el soporte de datos anterior, de modo que no será necesario construir las signaturas con igualdad asociadas a estas signaturas, o más precisamente, exigimos que la igualdad que se define para ellas sea la igualdad literal.

De esta forma, las álgebras que consideramos para la signatura $\mathtt{SGRP}$ forman la categoría $Alg^D(SGRP)$, que recordamos tiene por objetos las $\mathtt{SGRP}$-álgebras con soporte fijado por D y cuyos morfismos son las identidades. De estas álgebras nos quedamos con una subcategoría $\mathcal{C}^{\mathtt{SGRP}}$ tal que cada álgebra A de la subcategoría verifique que para cada $a, b, c \in D_g$,

$$prd_A(a, prd_A(b, c)) = prd_A(prd_A(a, b), c),$$

es decir, cumple la propiedad asociativa que define el concepto matemático de *semigrupo*.

Ahora, para la signatura $\mathtt{MND}$ nos quedamos con una subcategoría $\mathcal{C}^{MND}$ de $Alg^D(\mathtt{MND})$ tal que cada álgebra A de esta subcategoría cumpla que $A|\mathtt{SGRP} \in \mathcal{C}^{\mathtt{SGRP}}$ y además para cada $a \in D_g$,

$$prd_A(a, unt_A) = prd_A(unt_A, a) = a.$$

Similarmente, para la signatura $\mathtt{GRP}$ nos quedamos con una subcategoría $\mathcal{C}^{GRP}$ de $Alg^D(\mathtt{GRP})$ tal que cada álgebra A de esta subcategoría cumpla que $A|\mathtt{MND} \in \mathcal{C}^{\mathtt{MND}}$ y además para cada $a \in D_g$,

$$prd_A(a, inv_A(a)) = prd_A(inv_A(a), a) = unt_A.$$

Por último, para la signatura $\mathtt{AGRP}$ nos quedamos con una subcategoría $\mathcal{C}^{AGRP}$ de $Alg^D(\mathtt{AGRP})$ tal cada álgebra A de esta subcategoría cumpla que $A|\mathtt{GRP} \in \mathcal{C}^{\mathtt{GRP}}$ y además para cada $a, b \in D_g$,

$$prd_A(a, b) = prd_A(b, a).$$

Nosotros no pretendemos especificar una única estructura sino familias de estructuras. Entonces, para especificar familias de semigrupos construimos la signatura $\mathtt{SGRP}_{imp}$ que está formada por dos géneros g, $imp_{\mathtt{SGRP}}$ y una operación $imp\_prd\colon imp_{\mathtt{SGRP}} g\ g \to g$. Esta signatura es una signatura oculta con un único genero oculto $imp_{\mathtt{SGRP}}$, sobre el álgebra de datos (D, C), con C un conjunto de constantes, una por cada elemento de D.

Las álgebras que consideramos para esta signatura forman la categoría $HAlg^D(\mathtt{SGRP}_{imp})$, cuyos objetos son las $\mathtt{SGRP}_{imp}$-álgebras ocultas y cuyos morfismos son los $\mathtt{SGRP}_{imp}$-homomorfismos débiles ocultos. De estas álgebras nos quedamos con la subcategoría plena $\mathcal{C}^{\mathtt{SGRP}}_{imp}$ formada por las álgebras A que verifican que cada para cada elemento $I \in A_{imp_{\mathtt{SGRP}}}$ y cada $a, b, c \in D_g$,

$$imp\_prd_A(I, a, imp\_prd_A(I, b, c)) = imp\_prd_A(I, imp\_prd_A(I, a, b), c).$$

Ahora, para especificar familias de monoides construimos la signatura $\mathtt{MND}_{imp}$ que incorpora a $\mathtt{SGRP}_{imp}$ un género $imp_{\mathtt{MND}}$ y dos operaciones $imp\_unt\colon imp_{\mathtt{MND}} \to g$, $coer^{imp_{\mathtt{MND}}}_{imp_{\mathtt{SGRP}}}\colon imp_{\mathtt{MND}} \to imp_{\mathtt{SGRP}}$. Esta signatura es una signatura oculta que añade a $\mathtt{SGRP}_{imp}$ el genero oculto $imp_{\mathtt{MND}}$. El álgebra de datos para ella sigue siendo (D, C).

Otra vez, según el desarrollo teórico que hemos realizado, nos quedamos con una subcategoría $\mathcal{C}_{imp}^{\mathtt{MND}}$ de $HAlg^D(\mathtt{MND}_{imp})$ que verifique que cada álgebra A de esta categoría cumpla que $A|\mathtt{SGRP}_{imp} \in \mathcal{C}_{imp}^{\mathtt{SGRP}}$ y que para cada elemento $I \in A_{imp_{\mathtt{MND}}}$ y cada $a \in D_g$,

$$imp\_prd_A\big(coer_{imp_{\mathtt{SGRP}}}^{imp_{\mathtt{MND}}}(I), a, imp\_unt_A(I)\big) =$$
$$imp\_prd_A\big(coer_{imp_{\mathtt{SGRP}}}^{imp_{\mathtt{MND}}}(I), imp\_unt_A(I), a\big) = a.$$

Similarmente, para especificar familias de grupos construimos la signatura oculta $\mathtt{GRP}_{imp}$ que incorpora a $\mathtt{MND}_{imp}$ un género oculto $imp_{\mathtt{GRP}}$ y dos operaciones $imp\_inv\colon imp_{\mathtt{GRP}}g \to g$, $coer_{imp_{\mathtt{MND}}}^{imp_{\mathtt{GRP}}}\colon imp_{\mathtt{GRP}} \to imp_{\mathtt{MND}}$. Entonces, nos quedamos con una subcategoría de álgebras $\mathcal{C}_{imp}^{\mathtt{GRP}}$ de $HAlg^D(\mathtt{GRP}_{imp})$ que verifique que cada álgebra A de esta categoría cumpla que $A|\mathtt{MND}_{imp} \in \mathcal{C}_{imp}^{\mathtt{MND}}$ y que para cada elemento $I \in A_{imp_{\mathtt{GRP}}}$ y cada $a \in D_g$:

$$imp\_prd_A\big(coer_{imp_{\mathtt{SGRP}}}^{imp_{\mathtt{MND}}}(coer_{imp_{\mathtt{MND}}}^{imp_{\mathtt{GRP}}}(I)), a, imp\_inv_A(a)\big) =$$
$$imp\_prd_A\big(coer_{imp_{\mathtt{SGRP}}}^{imp_{\mathtt{MND}}}(coer_{imp_{\mathtt{MND}}}^{imp_{\mathtt{GRP}}}(I)), imp\_inv_A(a), a\big) =$$
$$imp\_unt_A\big(coer_{imp_{\mathtt{MND}}}^{imp_{\mathtt{GRP}}}(I)\big).$$

Por último, para especificar familias de grupos abelianos construimos la signatura $\mathtt{AGRP}_{imp}$ que incorpora a $\mathtt{GRP}_{imp}$ un género oculto $imp_{\mathtt{AGRP}}$ y una aplicación $coer_{imp_{\mathtt{GRP}}}^{imp_{\mathtt{AGRP}}}\colon imp_{\mathtt{AGRP}} \to imp_{\mathtt{GRP}}$. Entonces, nos quedamos con una subcategoría $\mathcal{C}_{imp}^{\mathtt{AGRP}}$ de $HAlg^D(\mathtt{AGRP}_{imp})$ que verifique que cada álgebra A de esta categoría cumpla que $A|\mathtt{GRP}_{imp} \in \mathcal{C}_{imp}^{\mathtt{GRP}}$ y que para cada elemento $I \in A_{imp_{\mathtt{AGPR}}}$ y cada $a, b \in D_g$:

$$imp\_prd_A\big(coer_{imp_{\mathtt{SGRP}}}^{imp_{\mathtt{MND}}}(coer_{imp_{\mathtt{MND}}}^{imp_{\mathtt{GRP}}}(coer_{imp_{\mathtt{GRP}}}^{imp_{\mathtt{AGRP}}}(I))), a, b\big) =$$
$$imp\_prd_A\big(coer_{imp_{\mathtt{SGRP}}}^{imp_{\mathtt{MND}}}(coer_{imp_{\mathtt{MND}}}^{imp_{\mathtt{GRP}}}(coer_{imp_{\mathtt{GRP}}}^{imp_{\mathtt{AGRP}}}(I))), b, a\big).$$

Debemos observar cómo para construir una signatura aprovechamos la estructura de otra. Cada signatura se caracteriza por tener un género destacado que denotamos con el símbolo *imp* con subíndice la misma signatura. Cuando una signatura hereda de otra, se produce una relación de orden entre sus géneros destacados, que en la práctica se traduce en la inclusión en la primera de una operación de coerción. Esta operación de coerción es la que permite obtener polimorfismo en las operaciones, de forma que podemos utilizar una operación que involucra argumentos de un género determinado sobre un elemento de un subgénero. Estas operaciones de coerción actúan de forma implícita en algunos trabajos, como por ejemplo en la forma de construir estructuras algebraicas en Coq [37]. Esto simplifica notablemente la notación sintáctica; no obstante, nosotros preferimos hacer explícita la intervención de estas operaciones como forma de representar la relación de herencia entre ellas.

Para terminar este ejemplo damos una descripción de los objetos finales que se tienen para cada categoría. Comenzamos con el objeto final de la categoría $\mathcal{C}_{imp}^{\mathtt{SGRP}}$ que denotamos por A^{SGRP}. Este objeto está definido por:

· $A^{SGRP}_{imp_{\mathtt{SGRP}}} := \{(*\colon D_g \times D_g \to D_g) \mid \langle D_g, * \rangle \in \mathcal{C}^{\mathtt{SGRP}}\}$,

· $imp\_prd_{A^{SGRP}}((*), a, b) = *(a, b)$, para cada $(*) \in A^{SGRP}_{imp_{\mathtt{SGRP}}}$ y cada $a, b \in D_g$.

El objeto final A^{MND} de la categoría $\mathcal{C}^{\mathtt{MND}}_{imp}$ está definido por:

· $A^{MND}|\mathtt{SGRP}_{imp} := A^{MND}$,

· $A^{MND}_{imp_{\mathtt{MND}}} := \{(*\colon D_g \times D_g \to D_g, e\colon \to D_g) \mid \langle D_g, (*, e) \rangle \in \mathcal{C}^{\mathtt{MND}}\}$,

· $imp\_unt_{A^{MND}}((*, e)) = e$, para cada $(*, e) \in A^{SGRP}_{imp_{\mathtt{MND}}}$,

· $(coer^{imp_{\mathtt{MND}}}_{imp_{\mathtt{SGRP}}})_{A^{MND}}((*, e)) := (*)$, para cada $(*, e) \in A^{MND}_{imp_{\mathtt{MND}}}$.

El objeto final A^{GRP} de la categoría $\mathcal{C}^{\mathtt{GRP}}_{imp}$ está definido por:

· $A^{GRP}|\mathtt{MND}_{imp} := A^{MND}$,

· $A^{GRP}_{imp_{\mathtt{GRP}}} := \{(*\colon D_g \times D_g \to D_g, e\colon \to D_g, -\colon D_g \to D_g) \mid \langle D_g, (*, e, -) \rangle \in \mathcal{C}^{\mathtt{GRP}}\}$,

· $imp\_inv_{A^{GRP}}((*, e, -), a) = -(a)$, para cada $(*, e, -) \in A^{GRP}_{imp_{\mathtt{GRP}}}$ y cada $a \in D_g$,

· $(coer^{imp_{\mathtt{GRP}}}_{imp_{\mathtt{MND}}})_{A^{GRP}}((*, e, -)) := (*, e)$, para cada $(*, e, -) \in A^{GRP}_{imp_{\mathtt{GRP}}}$.

Por último, el objeto final A^{AGRP} de la categoría $\mathcal{C}^{\mathtt{AGRP}}_{imp}$ está definido por:

· $A^{AGRP}|\mathtt{GRP}_{imp} := A^{GRP}$,

· $A^{AGRP}_{imp_{\mathtt{AGRP}}} := \{(*\colon D_g \times D_g \to D_g, e\colon \to D_g, -\colon D_g \to D_g) \mid \langle D_g, (*, e, -) \rangle \in \mathcal{C}^{\mathtt{AGRP}}\}$,

· $(coer^{imp_{\mathtt{AGRP}}}_{imp_{\mathtt{GRP}}})_{A^{AGRP}}((*, e, -)) := (*, e, -)$, para cada $(*, e, -) \in A^{AGRP}_{imp_{\mathtt{AGRP}}}$.

En todos estos objetos finales podemos observar cómo cada operación de coerción puede entenderse como un tipo de olvido de una parte de la estructura.

5.7 Especificación de los FD-complejos

Kenzo utiliza la estructura de FD-complejo definida por Eilenberg y MacLane para trabajar con conjuntos simpliciales. Para implementar esta estructura se ha definido una clase llamada `SIMPLICIAL-SET` que se construye por herencia a partir de la clase `CHAIN-COMPLEX` que corresponde a los complejos de cadenas (ver [34] y Sección 5.2). En esta sección trataremos de especificar la relación que existe entre ambas.

En el capítulo anterior hemos dado especificaciones de los complejos de cadenas y de los conjuntos simpliciales, así como del funtor (sólo a nivel de objetos) que construye el complejo de cadenas canónicamente asociado a un conjunto simplicial. En esta sección utilizaremos la misma técnica que usamos en el capítulo anterior para llegar a dichas especificaciones, pero de modo que ahora aprovecharemos las partes comunes en ambas para construir una a partir de la otra. Así, definimos dos signaturas Σ^{cc} y Σ^{ss} con idea de especificar los complejos de cadenas por un lado y los conjuntos simpliciales por otro.

Para especificar la estructura de FD-complejo damos una nueva signatura Σ^{cs} que incluye a Σ^{cc} y además contiene dos nuevos géneros; que llamamos nat y asm, y dos nuevas operaciones; $face\colon nat\ nat\ asm \rightarrow asm$ y $dgn\colon nat\ nat\ asm \rightarrow asm$. Es decir, ampliamos Σ^{cc} con algunos géneros y operaciones de Σ^{ss}. Debemos observar que el género que hemos destinado para los símplices geométricos es el género gnr de la signatura de los complejos de cadenas. Lo anterior está de acuerdo con la definición de FD-complejo, en la que los símplices geométricos del conjunto simplicial sirven de generadores para el complejo de cadenas. Para que esto sea efectivo debemos tomar como dominio de datos para el género gnr en el complejo de cadenas el conjunto $D^{cc}_{gnr} = \cup_{p\in\mathbb{Z}} G_p$ tal que $G_p = \emptyset$ si $p < 0$. A partir de esta última condición técnica podemos obtener una representación homogénea de los símplices geométricos y los generadores del complejo de cadenas. Entonces, para la signatura Σ^{cs} fijamos el dominio de datos D^{cs} a partir del dominio de datos D^{cc}, y para los géneros nat y asm se define un dominio de datos de modo similar a como hicimos para Σ^{ss}, considerando como soporte de los símplices geométricos a D^{cc}_{gnr}.

El siguiente paso es construir la signatura con igualdad $\Sigma^{cs,eq}$. Para ella fijamos el dominio de definición dom^{cs} que obtenemos a partir de dom^{cc} y dom^{ss}. Entonces, la subcategoría de álgebras $\mathcal{C}^{cs}$ de $P\tilde{A}lg^{D^{cs},dom^{cs}}(\Sigma^{cs,eq})$ que nos interesa está constituida por aquellas álgebras A que verifican la siguiente propiedad: $A/_{eq^{cs}_A}$ define un FD-complejo de modo que el cociente $(A|\Sigma^{cc,eq})/_{eq^{cc}_A}$ es el complejo de cadenas que se obtiene por olvido a partir de esta estructura de FD-complejo. Como tenemos que los generadores del complejo de cadenas son los símplices geométricos del conjunto simplicial entonces para que se dé esta condición es suficiente con exigir que la diferencial del complejo de cadenas se defina a través de las caras del conjunto simplicial de la forma conocida (ver página 166, por ejemplo).

Bajo estas condiciones, podemos aplicar a este caso la construcción general de la Sección 5.5. Así, por un lado definimos una signatura oculta $\Sigma^{cc,eq}_{imp}$ y nos restringimos a una subcategoría $\mathcal{C}^{cc}_{imp}$ de $HPAlg^{D^{cc},dom^{cc}}(\Sigma^{cc,eq}_{imp})$ que coincide con la categoría que definimos en el capítulo anterior para especificar familias de complejos de cadenas. Por otro lado, definimos la signatura oculta $\Sigma^{cs,eq}_{imp}$ que está formada por $\Sigma^{cc,eq}_{imp}$ junto con un nuevo género oculto, denotado por $imp_{\Sigma^{cs,eq}}$, y dos visibles, nat y asm, y presenta como nuevas operaciones $imp\_face\colon imp_{\Sigma^{cs,eq}} nat\ nat\ asm \rightarrow asm$ e $imp\_dgn\colon imp_{\Sigma^{cs,eq}} nat\ nat\ asm \rightarrow asm$, junto con las operaciones de igualdad para los nuevos géneros visibles.

Debemos observar que para construir una signatura que permita especificar familias de FD-complejos, hemos reutilizado la signatura a través de la cual especificamos familias de complejos de cadenas, a la cual le hemos añadido las operaciones que permiten definir un conjunto simplicial. Ahora, para hacer efectiva esta reutilización debemos establecer el siguiente orden entre los dos géneros ocultos: $imp_{\Sigma^{cs,eq}} < imp_{\Sigma^{cc,eq}}$. Esto se traduce en la inclusión en la signatura $\Sigma^{cs,eq}_{imp}$ la siguiente operación coerción:

$$coer^{imp_{\Sigma^{cs,eq}}}_{imp_{\Sigma^{cc,eq}}} : imp_{\Sigma^{cs,eq}} \to imp_{\Sigma^{cc,eq}}.$$

Con esta operación se completa la signatura $\Sigma^{cs,eq}_{imp}$. El álgebra de datos y el dominio de definición que fijamos para esta signatura vienen dados por D^{cs} y dom^{cs}.

Ahora sólo falta restringirnos a la subcategoría $\mathcal{C}^{cs}_{imp}$ de $HPAlg^{D^{cs},dom^{cs}}(\Sigma^{cs,eq}_{imp})$ tal que cada álgebra A de esta categoría verifique que $A|\Sigma^{cc,eq}_{imp} \in \mathcal{C}^{cc}_{imp}$ y para cada $a \in A_{imp_{\Sigma^{cs,eq}}}$, la $\Sigma^{cs,eq}_{imp}$-álgebra que se obtiene a partir de A si fijamos, para cada operación deconstructora con género $imp_{\Sigma^{cs,eq}}$ el primer argumento en a, y si es con género $imp_{\Sigma^{cc,eq}}$ el primer argumento en $(coer^{imp_{\Sigma^{cs,eq}}}_{imp_{\Sigma^{cc,eq}}})_A(a)$, verifique que es un objeto de $\mathcal{C}^{cs}$. Es decir, nos restringimos a aquellas álgebras cuyo soportes para los géneros destacados sirvan de índice para determinar FD-complejos.

El objeto final de la categoría $\mathcal{C}^{cs}_{imp}$ consta de tuplas de funciones como elementos de los soportes para cada uno de sus géneros ocultos. En particular, para el género $imp_{\Sigma^{cc,eq}}$, la tupla está formada por una función por cada operación de $\Sigma^{cc,eq}_{imp}$, es decir, por tuplas $(+, 0, -, *, df, =_{int}, =_{gnr}, =_{cmb})$, tal que definan un complejo de cadenas sobre D^{cc}. No obstante, como comentamos en el capítulo anterior, este objeto final admite una representación más simple ya que las funciones 0, $-$, $*$, $=_{int}$ están prefijadas a partir del dominio de datos y las funciones $+$, $=_{cmb}$ se pueden definir a partir de $=_{gnr}$, por lo que podemos prescindir de ellas y no aparecer en este soporte en el objeto final. Es decir, la tupla que representa a un complejo de cadenas en el objeto final está formada por dos funciones df y $=_{gnr}$. Para el género $imp_{\Sigma^{cc,eq}}$, la tupla está formada por las operaciones de la tupla para $imp_{\Sigma^{cc,eq}}$ más una función por cada operación deconstructora nueva en $\Sigma^{cs,eq}_{imp}$, es decir, por tuplas $(df, =_{gnr}, fc, dg, =_{nat}, =_{asm})$. Nuevamente, las funciones dg y $=_{nat}$ están prefijadas a partir del dominio de datos y la función $=_{asm}$ está definida a partir de $=_{gnr}$. Además, la función df se define a partir de la función fc. Entonces, la tupla que representa a un FD-complejo en el objeto final está formada por dos funciones $=_{gnr}$ y fc. La función de coerción permite utilizar las operaciones del complejo de cadenas sobre la estructura de FD-complejo, de forma que se olvidan las operaciones que se añaden a esta última para recuperar la estructura subyacente de complejo de cadenas.

Podemos comparar los soportes que hemos obtenido para este objeto final con la estructuras de complejos de cadenas y conjuntos simpliciales de Kenzo estudiados en la Sección 5.2. La conclusión que se obtiene es que esta modelización es adecuada para especificar la herencia presente en Kenzo entre estas dos estructuras.

5.8 Problemas abiertos

En esta sección introduciremos algunas cuestiones que quedan por resolver y que van más allá de los propósitos de esta tesis.

5.8.1 Diagrama de clases de Kenzo

Si observamos el diagrama de clases de Kenzo [34, 33, 91] que mostramos en la Figura 5.4 podemos ver que todavía queda mucho trabajo por realizar.

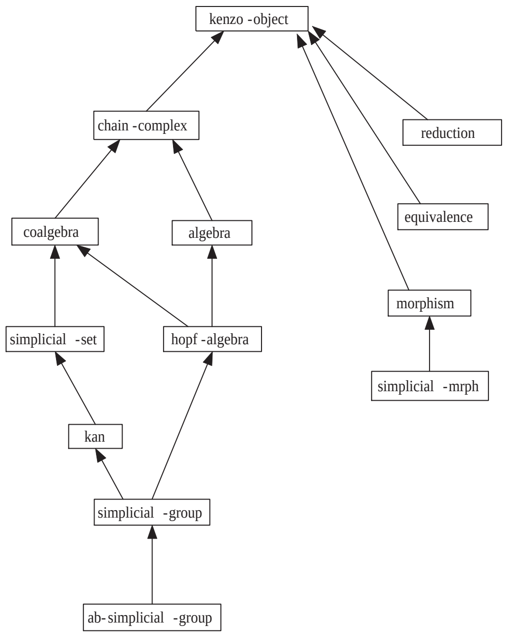

Figura 5.4: Diagrama de clases de Kenzo

En la parte de la izquierda del diagrama están situadas las principales categorías matemáticas que se utilizan en Topología Algebraica combinatoria. Como sabemos un *complejo de cadenas* es un módulo graduado diferencial; un *álgebra* es un complejo de cadenas con una estructura multiplicativa compatible, lo mismo para una *coálgebra* pero en este caso con una estructura comultiplicativa. Si una estructura multiplicativa y comultiplicativa se unen de modo que son compatibles, entonces obtenemos un *álgebra de Hopf*, y así continuamos construyendo el resto de las estructuras.

Podemos observar que la mayor parte de las relaciones entre estas estructuras se pueden especificar a través de la técnica de modelado que hemos presentado en este trabajo. Sin embargo, otra parte no. Las relaciones que no cubrimos corresponden con *herencia múltiple*, y sobre todo por la presencia de *diamantes* dentro de este diagrama.

Además, a lo largo de toda la tesis hemos estado trabajando con *objetos* de categorías matemáticas: semigrupos, grupos, conjuntos simpliciales... Pero, ¿qué ocurre con los morfismos? Algunas de las clases de la parte derecha del diagrama están dedicadas a implementar los morfismos que necesita Kenzo. La cuestión es si estas clases admiten una modelización parecida a la que hemos realizado sobre las clases de objetos.

Otra cuestión que puede resultar interesante es estudiar si podemos expresar el modelado que hemos realizado de las relaciones entre estructuras, y en particular de la herencia, dentro de un marco institucional, de modo similar a como hemos hecho para la construcción *imp*.

Por último, en este trabajo tratado la formalización de aplicaciones entre estructuras que tienen un único argumento. Podemos plantearnos generalizar este estudio de modo que incluya aplicaciones más complejas, que trabajen sobre más de un argumento. Por ejemplo, en Kenzo se trabaja con el operador que construye el producto tensorial de dos complejos de cadenas.

En las próximas secciones esbozaremos posibles formas de abordar estas cuestiones.

5.8.2 Herencia múltiple

Comenzamos con un problema interesante como es la herencia múltiple y los posibles diagramas en forma de *diamante* que se pueden originar a partir de ella. La herencia múltiple plantea problemas complicados por lo que muchos sistemas tratan de evitarla (por ejemplo, ver [94, 37]).

CLOS admite la herencia múltiple [93] y Kenzo la utiliza para definir sus estructuras. Por ejemplo, si observamos el diagrama, tenemos que la estructura *álgebra de Hopf* se construye por herencia múltiple a partir de las estructuras coálgebra y álgebra. Además como estas dos estructuras heredan de la estructura complejo de cadenas se forma un diamante dentro del diagrama. Un segundo ejemplo lo tenemos con la estructura *grupo simplicial*, que hereda de la estructuras de Kan y de álgebra de Hopf, formándose un segundo diamante con vértice superior en las coálgebras. Todavía existe un tercer diamante que engloba a estos dos últimos.

Para especificar la herencia múltiple podemos tratar de generalizar la técnica que hemos desarrollado para la herencia simple. Para ello podemos pensar en incluir en la signatura "hija" la unión de las operaciones de cada una de las posibles signaturas "madre" y además una operación de coerción por cada signatura que herede. De esta forma conseguimos reutilización sintáctica entre las signaturas. Además, en la signatura "hija" no se produce duplicación de operaciones ya que en el caso de una situación de diamante en los que dos posibles signaturas "madres" compartan una signatura "abuela" común, entonces como no definimos de nuevo las operaciones de ésta sino que simplemente se incorporan en las signaturas "madres", la unión de ambas para formar la "hija" determina que aparezca cada operación una única vez en dicha signatura. La presencia de coerción explícita en las signaturas nos permite utilizar en cada caso la operación de coerción que interese. Con coerción implícita el problema es más complicado y por ejemplo los autores de [37] no plantean solución alguna.

Además, debemos resolver el problema de decidir cuáles de los posibles caminos que conducen a través de las coerciones en un diagrama con forma de diamante debemos utilizar en cada caso. El problema se plantea cuando queremos utilizar una operación que está presente en una estructura "abuela" aplicada a un elemento de estructura "hija". En este caso, podemos simplemente exigir a nivel semántico una condición más: ambos caminos deben conducir a la misma operación. Para ello es suficiente con imponer que las posibles composiciones de coerciones aplicadas a un elemento del género destacado de la "hija" determinen el mismo elemento del género destacado de la "abuela". Imponer esta condición en un diamante del que conocemos su estructura es relativamente fácil a partir de las coerciones. Sin embargo, no parece sencillo expresar de manera natural esta condición para un diagrama general, ya que a partir de una estructura no conocemos, en principio, el árbol de herencia que la sostiene.

5.8.3 Especificación de los morfismos

La implementación que se hace en Kenzo (y también en EAT) de los morfismos presenta una solución muy elegante gracias a la programación funcional y al modo en que han sido implementadas las estructuras. En el diagrama de clases de Kenzo podemos observar dos clases cuyas instancias representan morfismos: la clase *morphism* que corresponde a morfismos de complejos de cadenas y la clase *simplicial-mrph* que es descendiente de la anterior.

Para tener determinada una instancia de morfismo necesitamos esencialmente tener una estructura *origen* (dominio) del morfismo, una estructura *destino* (codominio) del morfismo y una función que lleve cada elemento de la estructura origen a un elemento de la estructura final. Por ello, una clase para un morfismo consta fundamentalmente de tres campos: dos destinados a almacenar instancias de estructuras y un tercer campo funcional. Debemos observar como CLOS ofrece el ambiente adecuado para definir esta clase.

Si tratamos de especificar una clase de morfismos podemos pensar en utilizar una técnica similar a la que hemos utilizado para los objetos. Por ejemplo, supongamos que queremos obtener una signatura cuyos modelos sean morfismos de grupos. Entonces, podemos construir una signatura que contenga dos géneros: uno para los elementos de los grupos y otro para los propios grupos, y tres operaciones: una que abstrae la función que define el morfismo y dos constantes, que determinen los grupos origen y destino del morfismo. Debemos observar que tenemos las herramientas adecuadas para definir esta signatura, ya que hemos definido un género cuyo soporte está dedicado a representar grupos: el género destacado para la signatura de grupo. Si ahora pensamos en una posible signatura que permita especificar no un único morfismo sino una familia de ellos, podemos aplicar la operación *imp* a la anterior signatura. En este caso, obtenemos una signatura con tres géneros, denotados por g, $imp_{\texttt{GRP}}$ e $imp_{\texttt{M-GRP}}$, y tres operaciones $f\colon imp_{\texttt{M-GRP}}\, g\ g \to g$, $source\colon imp_{\texttt{M-GRP}} \to imp_{\texttt{GRP}}$ y $target\colon imp_{\texttt{M-GRP}} \to imp_{\texttt{GRP}}$. Si tratamos de ver esta signatura como signatura oculta, a la hora de declarar los posibles géneros visibles y ocultos tenemos dos posibilidades. Mientras que el género $imp_{\texttt{M-GRP}}$ es claro que debe ser oculto, para el género $imp_{\texttt{GRP}}$ podemos plantear una discusión parecida a la mantenida al hablar de la especificación de la herencia, sobre si es visible u oculto. En este caso, se tiene una clara relación de *uso* entre las dos estructuras: el morfismo de grupos no es un grupo, sino que utiliza grupos en su construcción (ver nuevamente [66]). Por ello, parece razonable pensar que la estructura grupo debe ser previa a la de morfismo y ser visible para ella. Como comentamos en aquella discusión, un buen candidato a servir como dominio de datos para este género es el soporte para dicho género en el objeto final.

Las cosas pueden complicarse un poco más si pensamos en la posible herencia entre clases que representan morfismos. Podemos pensar en definir por ejemplo una clase para los morfismos de los anillos que herede de la clase para representar los morfismos de los grupos. En esencia, las componentes que necesitamos para un morfismo de anillos son las mismas que para un morfismo de grupos: dos estructuras, ahora anillos, y una aplicación entre los elementos de esas estructuras. En este caso, no podemos reutilizar la signatura para los morfismos de grupos, tal y como la tenemos definida, para los morfismos de los anillos. Esto es debido a que las operaciones que nos permiten obtener el grupo origen y final ahora deben definir otras estructuras más específicas como son los anillos y la posible operación de coerción entre anillos y grupos va ahora en dirección opuesta a la que necesitamos. Para resolver este punto hace falta más investigación.

Además, el sistema Kenzo presenta en su implementación una característica particular que se puede tratar de especificar. Para implementar las diferenciales de los complejos de cadenas se ha utilizado la estructura de morfismo. Es decir, una diferencial es un endomorfismo del complejo de cadenas que se define a partir de esta diferencial. Esta peculiaridad plantea desde el punto de vista de la especificación que el género destacado de los morfismos es utilizado como visible dentro de la signatura de los complejos de cadenas. Pero la signatura de los morfismos de complejos de cadenas contiene como visible al género destacado de los complejos de cadenas. Entonces podemos entender que este último género está presente como género oculto y visible a la vez en esta signatura. Otra vez hace falta más investigación para aclarar este punto.

5.8.4 Interpretación institucional de la herencia

En el tratamiento que hemos realizado de la herencia no hemos utilizado un marco institucional. Podemos tratar de interpretar dicha herencia como una relación entre instituciones de forma similar a como hemos hecho con la operación *imp*.

Si pensamos en la relación de inclusión que existe entre una signatura hija respecto con respecto a una signatura madre como un morfismo de signaturas inclusión, obtenemos que dicho morfismo es un morfismo oculto bien definido, es decir verifica la propiedad de encapsulamiento. Este hecho da pie a tratar de ubicar estas relaciones de herencia en una institución oculta específica y a buscar una relación entre instituciones que recoja esta relación de herencia de modo similar a como plantemos la construcción *imp.*

5.8.5 Otros funtores entre categorías

En este trabajo hemos analizado las aplicaciones entre objetos de funtores que tienen una única categoría origen. No obstante, en Kenzo y también en EAT existen operaciones entre estructuras que tienen más de un argumento. Por ejemplo, una aplicación que construya el producto cartesiano de dos conjuntos simpliciales o el producto tensorial de dos complejos de cadenas.

Para especificar estos funtores necesitamos incluir en la signatura operaciones que tienen en su argumento más de un género oculto. Estas signaturas suponen una generalización de la teoría oculta de [42] y de las definiciones que hemos utilizado en esta memoria. Desde el punto de vista de la teoría oculta ya hay trabajos que han realizado una generalización de la definición de signatura oculta en este sentido (ver, por ejemplo [45, 27]). Nos tendríamos que basar en estas nuevas definiciones para poder llevar a cabo la especificación de estos funtores. No obstante, en estos artículos se cita que la inclusión de más de un género oculto conlleva en general la pérdida del objeto final en la categoría de álgebras ocultas. Un problema interesante puede consistir en estudiar si este objeto final se pierde también en nuestras signaturas particulares.

Conclusiones y trabajo futuro

El presente trabajo se ha dedicado a aplicar métodos formales en el estudio de los programas de Cálculo Simbólico EAT y Kenzo. Más concretamente, se han utilizado *especificaciones algebraicas* para establecer modelos formales de las estructuras de datos de los citados sistemas.

Para estudiar cada estructura de datos *aisladamente* (trabajo que ya había sido emprendido por otros miembros de nuestro equipo de investigación [58, 57, 72]) se ha utilizado el concepto de institución (Capítulos 2 y 3). A continuación, se han tratado las relaciones entre estructuras de datos en EAT (Capítulo 4), prestando especial atención a la relación entre conjuntos simpliciales y complejos de cadenas (estructuras esenciales en Topología Algebraica, campo de aplicación de EAT y Kenzo). Por último, se ha modelado parte de la organización orientada a objetos de Kenzo, concentrándonos sobre todo en la relación de herencia entre estructuras (Capítulo 5).

En lo relativo al uso de instituciones, una primera conclusión, muy positiva, es que se ha conseguido agrupar, de modo cohesionado, distintos resultados que aparecían separados en publicaciones previas [30], [29]. Se ha obtenido, de este modo, un conocimiento más profundo de las técnicas y conceptos involucrados en nuestro análisis. Sin embargo, el precio a pagar ha sido una cierta rigidez en las instituciones que intervienen en nuestra descripción del proceso de construcción de las estructuras de datos en EAT (la construcción *imp*). Esta rigidez se ve ilustrada por la aparición, en el Capítulo 2, de categorías *discretas* que, hasta cierto punto, vacían de contenido el uso de la Teoría de Categorías. Se ha elegido este tipo de presentación para mostrar un diagrama homogéneo de morfismos entre instituciones que modelan, de manera completa, la construcción *imp* y su relación con las técnicas ocultas y coalgebraicas. En el Capítulo 3, retiramos las restricciones sobre los morfismos en las categorías para retomar un marco más natural de especificación y mostramos con detalle cómo en este caso hay dificultades y limitaciones que impiden encontrar un diagrama homogéneo, similar al expuesto en el capítulo anterior. Se identifican con precisión los puntos delicados, mostrando con ejemplos concretos las obstrucciones para establecer morfismos generales, y se exploran diferentes variantes en la definición tanto de las instituciones como de los morfismos entre ellas. El lector atento encontrará una exhaustiva guía para utilizar las técnicas institucionales, ocultas y coalgebraicas, evitando falsos atajos que pueden confundir al principiante.

Los resultados presentados en los Capítulos 2 y 3, así como su proceso de génesis, nos permiten hacer ciertas reflexiones sobre el uso de instituciones en la tarea de modelización y especificación. Dando por hecho la justeza de los objetivos perseguidos por los creadores de la noción de institución (esencialmente, agrupar bajo un único concepto todas las componentes que intervienen en especificación algebraica), una primera observación es que utilizar (correctamente) las instituciones, es difícil. Más allá de ser una observación banal, esta afirmación queda justificada por los frecuentes errores documentados en la literatura (en ocasiones por autores, y en publicaciones, de primera línea), muy superiores en número a los que se pueden encontrar en otros temas. Algunos de esos errores, relacionados sobre todo con las interacciones entre instituciones y especificaciones ocultas, han sido detectados y descritos en esta memoria.

Una segunda observación tiene que ver con dos aspectos aparentemente contradictorios de las instituciones. Por una parte, son un instrumento excesivamente *liberal*. Y, por otra, su definición es demasiado *rígida*.

Respecto a su carácter excesivamente flexible, señalar, por ejemplo, que sobre la categoría de signaturas no se impone ninguna restricción. Así, cualquier categoría, por enorme o abstracta que sea, puede ser adoptada como categoría de signaturas, sin ninguna referencia al carácter *sintáctico* que solemos asociar con el término *signatura*. De este modo, el usuario del concepto de institución no dispone de intuiciones o directrices que le indiquen cómo diseñar sus distintas componentes. Esto sucede, por ejemplo, al intentar definir una institución para coálgebras. En nuestro caso particular (Capítulo 2), al tratarse de una institución coalgebraica que es la *imagen* de la construcción *imp*, hemos podido trasladar signaturas y sentencias sin dificultades.

La rigidez excesiva de las instituciones proviene de la relación de satisfacibilidad. Es ésta una parte esencial de cada institución, por establecer el vínculo entre el resto de componentes. La relación está expresada como un “si y sólo si”. Pese a que exigir la equivalencia, en lugar de sólo una condición necesaria o suficiente, es natural en los ejemplos que inspiran la definición de institución (la especificación algebraica ecuacional y sus variantes), no lo es en muchas otras ocasiones. Esta limitación ha sido señalada también por otros autores [85, 68]. El mismo problema se traslada a la condición de satisfacibilidad en la noción de morfismo de instituciones. Teniendo en cuenta que un morfismo es una “flecha” (con dirección y sentido, por tanto), parecería más natural exigir tan solo una implicación en lugar de una equivalencia. Ésta ha sido la causa principal de la gran extensión de nuestra presentación de nuestros diagramas de instituciones (véase de nuevo [68] para otra discusión sobre esta limitación de la definición de morfismo).

En los Capítulos 4 y 5 volvemos a un marco de especificación más habitual, sin recurrir al concepto de institución. En el Capítulo 4, utilizamos especificaciones ocultas, pero parciales y con relaciones de equivalencias variables sobre los dominios de datos. Estas extensiones nos sitúan en una posición intermedia entre la especificación algebraica pura (con álgebras totales y dominios de datos completamente fijados) y el análisis de las representaciones e implementaciones de Tipos Abstractos de Datos (al estilo de [49]; en este último campo, además de parcialidad y relaciones de equivalencia, hay que tratar con los problemas de semántica de los lenguajes de programación concretos, como en [58] y [72]). Incluso situándonos en este punto intermedio, los problemas son mucho más complejos y sutiles que en el caso algebraico puro. Esto queda bien ejemplificado en nuestra discusión sobre las interacciones entre fijar los dominios de las operaciones y trabajar con homomorfismos débiles o fuertes. Como nuestros contraejemplos señalan, sin una cuidadosa selección de categorías, podemos encontrarnos en ocasiones sin semántica final para las estructuras de datos funcionales de EAT o Kenzo.

La principal conclusión que puede ser extraída del Capítulo 5, es que modelar la herencia como una coerción explícita (idea que hemos tomado prestada de la Teoría de Tipos [100, 13]) es una decisión conveniente para la especificación algebraica de buena parte de la jerarquía de clases del sistema Kenzo. Aportamos también un estudio alternativo de dos posibles vías de análisis de la herencia: con dos géneros ocultos o con uno solamente. La primera opción es más natural, mientras que la segunda muestra muy claramente el papel dual de las semánticas inicial y final (o de la naturaleza algebraica y coalgebraica, si se prefiere) en nuestra aproximación (que es consistente con lo encontrado independientemente por otros autores; véase, por ejemplo, [84] o [54]).

En resumen, consideramos que los tres objetivos planteados en la introducción de la tesis han sido alcanzados. Esto es, hemos completado nuestra aproximación a la especificación de sistemas como EAT o Kenzo por medio de un diagrama de morfismos de instituciones. En segundo lugar, hemos analizado fragmentos de EAT que se basan en relaciones funtoriales entre sus estructuras. Por último, hemos sentado las bases para continuar nuestra modelización de los aspectos orientados a objetos de Kenzo y, en particular, de la herencia entre estructuras.

En un contexto más general, esta memoria puede entenderse como un paso más (parcial, pero no irrelevante) en el estudio de la aplicación de métodos formales a sistemas de software concretos y en funcionamiento. Así, este trabajo significa una contribución al debate sobre si dicha aplicación puede o no aproximarnos al final de la crisis del software, o, al menos, de si es realista continuar en esta línea.

Para terminar, señalamos que los problemas técnicos más importantes que quedan por resolver han sido recopilados en la última sección del capítulo precedente. A ellos dedicaremos, obviamente, parte de nuestros siguientes esfuerzos investigadores. Más en general, estamos interesados en aplicar nuestras técnicas y desarrollos a otros sistemas y entornos de Cálculo Simbólico. Citemos, por ejemplo, Aldor [1] o el sistema FOC [9], basado en el lenguaje de programación orientado a objetos OCaml [62]. En relación también con FOC, queremos orientar nuestra investigación hacia la búsqueda de una integración con los demostradores automáticos de teoremas (utilizados en nuestro grupo de investigación en [2], entre otros), que permita utilizar eficazmente nuestras técnicas de especificación algebraica para la corrección de fragmentos de Kenzo.

Bibliografía

[1] *The Aldor programming language.* `http://www.aldor.org`.

[2] J. Aransay, C. Ballarin y J. Rubio. Mechanising proofs in Homological Algebra. En Calculemus Autumn School 2002: Poster Abstracts, SEKI Report SR-02-06, pp. 13–18, 2002.

[3] E. Astesiano y M. Cerioli. Partial higher-order specifications. *Fundamenta Informaticae*, 16(2):101–126, 1992.

[4] E. Astesiano, H.-J. Kreowski y B. Krieg-Brückner, editores. *Algebraic Foundations of Systems Specification.* Springer, 1999.

[5] M. Barr. Terminal coalgebras in well-founded set theory. *Theoretical Computer Science*, 114(2):299–315, 1993.

[6] M. Barr y Ch. Wells. *Category Theory for Computer Science.* Prentice Hall International, 1995.

[7] F. Borceux. *Handbook of Categorical Algebra 1. Basic Category Theory.* Cambridge University Press, 1994.

[8] T. Borzyszkowski. Logical systems for structured specifications. *Theoretical Computer Science*, 286(2):197–245, 2002.

[9] S. Boulmé, T. Hardin, D. Hirschkoff, V. Ménissier-Morain y R. Rioboo. On the way to certify Computer Algebra Systems. En A. Armando y T. Jebelean, editores, *Federated Logic Conference - FLoC'99*, Electronic Notes in Theoretical Computer Science, vol. 23. Elsevier, 1999.

[10] V. Breazu-Tannen, T. Coquand, C. Gunter y A. Scedrov. Inheritance as explicit coercion. *Information and Computation*, 93:172–221, 1991.

[11] M. Broy. Equational specification of partial higher-order algebras. *Theoretical Computer Science*, 57:3–45, 1988.

[12] K. Bruce. The equivalence of two semantic definitions for inheritance in object-oriented languages. En S.D. Brookes, M.G. Main, A. Melton, M.W. Mislove y D.A. Schmidt, editores, *Mathematical Foundations of Programming Semantics*, Lecture Notes in Computer Science, vol. 598, pp. 102–124. Springer, 1992.

[13] K. Bruce y P. Wegner. An algebraic model of subtype and inheritance. En F. Bancilhon y P. Buneman, editores, *Advances in Database Programming Language*, pp. 75–96. ACM Press/Addison-Wesley, 1990.

[14] P. Burmeister. *A Model Theoretic Oriented Approach to Partial Algebras.* Akademie-Verlag, 1986.

[15] R.M. Burstall y R. Diaconescu. Hiding and behaviour: an institutional approach. En A. William Roscoe, editor, *A Classical Mind: Essays in Honour of C.A.R. Hoare*, pp. 75–92. Prentice-Hall, 1994.

[16] L. Cardelli. A semantics of multiple inheritance. *Information and Computation*, 76(2):138–164, 1988.

[17] L. Cardelli y P. Wegner. On understanding types, data abstraction, and polymorphism. *ACM Computing Surveys*, 17(4):471–522, 1985.

[18] G. Carlsson y R.J. Milgram. Stable homotopy and iterated loop spaces. En I.M. James, editor, *Handbook of Algebraic Topology*, pp. 505–583. North-Holland, 1995.

[19] M. Cerioli y J. Meseguer. May I borrow your logic? (Transporting logical structures along maps). *Theoretical Computer Science*, 173(2):311–347, 1997.

[20] M. Cerioli, T. Mossakowski y H. Reichel. From total equational to partial first-order logic. En E. Astesiano, H.-J. Kreowski y B. Krieg-Brückner, editores, *Algebraic Foundations of Systems Specification*, pp. 31–104. Springer, 1999.

[21] K. Ciesielski. *Set Theory for the Working Mathematician.* Cambridge University Press, 1997.

[22] C. Cîrstea. Coalgebra semantics for hidden algebra: parameterised objects and inheritance. En F. Parisi-Presicce, editor, *Recent Trends in Data Type Specification*, Lecture Notes in Computer Science, vol. 1376, pp. 174–189. Springer, 1998.

[23] C. Cîrstea. Semantic constructions for the specification of objects. *Theoretical Computer Science*, 260:3–25, 2001.

[24] Coq project. *The Coq Proof Assistant Reference Manual*, versión 7.3.1, 2002. `http://coq.inria.fr/`.

[25] F. Cornelius, M. Baldamus, H. Ehrig y F. Orejas. Abstract and behaviour module specifications. *Mathematical Structures in Computer Science*, 9(1):21–62, 1999.

[26] R. Diaconescu. Extra theory morphism for institutions: logical semantics. *Applied Categorical Structures*, 6(4):427–453, 1998.

[27] R. Diaconescu y K. Futatsugi. Behavioural coherence in object-oriented algebraic specification. *Journal of Universal Computer Science*, 6(1):74–96, 2000.

[28] C. Domínguez, L. Lambán, V. Pascual y J. Rubio. Hidden specification of a functional system. En R. Moreno-Díaz, B. Buchberger y J.L. Freire, editores, *Computer Aided Systems Theory - EUROCAST'2001*, Lecture Notes in Computer Science, vol. 2178, pp. 555–569. Springer, 2001.

[29] C. Domínguez, L. Lambán, V. Pascual y J. Rubio. Instituciones: Matemáticas para la especificación en computación. En L. Español y J.L. Varona, editores, *Margarita mathematica en memoria de José Javier (Chicho) Guadalupe Hernández*, pp. 221–233. Servicio de Publicaciones de la Universidad de La Rioja, 2001.

[30] C. Domínguez, L. Lambán, V. Pascual y J. Rubio. Modelling implementations as institution morphisms. En F. Orejas, F. Cuartero y D. Cazorla, editores, *Jornadas sobre Programación y Lenguajes - PROLE'2001*, pp. 229–239. Servicio de Publicaciones de la Universidad de Castilla La Mancha, 2001.

[31] C. Domínguez y J. Rubio. Dealing with inheritance in a symbolic computation system. En A. Montes, editor, *Encuentros de Álgebra Computacional y Aplicaciones - EACA'2000*, pp. 179–188. Universitat Politècnica de Catalunya, 2000.

[32] C. Domínguez y J. Rubio. Modeling inheritance as coercion in a symbolic computation system. En B. Mourrain, editor, *International Symposium on Symbolic and Algebraic Computation - ISSAC'2001*, pp. 107–115. ACM Press, 2001.

[33] X. Dousson. *Homologie effective des classifiants et calculs de groupes d'homotopie.* Thèse, Institut Fourier, 1999.

[34] X. Dousson, F. Sergeraert e Y. Siret. *The Kenzo program.* Institut Fourier, Grenoble, 1999. `http://www-fourier.ujf-grenoble.fr/~sergerar/Kenzo`.

[35] N. Doye. Automated coercion for AXIOM. En S. Dooley, editor, *International Symposium on Symbolic and Algebraic Computation - ISSAC'99*, pp. 229–235. ACM Press, 1999.

[36] H. Ehrig y B. Mahr. *Fundamentals of Algebraic Specification 1. Equations and Initial Semantics.* Springer, 1985.

[37] H. Geuvers, R. Pollack, F. Wiedijk y J. Zwanenburg. A constructive algebraic hierarchy in Coq. *Journal of Symbolic Computation*, 34:271–286, 2002.

[38] J.A. Goguen. Types as theories. En G.M. Reed, A.W. Roscoe y R.F. Wachter, editores, *Topology and Category Theory in Computer Science*, pp. 357–390. Oxford University Press, 1991.

[39] J.A. Goguen y R.M. Burstall. Institutions: Abstract model theory for specification and programming. *Journal of the Association for Computing Machinery*, 39(1):95–146, 1992.

[40] J.A. Goguen y R. Diaconescu. An Oxford survey of order sorted algebra. *Mathematical Structures in Computer Science*, 4(3):363–392, 1994.

[41] J.A. Goguen y R. Diaconescu. Towards an algebraic semantics for the object paradigm. En H. Ehrig y F. Orejas, editores, *Recent Trends in Data Type Specification*, Lecture Notes in Computer Science, vol. 785, pp. 1–29. Springer, 1994.

[42] J.A. Goguen y G. Malcolm. A hidden agenda. *Theoretical Computer Science*, 245(1):55–101, 2000.

[43] J.A. Goguen, G. Malcolm y T. Kemp. A hidden herbrand theorem: combining the object and logic paradigms. *Journal of Logic and Algebraic Programming*, 51:1–41, 2002.

[44] J.A. Goguen y J. Meseguer. Order-sorted algebra I: Equational deduction for multiple inheritance, overloading, exceptions and partial operations. *Theoretical Computer Science*, 105(2):217–273, 1992.

[45] J.A. Goguen y G. Roşu. Hiding more hidden algebra. En J.M. Wing, J. Woodcook y J. Davies, editores, *Formal Methods - FM'99*, Lecture Notes in Computer Science, vol. 1709, pp. 1704–1709. Springer, 1999.

[46] J.A. Goguen y G. Roşu. Institution morphisms. *Formal Aspects of Computing*, 13:274–307, 2002.

[47] R. Henniker y M. Bidoit. Observational logic. En A.M. Haeberer, editor, *Algebraic Methodology and Software Technology - AMAST'98*, Lecture Notes in Computer Science, vol. 1584, pp. 263–277. Springer, 1999.

[48] U. Hensel, M. Huisman, B. Jacobs y H. Tews. Reasoning about classes in object-oriented languages: logical models and tools. En Ch. Hankin, editor, *European Symposium on Programming - ESOP'98*, Lecture Notes in Computer Science, vol. 1381, pp. 105–121. Springer, 1998.

[49] C. Hoare. Proofs of correctness of data representations. *Acta Informatica*, 1:271–281, 1972.

[50] M Huisman y B. Jacobs. Inheritance in higher order logic: modeling and reasoning. En M. Aagaard y J. Harrison, editores, *Theorem Proving in Higher Order Logics - TPHOLs2000*, Lecture Notes in Computer Science, vol. 1869, pp. 301–319. Springer, 2000.

[51] B. Jacobs. Mongruences and cofree coalgebras. En V.S. Alagar y M. Nivat, editores, *Algebraic Methodology and Software Technology - AMAST'95*, Lecture Notes in Computer Science, vol. 936, pp. 245–260. Springer, 1995.

[52] B. Jacobs. Inheritance and cofree constructions. En P. Cointe, editor, *European Conference on Object-Oriented Programming - ECOOP'96*, Lecture Notes in Computer Science, vol. 1098, pp. 210–231. Springer, 1996.

[53] B. Jacobs. Objects and classes, co-algebraically. En B. Freitag, C.B. Jones, C. Lengauer y H.-J. Schek, editores, *Object-Orientation with Parallelism and Persistence*, pp. 83–103. Kluwer Academic Publishers, 1996.

[54] B. Jacobs y J. Rutten. A tutorial on (co)algebras and (co)induction. *Bulletin of the European Association for Theoretical Computer Science*, 62:222–259, 1997.

[55] S. Kamin y U. Reddy. Two semantic models of object-oriented languages. En C.A. Gunter y J.C. Mitchell, editores, *Theoretical Aspects of Object-Oriented Programming: Types, Semantics y Language Design*, pp. 463–495. MIT Press, 1994.

[56] L. Lambán, V. Pascual y J. Rubio. An object-oriented interpretation of the EAT system. *Aparecerá en Applicable Algebra in Engineering, Communication and Computation.*

[57] L. Lambán, V. Pascual y J. Rubio. Simplicial sets in the EAT system. En I. Bermejo, editora, *Encuentros de Álgebra Computacional y Aplicaciones - EACA'99*, pp. 267–276. Universidad de La Laguna, 1999.

[58] L. Lambán, V. Pascual y J. Rubio. Specifying implementations. En S. Dooley, editor, *International Symposium on Symbolic and Algebraic Computation - ISSAC'99*, pp. 245–251. ACM Press, 1999.

[59] L. Lambán y J. Rubio. Tipos Abstractos de Datos en Álgebra Homológica. En L. González-Vega, editor, *Encuentros de Álgebra Computacional y Aplicaciones - EACA'95*, pp. 113–122. Universidad de Cantabria, 1995.

[60] J. Lambek y P.J. Scott. *Introduction to higher order categorial logic.* Cambridge University Press, 1986.

[61] G.T. Leavens y D. Pigozzi. A complete algebraic characterization of behavioral subtyping. *Acta Informatica*, 36:617–663, 2000.

[62] X. Leroy. *The Objective Caml system, release 3.06.* 2002. `http://caml.inria.fr/ocaml/`.

[63] J. Loeckx, H.D. Ehrich y M. Wolf. *Specification of Abstract Data Types.* Wiley-Teubner, 1996.

[64] S. Mac Lane. *Homology.* Springer, 4th edition, 1994.

[65] J.P. May. *Simplicial Objects in Algebraic Topology.* Midway, 1982.

[66] B. Meyer. *Construcción de Software Orientado a Objetos.* Prentice Hall, segunda edición, 1999.

[67] R. Mitchell, J. Howse e I. Maung. As-a: a relationship to support code reuse. *Journal of Object-Oriented Programming*, 8(4):25–33, 1995.

[68] T. Mossakowski. *Representations, Hierarchies and Graphs of Institutions.* PhD thesis, University of Bremen, 1996.

[69] T. Mossakowski, A. Haxthausen y B. Krieg-Brückner. Subsorted partial higher-order logic as an extension of CASL. En D. Bert, C. Choppy y P. Mosses, editores, *Recent Trends in Data Type Specification*, Lecture Notes in Computer Science, vol. 1827, pp. 126–145. Springer, 2000.

[70] T. Mossakowski, A. Tarlecki y W. Pawlowski. Combining and representing logical systems. En E. Moggi y G. Rosolini, editores, *Category Theory and Computer Science - CTCS'97*, Lecture Notes in Computer Science, vol. 1290, pp. 177–196. Springer, 1997.

[71] F. Orejas, E. Pino y H. Ehrig. Institutions for logic programming. *Theoretical Computer Science*, 173(2):485–511, 1997.

[72] V. Pascual. *Objetos Localmente Efectivos y Tipos Abstractos de Datos.* Tesis doctoral, Universidad de La Rioja, 2002. ProQuest Information and Learning. Publicación electrónica ISBN 0-493-82725-0, 2003.

[73] Ch.C. Pinter. *Set Theory.* Addison-Wesley, 1971.

[74] H. Reichel. An approach to object semantics based on terminal coalgebras. *Mathematical Structures in Computer Science*, 5:129–152, 1995.

[75] H. Reichel. Specification semantics. En E. Astesiano, H.-J. Kreowski y B. Krieg-Brückner, editores, *Algebraic Foundations of Systems Specification*, pp. 131–158. Springer, 1999.

[76] J.C. Reynolds. *Theories of Programming Languages.* Cambridge University Press, 1998.

[77] D. Robinson. *A Course in the Theory of Groups.* Springer, second edition, 1996.

[78] G. Roşu. Equational axiomatizability for coalgebra. *Theoretical Computer Science*, 260:229–247, 2001.

[79] J. Rubio y F. Sergeraert. Locally effective objects and algebraic topology. En F. Eyssette y A. Galligo, editores, *Computational Algebraic Geometry*, Progress in Mathematics, vol. 109, pp. 235–251. Birkhäuser, 1993.

[80] J. Rubio y F. Sergeraert. Constructive algebraic topology. *Bulletin des Sciences Mathématiques*, 126:389–412, 2002.

[81] J. Rubio, F. Sergeraert e Y. Siret. *EAT: Symbolic Software for Effective Homology Computation.* Institut Fourier, Grenoble, 1997. `ftp://fourier.ujf-grenoble.fr/pub/EAT`.

[82] J. Rubio, F. Sergeraert e Y. Siret. Overview of EAT, a System for Effective Homology Computation. *The SAC Newsletter*, 3:69–79, 1998.

[83] J.J.M.M. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249(1):3–80, 2000.

[84] J.J.M.M. Rutten y D. Turi. Initial algebra and final coalgebra semantics for concurrency. En J.W. de Bakker, W.P. de Roever y G. Rozenberg, editores, *A Decade of Concurrency, Reflections and Perspectives*, Lecture Notes in Computer Science, vol. 803, pp. 530–582. Springer, 1994.

[85] A. Salibra y G. Scollo. A soft stairway to institutions. En M. Bidoit y C. Choppy, editores, *Recent Trends in Data Type Specification*, Lecture Notes in Computer Science, vol. 655, pp. 310–329. Springer, 1993.

[86] D. Sannella y A. Tarlecki. Essential concepts of algebraic specification and program development. *Formal Aspects of Computing*, 9:229–269, 1997.

[87] D. Sannella y A. Tarlecki. Algebraic preliminaries. En E. Astesiano, H.-J. Kreowski y B. Krieg-Brückner, editores, *Algebraic Foundations of Systems Specification*, pp. 13–30. Springer, 1999.

[88] F. Sergeraert. Homologie effective. *Comptes-Rendus l'Académie des Sciences*, 304:279–282 y 319–321, 1987.

[89] F. Sergeraert. Functional coding and effective homology. *Astérisque*, 192:57–67, 1990.

[90] F. Sergeraert. The computability problem in algebraic topology. *Advances in Mathematics*, 104:1–29, 1994.

[91] F. Sergeraert. Common lisp, typing and mathematics. Technical report, 2001. Conferencia impartida durante el taller *Implementation of symbolic computation systems in Common Lisp* previo al congreso *Encuentros de Álgebra Computacional y Aplicaciones - EACA'2000*. Disponible por red en: `http://www-fourier.ujf-grenoble.fr/~sergerar/Papers/`.

[92] A. Snyder. Inheritance and the development of encapsulated software components. En B. Shriver y P. Wegner, editores, *Research Directions in Object-Oriented Programming*, pp. 165–188. MIT Press, 1987.

[93] G. Steele. *Common Lisp. The language.* Digital Press, second edition, 1990.

[94] A. Taivalsaari. On the notion of inheritance. *ACM Computing Surveys*, 28(3):438–479, 1996.

[95] A. Tarlecki. Moving between logical systems. En M. Haveraaen, O. Owe y O.J. Dahl, editores, *Recent Trends in Data Type Specification*, Lecture Notes in Computer Science, vol. 1130, pp. 478–502. Springer, 1996.

[96] A. Tarlecki. Institutions: An abstract framework for formal specifications. En E. Astesiano, H.-J. Kreowski y B. Krieg-Brückner, editores, *Algebraic Foundations of Systems Specification*, pp. 31–104. Springer, 1999.

[97] A. Tarlecki, R.M. Burstall y J.A. Goguen. Some fundamental algebraic tools for the semantics of computation: Part 3. Indexed categories. *Theoretical Computer Science*, 91(2):239–264, 1991.

[98] The CoFI Task Group on Language Design. CASL, The Common Algebraic Specification Language - Summary. Version 1.0.1 Technical report, 2001. `http://www.brics.dk/Projects/CoFI/Documents/CASL/Summary/`.

[99] M. Wand. Final algebra semantics and data type extensions. *Journal of Computer and System Sciences*, 19:27–44, 1979.

[100] P. Wegner. The object-oriented classification paradigm. En B. Shriver y P. Wegner, editores, *Research Directions in Object-Oriented Programming*, pp. 479–560. MIT Press, 1987.

[101] M. Wirsing. Algebraic specification. En J.V. Leeuwen, editor, *Formal Models and Semantics*, pp. 677–788. Elsevier, 1992.