Existencia de coálgebras finales para funtores polinomiales

Andrés Felipe Téllez Crespo

Universidad del Valle
Facultad de Ciencias Naturales y Exactas
Programa Académico de Matemáticas
Santiago de Cali

2011

Existencia de coálgebras finales para funtores polinomiales

Andrés Felipe Téllez Crespo

andrestellez84@hotmail.com

Trabajo de grado presentado como requisito parcial para optar por el título de Matemático.

Director:
Guillermo Ortiz Rico

Matemático, Ph.D.

Universidad del Valle

Facultad de Ciencias Naturales y Exactas

Programa Académico de Matemáticas

Santiago de Cali

2011

Universidad del Valle
Facultad de Ciencias Naturales y Exactas
Programa Académico de Matemáticas

Andrés Felipe Téllez Crespo, 1984

Existencia de coálgebras finales para funtores polinomiales

Palabras clave: Coálgebras Finales, Funtores Polinomiales, Bisimulaciones, Completez, Autómatas

Nota de aprobación:

El trabajo de grado titulado **Existencia de coálgebras finales para funtores polinomiales** presentado por el estudiante **Andrés Felipe Téllez Crespo**, para optar por el título de matemático, fue revisado por el jurado y calificado como:.

APROBADO

Director:
Guillermo Ortiz Rico

Jurado
Raúl Gutiérrez de Piñerez Reyes

Agradecimientos

Agradezco principalmente a Guillermo Ortiz, por su guía y paciencia en los diversos cursos y en la elaboración del trabajo de grado, además por iniciarme en un tema que ha despertado mi interés y del cual espero aprender más. A mis padres por el apoyo en todos estos años de universidad y especialmente por la comprensión de mis decisiones. A Luisa Fernanda Higueras por su acompañamiento constante, por su motivación y por las infinitas charlas constructivas. A mi hermana Luz Elena por el interés que ha mostrado en mi formación profesional y por el apoyo en mis estudios. A Alberto León Giraldo por ser un excelente amigo y con sus buenos consejos ayudarme a concentrarme en momentos que lo necesitaba. Y por último gracias al sol que sin su ayuda diaria esto hubiera sido imposible.

Índice

Resumen

El trabajo es una reconstrucción de un resultado que dice que todo funtor polinomial Kripke tiene asociada una coálgebra final, este resultado fue probado por primera vez en 1989 por P. Aczel y N. Mendler en la revista Category Theory and Computer Science número 389 en las páginas 357-365. La reconstrucción se basó en varios textos pero principalmente en el trabajo de Bart Jacobs del libro Introduction to Coalgebra. Towards Mathematics of States and Observations. Se usa éste por su facilidad de asociar los diferentes resultados a ejemplos básicos de las ciencias de la computación. En el desarrollo del trabajo se usa lenguaje de categorías, se hace una introducción sencilla a los conceptos propios de las coálgebras como Bisimulación, Invariantes, funtor adjunto y un poco de lógica temporal junto con algunos ejemplos en ciencias de la computación para visualizar mejor los conceptos, esto prepara el terreno para la prueba del resultado de existencia de coálgebra final. Luego se prueban resultados importantes que son consecuencia de la existencia de una coálgebra final. Estos resultados incluyen: la construcción de coálgebras colibres, la completez de la categoría de las coálgebras que para estos funtores es bicompleta y por último se demuestra el Principio de Prueba Coinductiva.

Introducción

El objetivo general de este documento es presentar una introducción a la teoría de coálgebras, incluyendo algunos ejemplos en ciencias de computación que permitan apreciar una aplicación inmediata del álgebra en el ámbito de la ingeniería. La teoría de coálgebras se adapta muy bien capturando de manera abstracta el concepto de máquina, como ejemplo genérico de máquina se verán los Autómatas de los cuales existen casos específicos como las Máquina de Mealy, Máquina de Turing, entre otros.

La formulación y la presentación del tema se hará en el contexto de la teoría de categorías, intentando mantener la generalidad que este enfoque nos permite, además, introduciendo las coálgebras de esta manera podemos tener una gran conexión con las álgebras, que también es una herramienta usada en ciencias de computación. Las coálgebras nos permiten un acercamiento desde un punto de vista de comportamiento y observaciones, enfoque que se aprecia por ejemplo en Programación Orientada a Objetos, donde no hay una diferencia entre datos y métodos, en cambio se hace un enfoque en los objetos y que estos tengan su método asociado. De manera dual, desde el punto de vista algebraico se captura el concepto de constructor, es decir, en las álgebras se pueden construir nuevos objetos, en las coálgebras se ven comportamientos y se pueden modificar, pero no crear nuevos estados.

Siguiendo en la línea de los conceptos duales se tiene que las bisimulaciones son a las coálgebras como las congruencias son a las álgebras. Una bisimulación es una relación que se adapta al dinamismo que proporcionan las coálgebras, capturando los estados que en sucesivas interacciones tienen el mismo comportamiento, por lo tanto dos máquinas son observacionalmente iguales si sus estados están relacionados por medio de una bisimulación. Este concepto ha tenido un uso extensivo en varias áreas de la computación como lo son Lenguajes Funcionales, Lenguajes Orientados a Objetos, Tipos de Datos, Dominios, Bases de Datos, Optimización de Compiladores, Análisis de Programas y Herramientas de Verificación.

La teoría de coálgebras como la conocemos hoy se formó con aportes de varias áreas del conocimiento, entre ellas se encuentran: conjuntos no bien fundados, teoría de sistemas y lógica modal. El lenguaje con el que se estudian las coálgebras es la teoría de categorías, este fue desarrollado por Samuel Eilenberg y Saunders Mac Lane entre 1942-1945 como parte de su trabajo en topología, especialmente en topología algebraica.

En 1969 Kalman publica un artículo sobre sistemas dinámicos relacionados con teoría de autómatas. Durante los 70's Arbib, Manes y Goguen analizaron el trabajo de Kalman y se dieron cuenta que podían reformular y probar los resultados de manera más general usando construcciones categóricas. Y así lo hicieron en "Machines in a category" en 1980.

Por otro lado, Reichel en 1981 en "Behavioral equivalence" fue el primero en usar una división de tipos de datos entre visibles y ocultos. Los últimos capturaban estados que no son accesibles directamente. La igualdad fue usada sólo para elementos visibles. Para los elementos ocultos se usó igualdad de comportamiento. Esto significa que eran iguales mientras pudieran ser observados.

Otro aporte lo hace Aczel en 1988 en su publicación "Non-well-founded sets". Debido al uso de la terminología algebraica desde el principio, su teoría de conjuntos permitía infinitas cadenas descendentes. El desarrollo de su teoría fue motivada por el deseo de dar sentido a la teoría de Milner de procesos concurrentes con comportamiento potencialmente infinito. Otra contribución importante de Aczel fue mostrar cómo tratar la bisimulación en un contexto coalgebraico, especialmente por establecer el primer vínculo entre pruebas por bisimulación y la finalidad de coálgebras en su publicación "A final coálgebra theorem" de 1989.

Un desarrollo más reciente es la conexión entre coálgebras y lógica modal. En general tales lógicas intentan dar condiciones de verdad de las proposiciones sobre los conocimientos, creencias y tiempo. En ciencias de la computación tales lógicas son usadas para razonar acerca de cómo los programas se comportan, y expresar propiedades dinámicas de transiciones entre los estados. La lógica temporal es una parte de lógica modal la cual esta especialmente adecuada para razonar acerca de sistemas de estados, a través de sus operadores nexttime y lasttime. Como las coálgebras dan formalizaciones abstractas de tales sistemas de estados se espera una conexión. Fue Moss en 1999 en "Coalgebraic logic" quién primero asoció una lógica modal para coálgebras.

La parte I del documento se desarrollará haciendo hincapié en los conceptos básicos por medio de ejemplos que permitan tener cierta familiaridad con el comportamiento y las coálgebras finales, temas que se volverán más profundos a medida que se avance.

En la parte II se introducirá el concepto de bisimulación que como ya se dijo es fundamental para capturar la equivalencia de dos coálgebras. Se mostrará dos definiciones de bisimulación y se probará que son equivalentes cuando se habla especificamente de funtores polinomiales, también se harán acercamientos a los invariantes y los operadores temporales debido a la aplicación que tienen para la prueba del resultado principal que se hará en la siguiente parte del documento.

En la parte III se empezará con los resultados técnicos que se pasaron por alto en las dos primeras partes, mostrando propiedades de las bisimulaciones, el concepto de coálgebra observable, propiedades estructurales de las coálgebras y por último el resultado principal que trata de la existencia de coálgebras finales, lugar donde se ven reflejados los comportamientos.

En la parte IV del documento se probarán resultados profundos acerca de la estructura de las coálgebras que son consecuencia de la existencia de una coálgebra final, como que es una categoría bicompleta, es decir, existen todos sus límites y colímites. Por último, se incluirá un resultado importante que caracteriza el comportamiento de dos estados de una máquina, además, refleja la importancia de casi todos los conceptos utilizados, como la coálgebra final, bisimilaridad, comportamiento, entre otros.

Parte I
Introducción a las coálgebras

En este capítulo se mostrará de manera general la relación que hay entre algunos sistemas que se pueden considerar máquinas (como Autómatas y sistemas de transición) y su representación abstracta dada por una coálgebra. Se introducirá el concepto de coálgebra final, y se presentará la clase de funtores polinomiales, por último, se probarán algunos resultados relacionados con las coálgebras finales.

1. Acercamiento a las coálgebras

Se usará la idea intuitiva de una máquina para notar que el lenguaje categórico resulta ser el adecuado para la definición de una representación abstracta de ésta. Consideremos primero una máquina con un conjunto de estados S, y una aplicación que permite las transiciones entre estados llamada c. Ahora, supongamos que luego de hacer la acción c, la máquina tiene todas o alguna de las siguientes opciones: mostrar un comportamiento (que podría ser un sonido, una letra impresa, etc.) o brindar la posibilidad de una modificación (como el teclado de un computador). Estas posibilidades de observación y modificación es lo que en computación se conoce como entrada y salida o por sus siglas en inglés I/O. Para admitir ese tipo de interacción, lo adecuado sería cambiar el espacio de estados de llegada luego de aplicar c. Este espacio con nuevas posibilidades de interacción lo vamos a llamar $F(S)$. Veamos esto con un ejemplo.

1.1 Ejemplo
Supongamos que la máquina anterior tenga como espacio de estados los números reales pertenecientes al intervalo $[0, 1)$, y se necesita que muestre su representación decimal paso a paso, además de considerar si su representación es finita o infinita. De esta manera las posibles observaciones en cada interacción pertenecen al conjunto $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Sea r el número real en cuestión, así la coálgebra actúa de la siguiente manera:

$$c(r) = \begin{cases} \perp & si\ r = 0 \\ (d, 10r - d) & en\ otro\ caso,\ donde\ d \in A\ es\ tal\ que\ d \leq 10r < d + 1 \end{cases}$$

La manera como se expresa su resultado (d, r_2) quiere decir que el espacio después de aplicar la transición es el producto cartesiano $A \times [0, 1)$. Ya que cláramente $10r - d$ es un número que pertenece a $[0, 1)$.

El símbolo $\perp$ significa que la sucesión es finita y por lo tanto debe parar, como este resultado no pertenece ni a A ni a $[0, 1)$ se puede organizar como un elemento independiente de los anteriores, esto se puede hacer mediante el coproducto, finalmente el espacio de llegada es: $F([0, 1)) = A \times [0, 1) \cup \{\perp\}$.

Si aplicamos el proceso a $1/4$ resulta:
$c(\frac{1}{4}) = (2, \frac{1}{2})$ ya que $2 \leq \frac{5}{2} < 3$
si repetimos el proceso con $1/2$ tenemos:
$c(\frac{1}{2}) = (5, 0)$ ya que $5 \leq 5 < 6$
$c(0) = \perp$

De esta manera las repetidas aplicaciones de c a un determinado número en $[0, 1)$ muestra su representación decimal paso a paso, en este caso muestra 2, 5 y luego detiene el cómputo, representando a $\frac{1}{4} = 0{,}25$. El siguiente es un ejemplo de un número con representación infinita:

$c(\frac{1}{3}) = (3, \frac{1}{3})$ ya que $3 \leq \frac{10}{3} < 4$

donde se ve claramente que el proceso nunca finaliza representando el número $\frac{1}{3}$.

El ejemplo anterior (tomado de [2]) muestra la versatilidad de un cambio de espacio de estados, por un lado se pueden hacer observaciones de lo que está ocurriendo y por otro lado se tiene la posibilidad de frenar el proceso si las condiciones lo requieren. También está la opción de representar un programa que corra por siempre, como por ejemplo un sistema operativo. En el ejemplo 1.2 se verá claramente cómo organizar estos resultamos como una sucesión, además garantizar que la representación sea única.

El lenguaje de categorías resulta ser una herramienta natural en el desarrollo de la teoría de coálgebras, es decir, de estas aplicaciones o acciones (en el ejemplo anterior denotado por c) y el conjunto de estados S.

De esta manera, si se tienen dos coálgebras; $c : X \to F(X)$, y $d : Y \to F(Y)$, las podemos relacionar mediante otra aplicación $f : X \to Y$ llamada morfismo, que relaciona los elementos de X y Y, y otra $F(f)$ que relaciona los elementos $F(X)$ y $F(Y)$.

La relación con teoría de categorías se hace evidente cuando organizamos todas estas aplicaciones en un sólo diagrama conmutativo:

$$\begin{tikzcd} F(X) \arrow[r, "F(f)"] & F(Y) \\ X \arrow[u, "c"] \arrow[r, "f"'] & Y \arrow[u, "d"'] \end{tikzcd}$$

Además nos brinda la posibilidad de realizar aproximaciones más generales, y no restringirnos sólo a la categoría de conjuntos como en el ejemplo, el lenguaje de categorías también es útil cuando se usan los conceptos como transformaciones naturales y adjunciones de funtores, los cuales verán su aplicación un poco más adelante.

Por el momento, formalicemos algunas de estas ideas. Asumiremos que el lector tiene alguna familiaridad con los conceptos básicos de teoría de categorías (categoría, funtor, objeto final, entre otros (Ver definiciones en [1] y [2])), por lo tanto daremos la definición formal del concepto de coálgebras.

1.2 Definición: (Coálgebra)

Sea $\mathbb{C}$ una categoría arbitraria, con un endofuntor $F : \mathbb{C} \to \mathbb{C}$.

- Una F-coálgebra o simplemente una **coálgebra** cuando se sobreentiende F, consiste de un objeto $X \in \mathbb{C}$ junto con un morfismo $c : X \to F(X)$.

- Dadas dos coálgebras $c : X \to T(X)$ y $d : Y \to T(Y)$, un **homomorfismo de coálgebras** es un morfismo $f : X \to Y$ en $\mathbb{C}$ que conmuta con las estructuras, como se ve en el siguiente diagrama:

$$\begin{tikzcd} T(X) \arrow[r, "T(f)"] & T(Y) \\ X \arrow[u, "c"] \arrow[r, "f"'] & Y \arrow[u, "d"'] \end{tikzcd}$$

- Las coálgebras con los homomorfismos entre ellas forman una categoría, la denotaremos como $CoAlg(F)$.

En terminología computacional el morfismo c hace las veces de función de transición de un estado a otro, y X constituye el espacio de estados.

2. Ejemplo de coálgebra final

El desarrollo principal de este documento será sobre la existencia de objetos finales en la categoría $CoAlg(F)$, como ésta depende de F, también la existencia de objetos finales. De esta manera, cabe la posibilidad que en algunas de estas categorías exista el objeto final, mientras que en otras categorías no exista. Se probará que para una clase importante de funtores llamados polinomiales sí existen las coálgebras finales. También se mostrará que en general no tienen que existir para otro tipo de funtores.

El estudio de estos objetos finales no es un capricho o algo sin relevancia, es más, una coálgebra final muestra directamente cómo se comporta cualquier coálgebra de la misma signatura. Por definición existe un morfismo único a este objeto final, a este morfismo lo llamaremos "comportamiento". A continuación exhibiremos un ejemplo de una coálgebra final donde se puede apreciar lo anterior.

2.1 Ejemplo
Primero algo de notación, sea A^* el conjunto de las sucesiones finitas de elementos de A, y $A^{\mathbb{N}}$ el conjunto de las sucesiones infinitas de elementos de A, con base en estos dos conjuntos podemos definir el conjunto $A^\infty = A^* \cup A^{\mathbb{N}}$. A este conjunto le asignaremos una estructura de transición que llamaremos **next** definida de la siguiente manera:

$$A^\infty \xrightarrow{\quad next \quad} \{\perp\} \cup (A \times A^\infty)$$

$$\sigma \mapsto \begin{cases} \perp & si\ \sigma\ es\ la\ sucesi\acute{o}n\ vac\acute{\i}a\ \langle\rangle \\ (a, \sigma') & si\ \sigma = a.\sigma'\ con\ cabeza\ a \in A\ y\ cola\ \sigma' \in A^\infty \end{cases}$$

Por lo tanto, el estado siguiente de una sucesión es otra sucesión excepto que se le remueve su primer elemento, además que permite ver el objeto que se ha retirado, aplicando $next$ varias veces podremos observar los elementos de la sucesión. La aplicación **next** de hecho es un isomorfismo, ya que $next^{-1}$ envía $\perp$ a la sucesión vacía y un par $(a, \tau) \in A \times A^\infty$ a la sucesión $a.\tau$. Esto no es exclusivo de esta coálgebra, más adelante se probará que si una coálgebra es final, necesariamente es un isomorfismo. Ahora vamos a probar que $next : A^\infty \to \{\perp\} \cup A \times A^\infty$ es final para las álgebras de su tipo, es decir, que para cada coálgebra arbitraria $c : S \to \{\perp\} \cup A \times S$ sobre un conjunto S hay una única función "comportamiento" $beh_c : S \to A^\infty$ que es un homomorfismo de coálgebras. Esto quiere decir que para cada $x \in S$ se cumple:

- si $c(x) = \perp$ entonces $next(beh_c(x)) = \perp$
- si $c(x) = (a, x')$, entonces $next(beh_c(x)) = (a, beh_c(x'))$

Simplificando estos dos puntos en un diagrama conmutativo:

$$\begin{tikzcd}
\{\perp\} \cup A \times S \arrow[r, "\{\perp\} \cup A \times beh_c"'] & \{\perp\} \cup A \times A^{\infty} \\
S \arrow[u, "c"] \arrow[r, "beh_c"'] & A^{\infty} \arrow[u, "next"', "\cong"]
\end{tikzcd}$$

Ya que estamos tratando con procesos que por lo general tienen varios pasos, convendría obtener una versión iterada de beh_c y de c, explícitamente beh_c tiene la siguiente forma:

$$beh_c(x) = \begin{cases} \langle\rangle & si\ c(x) = \perp \\ \langle a \rangle & si\ c(x) = (a, x')\ y\ c(x') = \perp \\ \langle a, a' \rangle & si\ c(x) = (a, x')\ y\ c(x') = (a, x'')\ y\ c(x'') = \perp \\ \vdots & \end{cases}$$

para hallar una expresión para todo $n \in \mathbb{N}$ se necesita una versión iterada $c^n : S \to \{\perp\} \cup A \times S$ definida de la siguiente manera:

$$c^0(x) = c(x)$$

$$c^{n+1}(x) = \begin{cases} \perp & si\ c^n(x) = \perp \\ c(y) & si\ c^n(x) = (a, y) \end{cases}$$

De esta manera se puede redefinir beh_c actualizada a su versión iterada.

$$beh_c(x) = \begin{cases} \langle a_0, a_1, a_2, \ldots \rangle & si\ \forall n \in \mathbb{N}.c^n(x) \neq \perp,\ y\ c^i(x) = (a_i, x_i) \\ \langle a_{0,} \ldots, a_{m-1} \rangle & si\ m \in \mathbb{N}\ es\ el\ menor\ número\ con\ c^m(x) = \perp\ y\ c^i(x) = (a_i x_i)\ para\ i < m \end{cases}$$

ahora sí podemos mostrar la conmutatividad:

- Si $c(x) = \perp$ entonces el último m con $c^m(x) = \perp$ es 0, entonces $beh_c(x) = \langle\rangle$, y también $next(beh_c(x)) = \perp$.
- Si $c(x) = (a, x')$,entonces distinguimos dos casos:
  - Si $\forall n \in \mathbb{N}.c^n(x) \neq \perp$ entonces $\forall n \in \mathbb{N}.c^n(x') \neq \perp$ y $c^{i+1}(x) = c^i(x') = (a_i, x_i)$, entonces

$$\begin{aligned} next(beh_c(x)) &= next(\langle a, a_0, a_1, \ldots \rangle) \\ &= (a, \langle a_0, a_1, \ldots \rangle) \\ &= (a, beh_c(x')) \end{aligned}$$

  - Si m es el menor con $c^m(x) = \perp$, entonces $m > 0$ y $m - 1$ es el menor k con $c^k(x') = \perp$. Para $i < m - 1$ tenemos $c^{i+1}(x) = c^i(x') = (a_i, x_i)$, así:

$$\begin{aligned} next(beh_c(x)) &= next(\langle a, a_0, a_1, \ldots, a_{m-2} \rangle) \\ &= (a, \langle a_0, a_1, \ldots, a_{m-2} \rangle) \\ &= (a, beh_c(x')) \end{aligned}$$

La unicidad de beh_c se puede probar fácilmente si se supone que hay otro homomorfismo que cumple las mismas condiciones que beh_c y luego se razona por inducción para demostrar que realmente son iguales estos dos homomorfismos.

Un caso particular de la coálgebra $c : S \to \{\perp\} \cup A \times S$, es el Ejemplo 1.1. En este ejemplo el homomorfismo beh_c me está diciendo qué representación decimal tiene un número entre $[0, 1)$.
En este punto se evidencia claramente la importancia que un funtor tenga una coálgebra final.

Por un lado está la *existencia* del homomorfismo beh_c y me dice cómo puedo ver representado el comportamiento de un estado de la máquina en un conjunto abstracto que es el conjunto más grande que puede representar ese comportamiento, esto se llama **principio de definición coinductiva** y es el dual de lo que en las álgebras es la definición inductiva. Un ejemplo de esto en el campo de las álgebras lo dan los números naturales que son un álgebra inicial del funtor $F(X) = 1 + X$, es decir, $1 + \mathbb{N} \underset{\to}{\cong} \mathbb{N}$ es un isomorfismo y de hecho, $\mathbb{N}$ es el menor conjunto que cumple la definición inductiva (salvo isomorfismos), esto es, que tenga un elemento inicial y los otros que pertenecen a él se obtengan por medio de la operación sucesión. Tanto las álgebras iniciales como las coálgebras finales se pueden describir de una manera canónica: las álgebras iniciales están conformadas por los elementos que se pueden obtener por una aplicación iterada de las operaciones, y las coálgebras finales están conformadas sólo por observaciones que muestran las máquinas.

Por otro lado está la *unicidad* del homomorfismo que sirve para demostrar que dos morfismos son iguales. Como ejemplo pensemos en la coálgebra $p : A^\infty \to \{\perp\} \cup A \times A^\infty$ que me muestra los términos ubicados en las posiciones pares de una sucesión y otra coálgebra $i : A^\infty \to \{\perp\} \cup A \times A^\infty$ que haga lo mismo con los términos impares. Ambas tienen una representación en la coálgebra final de las sucesiones presentada en el ejemplo 2.1 por medio de dos homomorfismos comportamientos $par : A^\infty \to A^\infty$ e $impar : A^\infty \to A^\infty$ respectivamente. Estos homomorfismos son muy parecidos, hacen lo mismo pero trasladados una posición, llamemos al homomorfismo que hace la acción de trasladar $tras : A^\infty \to A^\infty$. De esta manera podemos notar que $impar = par \circ tras$, mostrando que tanto $impar$ como $par \circ tras$ son ambos homomorfismos $i \to next$, como $next$ es final, por unicidad se tiene que $impar = par \circ tras$.

3. Funtores Polinomiales

El objetivo principal es demostrar la existencia de coálgebras finales para una gama más amplia de funtores llamados funtores polinomiales, una idea intuitiva de éstos es que son de la forma
$F(X) = A + (B \times X)^C$. Los ejemplos que acabamos de ver representan al funtor $F(X) = 1 + A \times X$.

3.1 Definición: (Funtores polinomiales)

- La colección **SPF** de funtores polinomiales simples es la menor clase de funtores $Sets \to Sets$ que satisfacen las siguientes clausulas:

  1. El funtor identidad $Sets \to Sets$ está en SPF.
  2. Para cada conjunto A, el funtor constante $A : Sets \to Sets$ está en SPF. Recodar que a cada conjunto X le asocia el conjunto A, y a cada función f la identidad id_A sobre A.
  3. Si F y G están en SPF, entonces también su producto $F \times G$, definido como
     $X \mapsto F(X) \times G(X)$. Sobre funciones se define como $f \mapsto F(f) \times G(f)$.

4. Si F y G están en SPF, entonces también su coproducto $F+G$, definido como $X \mapsto F(X)+G(X)$. Sobre funciones se define como $f \mapsto F(f)+G(f)$.
5. Para cada conjunto A, si F es está en SPF, entonces también su exponente constante F^A definido como $X \mapsto F(X)^A$. Y envía la función $f : X \to Y$ a la función $F(f)^A = F(f)^{id_A}$ la cual envía $h : A \to F(X)$ a $F(f) \circ h : A \to F(Y)$.

- La clase **KPF** de funtores polinomiales (Kripke) es la clase de SPF definido por las clausulas anteriores, con SPF reemplazado por KPF, más dos reglas adicionales:

1. Si F está en KPF, entonces también el conjunto potencia $\mathcal{P}(F)$, definido como $X \mapsto \mathcal{P}(F(X))$ sobre conjuntos, y como $f \mapsto \mathcal{P}(F(f))$ sobre funciones.
2. Si F está en KPF, entonces también F^* definido como $X \mapsto F(X)^*$ de sucesiones finitas de elementos de $F(X)$. Sobre una función f se define como $F(f)^*$.

Ocasionalmente diremos que un funtor F es un KPF finito. Esto quiere decir que el funtor potencia es finito $\mathcal{P}_{fin}(F)$.

Todas estas construcciones funtoriales se pueden consultar en [2], como ejemplo se mostrará la construcción del exponente.

Dados dos conjuntos X y Y se puede considerar el conjunto $Y^X = \{f \mid f\ es\ una\ función\ X \to Y\}$ este conjunto además viene equipado con algunas operaciones básicas. Está la función evaluación $ev : Y^X \times X \to Y$ la cual envía el par (f, x) a la aplicación de la función $f(x)$. Y para una función $f : Z \times X \to Y$ está la función abstracción $\Lambda(f) : Z \to Y^X$ la cual envía $z \in Z$ a la función $x \mapsto f(z, x)$. La esencia de esta construcción se puede sintetizar en forma de una correspondencia biyectiva, llamada **Currificación** en referencia al lógico Haskell Curry, y usada comúnmente en ciencias de la computación para definir de manera más fácil algunas funciones, pero aquí se usará para definir el funtor exponente y luego para ver los autómatas como coálgebras.

$$Z \times X \to Y \Longleftrightarrow Z \to Y^X$$

Es simplemente una forma de evaluación, es decir, si una función tiene dos variables puedo evaluar la primera y luego la segunda.

Para dos aplicaciones $k : X \to U$ en $Sets^{op}$ y $h : Y \to V$ en $Sets$ se necesita definir una función $h^k : Y^X \to V^U$ entre exponentes. El hecho que $k : X \to U$ es un morfismos en $Sets^{op}$ quiere decir que realmente es una función $k : U \to X$ en $Sets$, de esta manera se puede definir h^k sobre una función $f \in Y^X$ como:

$$h^k(f) = h \circ f \circ k$$

La elección de $k : X \to U$ en $Sets^{op}$ se vuelve clara si miramos el siguiente diagrama, ya que lo que necesitamos es una función en V^U:

$$\begin{tikzcd} U \arrow[r, "h^k(f)"] & V \\ X \arrow[u, "k"] \arrow[r, "f"'] & Y \arrow[u, "h"'] \end{tikzcd}$$

La definición en lenguaje categórico quedaría de la siguiente manera.

Sea $\mathbb{C}$ una categoría con productos $\times$. El exponente de dos objetos $X, Y \in \mathbb{C}$ es un nuevo objeto $Y^X \in \mathbb{C}$ con un morfismo evaluación tal que para cada aplicación $f : Z \times X \to Y$ en $\mathbb{C}$ existe un único morfismo abstracción $\Lambda(f) : Z \to Y^X$ en $\mathbb{C}$ que hace conmutar el siguiente diagrama:

$$\begin{tikzcd} Y^X \times X \arrow[r, "ev"] & Y \\ Z \times X \arrow[u, "\Lambda(f)\times id_X"] \arrow[ur, "f"'] & \end{tikzcd}$$

Observación:
Nos restringimos a esta clase de funtores polinomiales principalmente por las siguientes razones:

1. Son concretos y fáciles de manipular
2. Las coálgebras asociadas a estos funtores polinomiales incluyen muchos ejemplos interesantes.
3. Como se definieron por estructura, resultan muy útiles en algunas definiciones (levantamiento de relaciones y predicados) y en algunas pruebas.

Entre los ejemplos de coálgebras asociadas a algunos funtores polinomiales están:

- Árboles binarios: $F(X) = A \times X \times X$
- Árboles en general: $F(X) = A \times X^*$
- Árboles binarios con opción de parar y dejar una marca de un elemento de B:
  $F(X) = B + A \times X \times X$

Un ejemplo importante son los **autómatas** que se definen como una 5-tupla (S, A, s_0, δ, F) donde:

S es un conjunto de estados
A es un alfabeto
s_0 es un estado inicial
δ es una función de transición
F es el conjunto de estados finales

El autómata se llama **determinista** si existe un único sucesor para cada estado y **no-determinista** si sucede lo contrario.

La idea del autómata determinista es que se escoge una letra del alfabeto y dependiendo de la letra se dirige a un estado o a otro, por lo tanto el proceso que se está haciendo se puede representar por la coálgebra $S \times A \to S$, pero por medio de Currying sabemos que podemos expresar ésto como: $S \to S^A$. Además si queremos que se muestre un comportamiento con salidas en B lo podemos hacer como en las sucesiones y transformar esta coálgebra en: $\langle \delta, \epsilon \rangle : S \to S^A \times B$, donde δ es la función de transición y ϵ es la función de observación, aquí como ejemplo de autómata determinista se muestra una Máquina de Mealy (ver Fig 1), las entradas o alfabeto son 0 ó 1 (en rojo) y las salidas o estados observables son también 0 ó 1 (en azul), esta máquina en particular calcula el "o" exclusivo (también conocido como XOR) de las últimas dos entradas. Otros ejemplos muy conocidos son las Máquinas de Turing, y las Máquinas de Moore.

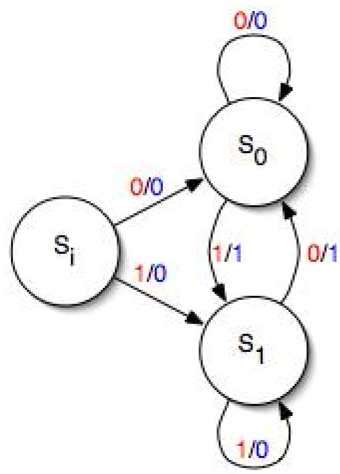

Fig. 1 XOR

La coálgebra asociada al autómata no-determinista es similar, excepto que como a cada estado le puede corresponder un subconjunto de estados de S, se usa el funtor potencia: $\langle \delta, \epsilon \rangle : S \to \mathcal{P}(S)^A \times B$.

De este autómata se desprenden varios casos interesantes, entre ellos:

- Sistemas de transición sin marcas (UTS) $\{A = 1, B = 1\}$ lo que corresponde a *relaciones* $\subseteq S \times S$. Son útiles para chequear modelos.
- Sistemas de transición con marcas (LTS) $\{B = 1\}$ lo que corresponde a *relaciones* $\subseteq S \times A \times S$. Se usan en teoría de procesos.
- Estructura Kripke: $S \to \mathcal{P}(S) \times \mathcal{P}(AtProp)$ $\{A = 1, B = \mathcal{P}(AtProp)\}$ donde $AtProp$ es el conjunto de proposiciones atómicas, este se usa en semántica de lógica modal.

4. Coálgebras Finales

Las coálgebras finales gozan de algunas propiedades que no tienen las coálgebras no-finales, algunas de ellas ya las vimos en los Ejemplos 1.1 y 2.1, pero en esta sección serán formalizadas. Los funtores que admiten coálgebra final son de especial interés, por ejemplo, en el caso de coálgebras basadas en estados, las finales dan una idea intuitiva de la noción de comportamiento. Y de hecho muchos ejemplos interesantes de objetos matemáticos bien conocidos pueden ser asociados naturalmente con la coálgebra final de algún funtor: sucesiones, lenguajes aceptados por autómatas, en lógica modal tiene aplicaciones a Marcos Generales Descriptivos (DGF) usando Funtores llamados de Vietoris, para garantizar la existencia de la coálgebra final notan la dualidad entre Álgebra Modal (MA) y DGF, y teniendo en cuenta que MA tiene objeto inicial, DGF debe tener final un análisis más detallado se puede ver en [3]

La prueba de existencia de coálgebras finales para funtores polinomiales Kripke que se hará en el documento fue hecho por primera vez por Aczel y Mendler [4], aunque realmente se demostró algo más general, para los llamados $\omega - accesibles$ o $\omega - peque\tilde{n}os$. Este será el camino que se tomará en este documento para probar la existencia de estas coálgebras finales, así que en términos generales se trata de una reconstrucción más detallada. Aczel y Mendler también hicieron un aporte adicional y es la construcción de una coálgebra final para aquellas que no la tenían (es decir las que no estaban basadas en conjuntos si no en clases) y de esta manera usando una nueva categoría SET de las clases como objetos y funtores $\omega - continuos$ como morfismos, se demostró que en esta categoría el endofuntor $SET \to SET$ siempre tenía una coálgebra final. Como una consecuencia importante de esta idea Aczel demostró que el funtor potencia tiene como coálgebra final (en su extensión a clases) la clase de los conjuntos no bien fundados [5]. Ahora veamos algunas propiedades básicas de las coálgebras finales.

Proposición 4.1
Una coálgebra final $\zeta : Z \to F(Z)$, si existe, es única salvo isomorfismos. Además ella misma es un isomorfismo $\zeta : Z \underset{\to}{\cong} F(Z)$.
Demostración
Esto se puede hacer de una manera más amplia y en general ya que se cumple para el objeto final de cualquier categoría: si 1 y $1'$ son ambos objetos finales, entonces existen únicos morfismos $f : 1 \to 1'$ y $g : 1' \to 1$, y se tienen además dos morfismos $1 \to 1$, uno es la identidad id_1 y el otro es $g \circ f$, pero deben ser iguales por la finalidad de 1, de la misma manera se obtiene $f \circ g = id_{1'}$. Por lo tanto $1 \cong 1'$.
Para probar que $\zeta : Z \underset{\to}{\cong} F(Z)$ necesitamos construir la inversa de ζ, para eso apliquemos nuevamente el funtor $F : \mathbb{C} \to \mathbb{C}$ a la coálgebra y obtenemos el siguiente diagrama:

$$\begin{tikzcd}
F(F(Z)) \arrow[r, "F(f)"] & F(Z) \\
F(Z) \arrow[u, "F(\zeta)"] \arrow[r, "f"'] & Z \arrow[u, "\zeta"']
\end{tikzcd}$$

donde el homomorfismo $f : F(Z) \to Z$ es único por finalidad de ζ. El objetivo es mostrar que f es el inverso de ζ y para esto vamos a considerar $f \circ \zeta : Z \to Z$ y mostrar que es la identidad. Como la identidad id_Z es el único homomorfismo $\zeta \to \zeta$, es suficiente con mostrar que $f \circ \zeta$ es también un homomorfismo $\zeta \to \zeta$ esto sigue del siguiente diagrama:

$$\begin{tikzcd}
F(Z) \arrow[r, "F(\zeta)"] & F(F(Z)) \arrow[r, "F(f)"] & F(Z) \\
Z \arrow[u, "\zeta"] \arrow[r, "\zeta"'] & F(Z) \arrow[u, "F(\zeta)"'] \arrow[r, "f"'] & Z \arrow[u, "\zeta"']
\end{tikzcd}$$

El rectángulo de la derecha conmuta ya que es la definición de f y el de la derecha conmuta obviamente, por lo tanto el rectángulo exterior conmuta, de esta manera $f \circ \zeta = id_Z$. $\zeta \circ f = id_{F(Z)}$ sigue del siguiente cálculo:

$$\begin{aligned}
\zeta \circ f &= F(f) \circ F(\zeta) \\
&= F(f \circ \zeta) \\
&= F(id_Z) \\
&= id_{F(Z)}
\end{aligned}$$

La primera línea es por la finalidad de f, la segunda por ser F un funtor, la tercera la acabamos de probar, y la cuarta se debe otra vez por ser F funtor.
De esta manera queda probado que ζ es un isomorfismo.$\square$

Este resultado viéndolo de forma contrarecíproca dice que si una coálgebra no es isomorfismo entonces no es final, por lo tanto se deduce

4.2 Corolario
El funtor potencia $\mathcal{P}(-) : Sets \to Sets$ no tiene coálgebra final.
Demostración
Por medio de la diagonal de cantor se puede llegar al conocido resultado $\mathcal{P}(\mathbb{N}) \ncong \mathbb{N}$ de esta manera usando la proposición anterior el funtor potencia no tiene coálgebra final.$\square$

Las coálgebras finales de los autómatas deterministas con funtor polinomial $F(X) = X^A \times B$ son de la forma $\langle \zeta_1, \zeta_2 \rangle : B^{A^*} \to (B^{A^*})^A \times B$, hay algunos casos interesantes para observar, por ejemplo: si $A = 1$ entonces $A^* = \mathbb{N}$ que corresponde al funtor $F(X) = X \times B$ que podría ser como el Ejemplo 1.1 pero sin la opción de parar. Su coálgebra final es el conjunto $B^{\mathbb{N}}$ que corresponde a las sucesiones infinitas de elementos de B, en particular si hacemos $B = \mathbb{R}$ se tiene la coálgebra final $\mathbb{R}^{\mathbb{N}} \to \mathbb{R}^{\mathbb{N}} \times \mathbb{R}$.

Como ilustración, veamos un ejemplo interesante de coálgebra final esta vez aplicado al análisis, será el conjunto de las funciones analíticas visto como coálgebra final del funtor $F(X) = X \times \mathbb{R}$. Este objeto final ya tenía otra representación y eran las sucesiones, pero esto no es problema, ya que las dos son estructuralmente isomorfas, se recuerda nuevamente que no importa mucho la apariencia de la coálgebra final, sólo importa la forma como operamos con ella.

Las funciones analíticas $f : \mathbb{R} \to \mathbb{R}$ son las que tienen derivadas de todos los órdenes y coincide con su desarrollo en series de Taylor en la vecindad de cada punto. Sea $\mathcal{A}$ el conjunto de las funciones analíticas con una estructura de coálgebra definida de la siguiente forma:

$$\mathcal{A} \xrightarrow{\quad d \quad} \mathcal{A} \times \mathbb{R}$$

$$f \longmapsto f' \times f(0)$$

Donde f' es la derivada de f, de esta manera el homomorfismo comportamiento $beh_d : \mathcal{A} \to \mathbb{R}^{\mathbb{N}}$ a cada f analítica la envía en la sucesión $\big(f(0), f'(0), f''(0), \ldots, f^{(n)}(0), \ldots\big)$, es decir, su expansión en serie de Taylor. Además, como los coeficientes de Taylor también determinan la función por medio de la fórmula de Taylor, entonces beh_d es un isomorfismo y se puede asumir a $\mathcal{A}$ como coálgebra final.

Parte II
Bisimulaciones e Invariantes

Como ya vislumbramos en el Ejemplo 2.1 y probamos en la Proposición 4.1, es que si una coálgebra es final entonces es un isomorfismo, esto tiene como implicación que en las coálgebras finales no hay diferencia entre los estados y su comportamiento, y por eso no importa mucho saber cómo es la forma de la coálgebra final, lo que más importa en la práctica es el comportamiento, de esta manera se puede hacer una equivalencia entre estados y comportamientos asociados, a esto se le llama bisimilaridad, por lo tanto, si dos estados tienen el mismo comportamiento entonces son indistinguibles desde el punto de vista observacional, es decir, son bisimilares.

Viéndolo desde el punto de vista computacional, una bisimulación es una relación sobre el espacio de estados, de tal manera que se mantiene a través de las sucesivas transiciones, además deja iguales las observaciones. El concepto de bisimilaridad del que hablamos en el párrafo anterior es la bisimulación más grande de todas, así, si dos estados tienen el mismo comportamiento necesariamente deben estar en la relación de bisimilaridad.

El nombre bisimulación viene de una doble simulación; una coálgebra c simula una coálgebra d, y viceversa, pero con las relaciones R y R^{-1}.

5. Introducción a la bisimulación

Consideremos de nuevo la coálgebra del Ejemplo 2.1, es decir, la que muestra la "cabeza" de una sucesión $c : X \to 1 + (A \times X)$, como explicamos anteriormente, lo que intuitivamente queremos de una bisimulación es que se cumpla lo siguiente para dos estados $x, y \in X$:

$$R(x, y) \Longrightarrow R'(c(x), c(y))$$

El problema se encuentra en el espacio de llegada, ya que como vemos, $R \subseteq X \times X$, pero $R' \subseteq (1 + (A \times X)) \times (1 + (A \times X))$, es decir se necesita redefinir la relación R, de tal manera que R' capture esta misma relación pero en un espacio alterado. Esto se hará por medio del "Levantamiento de Relación".

5.1 Definición (Levantamiento de Relación)
Sea $F : \mathit{Sets} \to \mathit{Sets}$ un funtor polinomial, y sean X, Y conjuntos arbitrarios. La aplicación $\mathit{Rel}(F)$ que envía una relación $R \subseteq X \times Y$ a la relación levantada $\mathit{Rel}(F)(R) \subseteq F(X) \times F(Y)$ es definida por inducción sobre la estructura de F.

1. Si F es el funtor identidad, entonces $\mathit{Rel}(F)(R) = R$
2. Si F es un funtor constante $Z \mapsto A$, entonces $\mathit{Rel}(F)(R) = \mathit{Eq}(A) = \{(a, a) \mid a \in A\}$
3. Si $F = F_1 \times F_2$, entonces
   $\mathit{Rel}(F)(R) = \{((u_1, u_2), (v_1, v_2)) \mid \mathit{Rel}(F_1)(R)\,(u_1, v_1) \wedge \mathit{Rel}(F_2)(R)\,(u_2, v_2)\}$
4. Si $F = F_1 + F_2$, entonces
   $\mathit{Rel}(F)(R) = \{(\kappa_1(u), \kappa_1(v)) \mid \mathit{Rel}(F_1)(R)\,(u, v)\} \cup \{(\kappa_2(u), \kappa_2(v)) \mid \mathit{Rel}(F_2)(R)\,(u, v)\}$

5. Si $F = G^A$, entonces $Rel(F)(R) = \{(f, g) \mid \forall a \in A.Rel(G)(R)\,(f(a), g(a))\}$

6. Si $F = \mathcal{P}(G)$, entonces
$Rel(F)(R) = \{(U, V) \mid \forall u \in U.\exists v \in V.Rel(G)(R)(u, v) \wedge \forall v \in V.\exists u \in U.Rel(G)(R)(u, v)\}$

7. Si $F = G^*$, entonces $Rel(F)(R) = \{(\langle u_1, \ldots, u_n\rangle, \langle v_1, \ldots, v_n\rangle) \mid \forall i \leq n.Rel(G)(R)(u_i, v_i)\}$

Apliquemos la definición a algunos casos desarrollados por el autor, por ejemplo al aplicarla al funtor $F(X) = A \times X$ (sucesiones), la relación R se levanta a la relación:

$$Rel(A \times id)(R) = \{((a_1, x), (a_2, y)) \,|\, Rel(A)(R)\,(a_1, a_2) \wedge (x, y) \in R\}$$
$$Rel(A \times id)(R)((a_1, x), (a_2, y)) \iff R(x, y) \wedge a_1 = a_2$$

El resultado coincide con lo que se espera, o sea, la relación entre x, y no cambia, pero además la primera componente debe ser igual para seguir relacionados.

Si se trata del funtor de autómata determinista $F(X) = X^A \times B$, el cálculo sería:

$$Rel(id^A \times B)(R) = \left\{((f, u_2), (g, v_2)) \,|\, Rel(id^A)(R)\,(f, g) \wedge Rel(B)(R)\,(u_2, v_2)\right\}$$

Aplicando otra vez la definición para $Rel(id^A)(R)\,(f, g)$ finalmente se tiene:

$$Rel(id^A \times B)(R)\,((f, u_2), (g, v_2)) \iff \forall a \in A.R\,(f(a), g(a)) \wedge u_2 = v_2$$

Ahora que ya sabemos levantar relaciones, continuemos con el objetivo principal de la sección que es la bisimulación.

5.2 Definición (Bisimulación)
Sea $F : Sets \to Sets$ un funtor polinomial, una **bisimulación** para coálgebras $c : X \to F(X)$ y $d : Y \to F(Y)$ es una relación $R \subseteq X \times Y$ tal que
$(x, y) \in R \Longrightarrow (c(x), d(y)) \in Rel(F)(R)$, para todos los $x \in X, y \in Y$

Notemos que esta definición necesariamente no es de equivalencia, pero otra relación de mucha importancia que introduciremos luego llamada bisimilaridad sí es de equivalencia.

Retomando el ejemplo del Autómata Determinista definido por la coálgebra $\langle \delta, \epsilon \rangle : S \to S^A \times B$, podemos describir cómo son las bisimulaciones para una relación $R \subseteq S \times S$:

$$R(x, y) \Longrightarrow Rel(F)(R)((\delta(x), \epsilon(x)), (\delta(y), \epsilon(y)))$$
$$R(x, y) \Longrightarrow \forall a \in A.R\,(\delta(x)(a), \delta(y)(a)) \wedge \epsilon(x) = \epsilon(y)$$

De esta manera dos estados de un autómata pertenecen a una bisimulación si dejan las mismas observaciones $\epsilon(x) = \epsilon(y)$, y si los estados correspondientes luego de aplicar una transición siguen estando relacionados $R\,(\delta(x)(a), \delta(y)(a))$, lo que parece muy natural de acuerdo a nuestra idea original e intuitiva de una bisimulación.

Para Autómatas normalmente se tiene la siguiente notación:

$x \xrightarrow{a} x'$ para $x' = \delta(x)(a)$
$x \downarrow b$ para $b = \epsilon(x)$

De esta manera se puede reescribir la condición para que R sea una bisimulación de la siguiente manera

$$R(x,y) \Longrightarrow \begin{cases} x \xrightarrow{a} x' \wedge y \xrightarrow{a} y' \Longrightarrow R(x',y') \\ x \downarrow b \wedge y \downarrow c \Longrightarrow b = c \end{cases}$$

Si ahora consideramos Autómatas no Deterministas se puede hacer la misma deducción anterior pero teniendo cuidado con el funtor de la coálgebra $\langle \delta, \epsilon \rangle : S \to \mathcal{P}(S)^A \times B$, una relación R es bisimulación si:

$$R(x,y) \Longrightarrow \begin{cases} x \xrightarrow{a} x' \Longrightarrow \exists y' \; con \; y \xrightarrow{a} y' \wedge R(x',y') \\ y \xrightarrow{a} y' \Longrightarrow \exists x' \; con \; x \xrightarrow{a} x' \wedge R(x',y') \\ x \downarrow b \wedge y \downarrow c \Longrightarrow b = c \end{cases}$$

Esta es la definición de bisimulación encontrada comúnmente en libros sobre autómatas como en [6] y [15], Definición 3.3. La primera línea se conoce como una simulación, y la segunda línea es otra simulación pero considerando la inversa de R, la tercera línea es simplemente igualdad de observaciones.

Como ejemplo consideremos los siguientes LTS no determinista ($B = 1$) aparentemente distintos:

Fig 2 (LTS)

Se puede probar que $R = \{(p_0, q_0), (p_0, q_2), (p_1, q_1), (p_2, q_1)\}$ es una bisimulación, mostrando que R es una simulación y luego que R^{-1} también lo es.

Veamos algunos casos de la prueba:
Como $R(p_0, q_0)$ y $p_0 \xrightarrow{a} p_1$ entonces debe existir un q tal que $R(p_1, q)$ y $q_0 \xrightarrow{a} q$, se puede ver en R y los sistemas de transición, que este q existe y es $q = q_1$.

Igualmente se tiene que como $R(p_0, q_0)$ y $p_0 \xrightarrow{a} p_2$ entonces debe existir un q tal que $R(p_2, q)$ y $q_0 \xrightarrow{a} q$, se puede ver en R y los sistemas de transición que otra vez $q = q_1$.

De esta manera se hacen cada uno de los pares y se prueba que siempre existe uno en el LTS de la derecha que cumple lo que queremos. Esto probaría que R es una simulación, para probar que R^{-1} también es simulación se hace lo mismo pero esta vez partiendo del LTS de la derecha.

Veamos dos casos:
Como $R^{-1}(q_1, p_2)$ y $q_1 \xrightarrow{a} q_1$ entonces debe existir un p tal que $R^{-1}(q_1, p)$ y $p_2 \xrightarrow{a} p$, se puede ver en R^{-1} y los sistemas de transición que este p existe y es $p = p_2$.

Tomando otro caso, $R^{-1}(q_1, p_2)$ y $q_1 \xrightarrow{b} q_2$ entonces debe existir un p tal que $R^{-1}(q_2, p)$ y $p_2 \xrightarrow{b} p$, se puede ver en R^{-1} y los sistemas de transición que este p existe y es $p = p_0$.

6. Bisimulaciones como lazos

En esta sección se dará una definición de bisimulación que es anterior a la que hemos presentado, fue presentada por Aczel y Mendler [7]. Aunque hay otro acercamiento a la bisimulación desde el lenguaje de categorías en la referencia propuesta por Joyal, Nielsen y Winskel, ésta no se tratará en este documento.

La definición de bisimulación que tenemos hasta ahora se basa en el levantamiento de relación, que depende exclusivamente que estemos hablando de funtores polinomiales, por lo tanto cabe la pregunta si no hay una definición más general de bisimulación que incluya los casos cuando un funtor no es polinomial. La respuesta es afirmativa y se conoce como la bisimulación de Aczel-Mendler, para comparar ambas definiciones se hará una restricción a Funtores Polinomiales y se mostrará un teorema que concluye que para funtores polinomiales las dos definiciones coinciden. Antes de probar el teorema, necesitaremos dos resultados intermedios.

Desde este momento se asume que la aplicación $Rel(F) : Rel \to Rel$ es funtorial, esto se demuestra rutinariamente, mostrando que si $(f, g) : R \to S$ es un morfismo en Rel entonces
$(F(f), F(g)) : Rel(F)(R) \to Rel(F)(S)$. Como $\coprod_{f \times g} R \subseteq S$ y usando que el levantamiento de relación es monótono y reserva imágenes inversas (se prueba en los puntos 4 y 5 del lema 9.1 de la parte III) se concluye:

$Rel(F)(R) \subseteq Rel(F)((f \times g)^{-1}(S)) = (F(f) \times F(g))^{-1} Rel(F)(S)$ es decir,
$\coprod_{F(f) \times F(g)} Rel(F)(R) \subseteq Rel(F)(S)$, y $Rel(F) : Rel \to Rel$ es funtorial.

La bisimulación dada por el levantamiento de relación es muy intuitiva en el contexto de los autómatas, pero la definición de Aczel-Mendler (ver definición 6.3) no parece tener relación con ella. Para hacer esto se va a probar que las proyecciones de la relación son homomorfismos de coálgebras, esto dejará las dos coálgebras relacionadas con sus estados posteriores en otra relación $Rel(F)(R)$ que en este caso resulta ser $F(R)$. Es decir, se probará que $R \to F(R)$ también es una coálgebra. Para esto se probarán unos resultados previos que se encargarán de darle la estructura de coálgebra a la bisimulación.

6.1 Lema
Una bisimulación es una $Rel(F) - coálgebra$.
Demostración
Una $Rel(F) - coálgebra$: $R \to Rel(F)(R)$ para $R \subseteq X \times Y$, consiste en dos aplicaciones subyacentes $c : X \to F(X)$ y $d : Y \to F(Y)$ con:

$$\begin{tikzcd} R \arrow[r] \arrow[d, "\gamma"] & Rel(F)(R) \arrow[d, "\delta"] \\ X \times Y \arrow[r, "c \times d"'] & F(X) \times F(Y) \end{tikzcd}$$

Donde γ y δ son ambas inclusiones. Como ambas $c(x)$ y $d(y)$ siguen relacionadas en $Rel(F)(R)$ quiere decir que R es una bisimulación, esto demuestra un si y sólo si ya que si se tiene una bisimulación se puede construir el diagrama que es la definición de coálgebra con el funtor $Rel(F)$ acompañada de los funtores olvido respectivos (γ y δ).$\square$

6.2 Lema:
Sea F un funtor polinomial, y $R \subseteq X \times Y$ una relación arbitraria, escrita vía sus funciones explícitas $\langle r_1, r_2 \rangle : R \rightarrowtail X \times Y$. La relación levantada $Rel(F)(R)$, considerada como un conjunto es una retracción de $F(R)$: Existen funciones $\alpha : Rel(F)(R) \to F(R)$ y $\beta : F(R) \to Rel(F)(R)$ con $\beta \circ \alpha = id_{Rel(F)(R)}$. Más aún, hacen el siguiente diagrama conmutar:

$$\begin{tikzcd} Rel(F)(R) \arrow[rr, bend right, "\alpha"] \arrow[dr] & & F(R) \arrow[ll, bend right, "\beta"'] \arrow[dl, "{\langle F(r_1), F(r_2) \rangle}"] \\ & F(X) \times F(Y) & \end{tikzcd}$$

Demostración
La demostración se hace por inducción sobre la estructura de F, la demostración no es difícil ya que α y β se definen de manera natural pero si es larga porque se debe probar el resultado con todos los casos de la definición de Funtor polinomial. Se mostrará la manera de proceder en el caso $F = F_1 \times F_2$. El resto de casos son similares.
Se asumen las funciones $\alpha_i : Rel(F_i)(R) \to F_i(R)$ y $\beta_i : F_i(R) \to Rel(F_i)(R)$ para $i = 1, 2$. Las funciones α y β son:

$$\alpha\left(\left(u_1, u_2\right), \left(v_1, v_2\right)\right) = \left(\alpha_1\left(u_1, v_1\right), \alpha_2\left(u_2, v_2\right)\right)$$

$$\beta\left(w_1, w_2\right) = \left(\left(\pi_1\beta_1(w_1), \pi_1\beta_2(w_2)\right), \left(\pi_2\beta_1(w_1), \pi_2\beta_2(w_2)\right)\right)$$

donde efectivamente:

$$(\beta \circ \alpha)\left(\left(u_1, u_2\right), \left(v_1, v_2\right)\right) = \left(\left(\pi_1\beta_1\alpha_1\left(u_1, v_1\right), \pi_1\beta_2\alpha_2\left(u_2, v_2\right)\right), \left(\pi_2\beta_1\alpha_1\left(u_1, v_1\right), \pi_2\beta_2\alpha_2\left(u_2, v_2\right)\right)\right)$$

como por inducción se tiene que $\beta_i\alpha_i = id_{Rel(F_i)(R)}$ finalmente resulta:

$$(\beta \circ \alpha)\left(\left(u_1, u_2\right), \left(v_1, v_2\right)\right) = \left(\left(u_1, u_2\right), \left(v_1, v_2\right)\right)$$

$$\beta \circ \alpha = id_{Rel(F)(R)}$$

Si por ejemplo se trata del caso del funtor potencia $F = \mathcal{P}(G)$ con funciones $\alpha_1 : Rel(G)(R) \to G(R)$ y $\beta_1 : G(R) \to Rel(G)(R)$

$$\alpha\,(U,V) = \{\alpha_1\,(u,v) \mid u \in U \wedge v \in V \wedge Rel(G)(R)\,(u,v)\}$$
$$\beta(W) = (\{\pi_1\beta_1(w) \mid w \in W\}\,, \{\pi_2\beta_1(w) \mid w \in W\})$$

y se prueba de la misma manera $\beta \circ \alpha = id_{Rel(F)(R)}$.$\square$

Habiendo dotado a la bisimulación de una estructura algebraica, se revelará la relación con la siguiente definición en el Teorema 6.4. La prueba de este teorema, así como los lemas previos son una reconstrucción del desarrollo presentado en [2], tratando de hacer una reconstrucción de un caso no incluido en la referencia, o un caso que requiera cierto cuidado.

6.3 Definición: (Bisimulación de Aczel-Mendler)
Sea $\langle r_1, r_2\rangle : R \rightarrowtail X \times Y$ una relación, se dice que es una bisimulación de Aczel-Mendler si ella misma es el conjunto subyacente de una coálgebra $e : R \to F(R)$, haciendo r_i homomorfismos de coálgebras:

$$\begin{tikzcd}
F(X) & F(R) \arrow[l, "F(r_1)"'] \arrow[r, "F(r_2)"] & F(Y) \\
X \arrow[u, "c"] & R \arrow[l, "r_1"] \arrow[r, "r_2"'] \arrow[u, "e"'] & Y \arrow[u, "d"']
\end{tikzcd}$$

6.4 Teorema
Sean $c : X \to F(X)$ y $d : Y \to F(Y)$ dos coálgebras de un funtor polinomial F. Una relación $\langle r_1, r_2\rangle : R \rightarrowtail X \times Y$ es una bisimulación para c y d si y sólo si R es una bisimulación de Aczel-Mendler.

Demostración
La demostración se hará con base en el siguiente diagrama:

$$\begin{tikzcd}
R \arrow[r, "f"] \arrow[d, "\langle r_1, r_2\rangle"'] & Rel(F)(R) \arrow[d] \arrow[r, bend right, "\alpha"'] & F(R) \arrow[l, bend right, "\beta"'] \arrow[dl, "\langle F(r_1), F(r_2)\rangle"] \\
X \times Y \arrow[r, "c \times d"'] & F(X) \times F(Y) &
\end{tikzcd}$$

Si R es una bisimulación entonces tiene una coálgebra $Rel(F) - \textit{coálgebra}$ por el lema 6.1, y el cuadro de la izquierda conmuta. Las funciones α, β existen gracias al lema 6.2. Por lo tanto $\alpha \circ f$ es una $F - \textit{coálgebra}$ que convierte a r_i en homomorfismos de coálgebras, de esta manera R también es de Aczel-Mendler.
Si R es de Aczel-Mendler y hay una coálgebra $e : R \to F(R)$ haciendo los r_i homomorfismos, entonces $\beta \circ e : R \to Rel(F)(R)$ mostrando de nuevo que tanto c como d son cerradas, por lo tanto R es una bisimulación.$\square$

A pesar que las dos formulaciones presentadas coinciden en funtores polinomiales, es conveniente mostrar las diferencias. La definición por levantamiento de relación describe las bisimulaciones como relaciones con una propiedad especial, mientras que las bisimulaciones por lazos usa una estructura sobre las relaciones. Como el levantamiento de relación se definió por inducción en la estructura de funtores polinomiales hace que los funtores a los que se le puede aplicar la definición de bisimulación sean los que se pueden generar de esa manera, problema que no tiene la bisimulación por lazos en la que sólo se necesita que las proyecciones conmuten con todo el diagrama.

7. Invariantes

Los invariantes es la versión unaria análoga a la bisimulación, mientras que la bisimulación es una relación binaria que es cerrada. Un invariante es un predicado que es cerrado, es decir, que una vez un invariante sucede, seguirá sucediendo sin importar las operaciones que se hagan. Su importancia radica en que expresan implícitamente propiedades seguras como "es seguro dividir por este número ya que nunca va a ser cero", a menudo expresan que no pasará algo indeseable.

De la misma manera que se encontraron dos formas de definir bisimulaciones, una por acercamiento lógico (levantamiento de relación) y otra por acercamiento estructural (bisimulaciones por lazos), en invariantes también encontramos estas dos aproximaciones, por levantamiento de predicado y por subcoálgebras.

Otro aspecto importante de los invariantes es que se usan para definir operadores de lógica temporal, los cuales se usarán posteriormente para probar la existencia de coálgebras finales.

Un predicado es un subconjunto del espacio de estados, por ejemplo, si se trata de números naturales un predicado podría ser $P(X) = \{x \mid x\, es\, múltiplo\, de\, 3\}$. El levantamiento de predicado es a los invariantes, así como el levantamiento de relación es a las bisimulaciones. De esta manera después de aplicar una transición a un subconjunto del espacio de estados P, nos gustaría que en el conjunto de llegada, es decir, en $\coprod_c P \subseteq F(X)$ sea una buena representación de lo que había antes de hacer la transición.

7.1 Definición (Levantamiento de Predicado)
Sea $F : Sets \to Sets$ un funtor polinomial y sea X un conjunto arbitrario. La aplicación $Pred(F)$ que envía un predicado $P \subseteq X$ a un predicado levantado $Pred(F)(P) \subseteq F(X)$ es definido por inducción sobre la estructura de F.

1. Si F es el funtor identidad, entonces $Pred(F)(P) = P$
2. Si F es un funtor constante $Y \mapsto A$, entonces $Pred(F)(P) = \top_A = (A \subseteq A)$.
3. Si $F = F_1 \times F_2$, entonces $Pred(F)(P) = \{(u, v) \mid Pred(F_1)(P)(u) \wedge Pred(F_2)(P)(v)\}$
4. Si $F = F_1 + F_2$, entonces $Pred(F)(P) = \{\kappa_1\,(u) \,|\, Pred(F_1)(P)(u)\} \cup \{\kappa_2\,(v) \,|\, Pred(F_2)(P)(v)\}$
5. Si $F = G^A$, entonces $Pred(F)(P) = \{f \mid \forall a \in A.Pred(G)(P)(f(a))\}$
6. Si $F = \mathcal{P}(G)$, entonces $Pred(F)(P) = \{U \mid \forall u \in U.Pred(G)(P)(u)\}$
7. Si $F = G^*$, entonces $Pred(F)(P) = \{\langle u_1, \ldots, u_n\rangle \mid \forall i \leq n.Rel(G)(P)(u_i)\}$

Desarrollando el ejemplo de los Autómatas Deterministas el levantamiento de predicado sería de la siguiente manera, considerando a $F = id^A \times B$ entonces
$(f, b) \in Pred(F)(P) \Longleftrightarrow \forall a \in A.f(a) \in P \wedge b \in B$

7.2 Definición (Invariante)
Sea $F : Sets \to Sets$ un funtor polinomial. Un invariante para una coálgebra $c : X \to F(X)$ es un predicado $P \subseteq X$ tal que para $x \in X$ se cumple que $x \in P \Longrightarrow c(x) \in Pred(F)(P)$.

Si ahora aplicamos la definición a la coálgebra $\langle \delta, \epsilon \rangle : S \to S^A \times B$, podemos apreciar que la definición corresponde a lo que intuitivamente creemos que debe ser un invariante en un autómata determinista. P es un invariante si para $x \in S$ se tiene:

$$x \in P \Longrightarrow (\delta(x)(a), \epsilon(x)) \in Pred(F)(P)$$

$$x \in P \Longrightarrow \forall a \in A. \delta(x)(a) \in P \wedge \epsilon(x) \in B$$

Expresando esto en lenguaje computacional tendríamos que P es un invariante si

$$x \in P \wedge x \xrightarrow{a} x' \Longrightarrow x' \in P$$

además garantiza que si $x \downarrow b$ ese b pertenezca al conjunto de las observaciones.
Otro aspecto que se puede observar fácilmente de este ejemplo y que además sugiere el nombre de invariante es que una vez un estado está en un invariante todas sus futuras transiciones están también en el invariante.

Como comentamos anteriormente existe una definición análoga a "bisimulación por lazos" (estructural) pero en invariantes llamadas subcoálgebras, para mostrar la equivalencia con la definición anterior de invariante se necesitan resultados preliminares. Es interesante notar las analogías entre las versiones unaria y binaria del mismo problema, por ejemplo, el lema 7.3 es análogo al lema 6.2, la definición de subcoálgebra es análoga a la bisimulación de Aczel-Mendler, el teorema 7.5 es análogo al teorema 6.4. El lema 7.3, el teorema 7.5 y la definición de subcoálgebra fueron tomadas de [2]

7.3 Lema
Para un funtor polinomial F, el levantamiento de predicado $Pred(F)(P)$ para un predicado $m : P \rightarrowtail X$, es lo mismo que una aplicación funtorial en:

$$\begin{tikzcd}
Pred(F)(P) \arrow[rr, shift right] \arrow[dr] & & F(P) \arrow[ll, shift right] \arrow[dl, "F(m)"] \\
& F(X) &
\end{tikzcd}$$

Demostración
Esto se demostrará mostrando por inducción sobre la estructura del funtor que para $z \in F(X)$ se cumple:

$$z \in Pred(F)(P) \quad \Leftrightarrow \quad \exists!\, z' \in F(P). F(m)(z') = z$$

Es suficiente mostrar la manera de proceder con uno de los casos pues los otros son totalmente análogos. Cuando F es la identidad o el funtor constante es trivial, por lo tanto se mostrará cuando $F = F_1 \times F_2$. En este caso (u, v) hace las veces de z en lo que queremos demostrar por inducción, precisamente porque se trata del funtor producto.

Sea $(u, v) \in F(X)$ entonces $u \in F_1(X) \wedge v \in F_2(X)$

$$\begin{aligned}
(u, v) \in Pred(F)(P) &\Longleftrightarrow u \in Pred(F_1)(P) \wedge v \in Pred(F_2)(P) \\
&\Longleftrightarrow (\exists!\, u' \in F_1(P). F_1(m)(u') = u) \wedge (\exists!\, v' \in F_2(P). F_2(m)(v') = v) \\
&\Longleftrightarrow \exists!(u', v') \in (F_1 \times F_2)(P).(F_1 \times F_2)(m)(u', v') = (u, v)
\end{aligned}$$

Donde la primera línea es la definición de levantamiento de predicado para el producto, la segunda es el paso inductivo, y la tercera reorganizando los resultados como un funtor producto.$\square$

7.4 Definición
Sea F un funtor polinomial. Un predicado $m : P \rightarrowtail X$ sobre el espacio de estados de una coálgebra $c : X \to F(X)$ es una subcoálgebra si existe una estructura de coálgebra $P \to F(P)$ tal que hace a $m : P \rightarrowtail X$ un homomorfismo de coálgebras:

$$\begin{tikzcd} F(P) \arrow[r, "F(m)"] & F(X) \\ P \arrow[u] \arrow[r, "m"'] & X \arrow[u, "c"'] \end{tikzcd}$$

7.5 Teorema
Sea F un funtor polinomial. Un predicado $m : P \rightarrowtail X$ sobre el espacio de estados de una coálgebra $c : X \to F(X)$ es un invariante si y sólo si $P \rightarrowtail X$ es una subcoálgebra.

Demostración
El resultado es inmediato a partir de la definición 7.4 y el Lema 7.3 que es una doble implicación diciendo que $Pred(F)(P)$ es realmente una aplicación funtorial que se puede expresar inmediatamente en términos de $F(P)$.$\square$

Como podemos notar, los conceptos de bisimulación e invariante con sus respectivos levantamientos contienen muchas cosas en común, por eso no sería sorpresa que estuvieran relacionados por algún resultado, primero relacionemos los levantamientos. Las ideas de las demostraciones se tomaron de [2], en la referencia no se hacían todos los casos de la proposición 7.6, así que acá desarrollamos unos distintos, para mostrar la manera de proceder.

7.6 Proposición

1. El levantamiento de relación $Rel(F)$ y el levantamiento de predicado $Pred(F)$ para un funtor polinomial $F : Sets \to Sets$ están relacionados de la siguiente manera:
   $Rel\,(F)\left(\coprod_{\delta_X}(P)\right) = \coprod_{\delta_{F(X)}}(Pred\,(F)\,(P))$
   donde $\delta_X = \langle id_X, id_X\rangle$ de esta manera $\coprod_{\delta_X}(P) = \{\delta_X(x,x) \mid x \in P\} = \{(x,x) \mid x \in P\}$
2. Similarmente,
   $Pred(F)\left(\coprod_{\pi_i}(R)\right) = \coprod_{\pi_i}(Rel\,(F)\,(R))$ donde $\coprod_{\pi_1}(R)$ y $\coprod_{\pi_2}(R)$ son respectivamente el dominio y el codominio de R
3. Como un resultado el levantamiento de predicado se puede expresar en términos del levantamiento de relación:
   $Pred(F)(P) = \coprod_{\pi_i}(Rel\,(F)\,(\coprod_{\delta}(P)))$

Demostración
Tanto el punto 1 como el punto 2 se prueban por inducción sobre la estructura de F. Los casos del funtor identidad y constante triviales, por ejemplo en el punto 1, si $F = id$ ambos lados de la igualdad son el conjunto $\{(x,x) \mid x \in P\}$, si es el F es el funtor constante $X \to A$ entonces ambos lados de la igualdad son el conjunto $\{(a,a) \mid a \in A\}$. Vamos a hacer la prueba de los puntos 1 y 2 para el funtor exponente como ilustración, el resto de casos son análogos.

1. $F = G^A$

$$\begin{aligned}
(f,f) \in \coprod_{\delta_{F(X)}} (Pred\,(F)\,(P)) &\iff \forall a \in A. f(a) \in Pred(G)(P) \\
&\iff \forall a \in (f(a), f(a)) \in \coprod_{\delta_{G(X)}} Pred(G)(P) \\
&\iff \forall a \in (f(a), f(a)) \in Rel(G)\left(\coprod_{\delta_X}(P)\right) \\
&\iff (f,f) \in Rel(F)\left(\coprod_{\delta_X}(P)\right)
\end{aligned}$$

Donde la primera linea es la definición de $\coprod_{\delta_{F(X)}} (Pred\,(F)\,(P))$ aplicada al funtor potencia, la segunda línea se tiene igualmente por la definición de $\coprod_{\delta_{G(X)}} (Pred\,(G)\,(P))$, la tercera es el paso inductivo, y la última es la definición de $Rel(F)$ para un producto.

2. $F = G^A$

$$\begin{aligned}
f \in \coprod_{\pi_1} Rel(F)(R) &\iff \exists g.(f,g) \in Rel(G^A)(R) \\
&\iff \exists g. \forall a \in A.(f(a), g(a)) \in Rel(G)(R) \\
&\iff \forall a \in A. f(a) \in \coprod_{\pi_1} Rel(G)(R) \\
&\iff \forall a \in A. f(a) \in Pred(G)(\coprod_{\pi_1}(R)) \\
&\iff f \in Pred(F)(\coprod_{\pi_1}(R))
\end{aligned}$$

Donde la primera línea es la definición de estar en el dominio, la segunda es el levantamiento de relación, la tercera nuevamente la definición de dominio, la cuarta la hipótesis inductiva y por último la definición de levantamiento de predicado.

3. Si usamos los dos puntos anteriores y teniendo presente que $\coprod_{\pi_i} \circ \coprod_{\delta} = id$ se tiene:

$$\begin{aligned}
Pred(F)(P) &= \coprod_{\pi_i} \coprod_{\delta} Pred(F)(P) \\
&= \coprod_{\pi_i} Rel\,(F)\left(\coprod_{\delta_X}(P)\right)
\end{aligned}$$

□

Como ya hemos visto los conceptos de bisimulación y predicado son análogos además comparten muchas propiedades de estructura, así que esperariamos un resultado que los relacione de alguna manera, en la siguiente proposición se dan dos puntos de relación, el primero es cómo se puede producir un invariante a partir de una bisimulación, el segundo produce una bisimulación a partir de un invariante.

7.7 Proposición
Considere dos coálgebras $c : X \to F(X)$ y $d : Y \to F(Y)$ de un funtor polinomial F. Entonces:

1. Si $R \subseteq X \times Y$ es una bisimulación, entonces su dominio $\coprod_{\pi_1}(R)$ y su codominio $\coprod_{\pi_2}(R)$ son invariantes.

2. Si $P \subseteq X$ es un invariante, entonces $\coprod_\delta P = \{(x, x) \mid x \in P\} \subseteq X \times X$ es una bisimulación.

Demostración

1. Si R es un bisimulación entonces:

$$\begin{aligned} \coprod_c \coprod_{\pi_1} R &= \coprod_{\pi_1} \coprod_{c\times d} R \\ &\subseteq \coprod_{\pi_1} Rel(F)(R) \\ &= Pred(F)\left(\coprod_{\pi_i} R\right) \end{aligned}$$

   Donde en la segunda línea se usa el hecho de R ser bisimulación y en la tercera se usa la parte 2 de la proposición 7.6

2. Si $P \subseteq X$ es un invariante entonces:

$$\begin{aligned} \coprod_{c\times c} \coprod_{\delta} P &= \coprod_{\delta} \coprod_{c} P \\ &\subseteq \coprod_{\delta} Pred(F)(P) \\ &= Rel(F)(\coprod_{\delta} P) \end{aligned}$$

   La segunda línea es por ser P invariante, y la tercera por la parte 2 de la proposición 7.6$\square$

Si nos preguntamos por la razón de estudiar la definición de invarianza en este trabajo, la importancia radica en que es crucial para mostrar existencia de coálgebras finales, en especial cuando se trata de un funtor potencia finito, ya tenemos que dar cuenta de estados futuros asegurando que las propiedades se conserven, esto se verá en la parte III del documento. Por ejemplo, por medio del levantamiento de predicado se pueden definir operadores temporales, que me dicen lo que puede suceder en el futuro, como si la próxima vez sucederá algo, si ese algo sucederá por siempre, o si sucederá alguna vez.

8. Operadores Temporales

Los operadores temporales son los operadores de la lógica temporal que es una extensión de la lógica donde los valores de verdad dependen del tiempo, por ejemplo el valor de verdad de una frase como el cielo es azul depende del tiempo, de día es cierta pero de noche es falsa. Los operadores son de dos clases, los conocidos operadores lógicos ($\neg, \wedge, \vee, \rightarrow$) y los operadores modales que son los encargados del tiempo, algunos de ellos son "la próxima vez", "de aquí en adelante" y "eventualmente" denotados por $\bigcirc, \Box, \Diamond$ respectivamente. Aunque estos operadores se usarán en el documento en su versión unaria, también cabe decir hay una versión binaria, por ejemplo, uno de ellos el "hasta que" es el encargado de que algo se cumpla hasta que otra cosa pase. Una introducción a estos operadores se puede encontrar en la página de la enciclopedia de filosofía de Stanford [8].

8.1. Razonamiento hacia adelante

Los operadores temporales se aplican a predicados, por ejemplo el operador $\bigcirc P$ significa los estados que en la próxima interacción aún cumplirán con el predicado P. Algunos de los operadores temporales se definen en términos de la coálgebra y del levantamiento de predicado (del funtor implícitamente), y otros en términos de otros operadores.

Si se puede razonar de algunos estados presentes que en el futuro cumplen alguna propiedad, también se puede razonar acerca de los estados presentes que en el pasado algunos estados cumplían la propiedad. Este tipo de razonamiento hacia atrás se puede hacer definiendo una adjunción a izquierda del funtor $Pred(F)(P)$. Uno de estos operadores de razonamiento hacia el pasado será un paso importante en la demostración de la existencia de coálgebras finales.

8.1.1 Definición ("Próxima vez")
Sea $c : X \to F(X)$ una coálgebra de un funtor polinomial F. Definimos el operador "próxima vez" $\bigcirc : \mathcal{P}(X) \to \mathcal{P}(X)$ como:

$$\begin{aligned} \bigcirc P &= c^{-1}(Pred(F)(P)) \\ &= \{x \in X \mid c(x) \in Pred(F)(P)\} \end{aligned}$$

De esta definición se sigue inmediatamente que P es un invariante si y sólo si $P \subseteq \bigcirc P$. Es decir, si $x \in P$ invariante, por definición cumple la propiedad para estar en $\bigcirc P$, la otra implicación también es obvia. Esta equivalencia se usa muchas veces como la definición de invariante.
Ahora vamos a definir el operador "de aquí en adelante", que hace lo que su nombre indica: selecciona los estados del presente que cumplen con el predicado y además lo siguen cumpliendo en todos los estados futuros.

8.1.2 Definición ("De aquí en adelante")
Sea $c : X \to F(X)$ una coálgebra de un funtor polinomial F. Definimos el operador "De aquí en adelante" $\Box : \mathcal{P}(X) \to \mathcal{P}(X)$ como:

$$x \in \Box P \Longleftrightarrow \exists Q \subseteq X.\, Q\; es\; un\; invariante\; para\; c \wedge Q \subseteq P \wedge Q(x)$$

Es decir, $\Box P$ es la unión de todos los invariantes que están contenidos en P, por esto $\Box P$ a veces se llama el *invariante más grande* contenido en P (predicado). Que sea el más grande no se discute, pero que sea invariante no es claro, aunque esto es una propiedad de los invariantes; ser cerrados bajo uniones arbitrarias, que sigue inmediatamente de la definición:

$$P_i \subseteq c^{-1}(Pred(F)(P_i) \subseteq c^{-1}(Pred(F)(\cup P_i)$$

la primera contenencia es por la definición de invarianza, la segunda por ser la aplicación $Pred(F)$ monótona. Como cada P_i está contenido en $c^{-1}(Pred(F)(\cup P_i)$, también su unión y esto prueba que $\Box P$ es un invariante.

Nota: La prueba de la monotonía de $Pred(F)$ se puede hacer por inducción en la estructura, incluso se puede probar un resultado más general y es que $Pred(F)$ preserva intersecciones arbitrarias.
Otro operador temporal es "eventualmente" P, y significa que en algún momento en el futuro un estado cumplirá la condición del predicado, la notación de este operador es $\Diamond P$. La definición es más sencilla y se puede dar en términos de $\Box$,

$$\Diamond P = \neg\Box\neg P$$

Que significa: es falso que en todos los estados futuros no se dé P, es decir, que se dará en algún momento. Existen más operadores hacia adelante pero estos son los que vamos a manejar en el trabajo.

8.2. Razonamiento hacia atrás.

A continuación vamos a definir los operadores que permiten razonar hacia atrás, para esto necesitamos un concepto como "reducción de predicados" $\underleftarrow{Pred}(F)(P)$, análogo al "levantamiento de predicados" $Pred(F)(P)$ pero como el nombre sugiere es una aplicación dual. Este nuevo funtor es lo que en teoría de categorías se conoce como una adjunción a izquierda del funtor $Pred(F)(P)$, concepto que explicaremos a continuación.

Una adjunción consiste básicamente en un par de funtores que van en direcciones opuestas:

$$\begin{tikzcd} \mathbb{C} \arrow[r, bend right, "F"'] & \mathbb{D} \arrow[l, bend right, "G"'] \end{tikzcd}$$

que satisfacen ciertas propiedades, como ejemplo empezaremos con un tipo especial de adjunción definida en copos (o posets en inglés) C, D, o sea, funciones monótonas $f : C \to D$ y $g : D \to C$ entre ellos. Estas funciones forman una Conexión de Galois, si cumplen:

$$f(x) \leq y \Longleftrightarrow x \leq g(y)$$

Otro ejemplo con respecto al trabajo y expuesto por el autor son los objetos exponenciales en una categoría cerrada cartesiana, con la relación Currying antes descrita.

$$(X \times Y) \to Z \Leftrightarrow X \to Z^Y$$

Como vemos tiene una forma análoga a la conexión de Galois, con los funtores $F(X) = (X \times Y)$ y $G(Z) = Z^Y$. Este tipo de conexiones aparecen en muchas ramas de la matemática, otros ejemplos son Construcción de Álgebras Libres y Coálgebras Colibres (que veremos más adelante) con funtores olvido, en topología está la Compactificación de Stone-Čech con un funtor olvido entre Espacios de Hausdorff compactos y Espacios Topológicos y su funtor adjunto a izquierda que sirve para volver compacto un conjunto que en un principio no lo era.

Viendo estos casos recurrentes se ha hecho una definición general llamada adjunción que resume toda esta información con la generalidad de las categorías.

8.2.1 Definición (Adjunción)
Considere dos categorías $\mathbb{C}, \mathbb{D}$ con funtores $F : \mathbb{C} \to \mathbb{D}$ y $G : \mathbb{D} \to \mathbb{C}$ entre ellas. Estos funtores forman una adjunción, que se denota como $F \dashv G$, si para todos los objetos $X \in \mathbb{C}$ y $Y \in \mathbb{D}$ existe una correspondencia biyectiva entre morfismos de $\mathbb{C}\, y\, \mathbb{D}$

$$F(X) \to Y \iff X \to G(Y)$$

.

Que son naturales en X y Y . En este caso se dice que F es el adjunto a izquierda y G es el adjunto a derecha. Si escribimos la correspondencia biyectiva como funciones: $\psi : \mathbb{D}(F(X), Y) \xrightarrow{\cong} \mathbb{C}(X, G(Y))$, el requerimiento de naturalidad quiere decir que para morfismos $f : X' \to X$ en $\mathbb{C}$ y $g : Y \to Y'$ en $\mathbb{D}$ y para $h : F(X) \to Y$, se tiene:

$$G(g) \circ \psi(h) \circ f = \psi(g \circ h \circ F(f))$$

Esta condición de naturalidad que a primera vista no parece tan natural, realmente está diciendo que evaluar ψ en el homomorfismo $F(X') \to Y'$ es lo mismo que el homomorfismo $X' \to G(Y')$, es decir, la biyección se sigue cumpliendo aún en otros elementos de $\mathbb{C}, \mathbb{D}$ con morfismos f y g. Así como lo explica el siguiente diagrama:

$$\begin{tikzcd}
X' \arrow[r] \arrow[d, "f"] & F(X') \arrow[r] \arrow[d, "F(f)"] & Y' \arrow[r] & G(Y') \\
X \arrow[r] \arrow[rrr, bend right, "\psi(h)"'] & F(X) \arrow[r, "h"] & Y \arrow[r] \arrow[u, "g"] & G(Y) \arrow[u, "G(g)"]
\end{tikzcd}$$

En copos el requerimiento se tiene automáticamente ya que las funtores F y G preservan el orden, así si $x' \leq x$ y $y \leq y'$ entonces $F(x') \leq F(x)$ y $G(y) \leq G(y')$ por preservar orden, por lo tanto si $F(x) \leq y$ entonces $F(x') \leq y'$ es decir, cumple naturalidad.

El siguiente razonamiento hecho por el autor del documento se hace para legitimar la definición de una adjunción a izquierda. Para construir una adjunción a izquierda necesitamos que en los predicados exista un funtor que conserve la contenencia. Por ejemplo, como se dijo anteriormente (no es complicado mostrarlo por inducción en la estructura de F), se tiene que si $P \subseteq Q$ entonces $Pred(P) \subseteq Pred(Q)$. Además del resultado más general: $Pred(F)(\cap P_i) = \cap Pred(F)(P_i)$ y usando el teorema de la sección 4.2 del capítulo I del libro [9] que dice que si $g : A \to B$ es una aplicación que preserva el orden entre copos entonces si A tiene todos sus ínfimos y g los preserva entonces g tiene adjunto a la izquierda, en particular $\mathcal{P}(X)$ tiene todas sus intersecciones y $Pred(F)(P) : \mathcal{P}(X) \to \mathcal{P}(F(X))$ las preserva entonces tiene adjunción a la izquierda. Esto prueba la existencia de la adjunción que necesitamos para el razonamiento hacia atrás y le daremos la notación sugestiva $\underleftarrow{Pred}(F)(P)$. Esto es gráficamente:

$$\begin{tikzcd}
\mathcal{P}(X) \arrow[r, bend right, "Pred(F)(P)"'] & \mathcal{P}(F(X)) \arrow[l, bend right, "\underleftarrow{Pred}(F)(P)"']
\end{tikzcd}$$

Y la correspondencia biyectiva es: $\underleftarrow{Pred}(F)(Q) \subseteq P \iff Q \subseteq Pred(F)(P)$, la definición de la reducción de predicado por estructura es la siguiente:

8.2.2 Definición (Reducción de predicado)
Sea $F : Sets \to Sets$ un funtor polinomial y sea $F(X)$ un conjunto arbitrario. La aplicación $\underleftarrow{Pred}(F)$ que envía un predicado $Q \subseteq F(X)$ a un predicado reducido $\underleftarrow{Pred}(F)(Q) \subseteq X$ es definido por inducción sobre la estructura de F.

1. Si F es el funtor identidad, entonces $\underleftarrow{Pred}(F)(Q) = Q$
2. Si F es un funtor constante $Y \mapsto A$, entonces $\underleftarrow{Pred}(F)(Q) = \bot_A = (\emptyset \subseteq A)$.

3. Si $F = F_1 \times F_2$, entonces $\underleftarrow{Pred}(F)(Q) = \underleftarrow{Pred}(F_1)(\coprod_{\pi_1}(Q)) \cup \underleftarrow{Pred}(F_2)(\coprod_{\pi_2}(Q))$
4. Si $F = F_1 + F_2$, entonces $\underleftarrow{Pred}(F)(Q) = \underleftarrow{Pred}(F_1)(\kappa_1^{-1}(Q)) \cup \underleftarrow{Pred}(F_2)(\kappa_2^{-1}(Q))$
5. Si $F = G^A$, entonces $\underleftarrow{Pred}(F)(Q) = \underleftarrow{Pred}(G)(\{f(a) \mid a \in A \wedge Q(f)\})$
6. Si $F = \mathcal{P}(G)$, entonces $\underleftarrow{Pred}(F)(Q) = \underleftarrow{Pred}(G)(\cup Q)$
7. Si $F = G^*$, entonces $\underleftarrow{Pred}(F)(Q) = \underleftarrow{Pred}(G)(\{u_i \mid Q(\langle u_1, \ldots, u_n \rangle)\})$

La definición se hace teniendo en cuenta que cada línea cumpla con la biyección.

En la subsección anterior vimos los operadores temporales, resulta que con la definición de $\underleftarrow{Pred}(F)$ tenemos la contraparte de razonamiento hacia atrás.

8.2.3 Definición (Operadores hacia atrás)
Para una coálgebra $c : X \to F(X)$ de un funtor polinomial, y un predicado $P \subseteq X$, definimos nuevos predicados llamados "última vez", "alguno en el pasado" y "de aquí hacia atrás", respectivamente:

$$\underleftarrow{\bigcirc} P = \underleftarrow{Pred}(F)(\{c(x) \mid x \in P\})$$

$$\underleftarrow{\Diamond} P = \{x \in P \mid \forall Q \; invariante \wedge P \subseteq Q \Longrightarrow Q(x)\}$$

$$\underleftarrow{\Box} P = \neg \underleftarrow{\Diamond} \neg P$$

$\underleftarrow{\Diamond} P$ también se puede ver como el menor invariante que contiene a P, y se cumple para los estados que tienen un estado pasado para el cual se cumplía P. Estos operadores (los de atrás y los de adelante) fueron definidos a partir la adjunción$\underleftarrow{Pred}(F) \dashv Pred(F)$, así que no es de extrañar que existan pares de operadores que sean adjuntos. En los casos que hemos presentado las adjunciones son:

$$\begin{array}{ccc} \underleftarrow{\bigcirc} & \dashv & \bigcirc \\ \underleftarrow{\Diamond} & \dashv & \Box \\ \underleftarrow{\Box} & \dashv & \Diamond \end{array}$$

Estas definiciones de razonamiento temporal parecen decir lo que queremos de ellas, pero para probar que realmente es así, vamos a aplicarla a un caso muy sencillo de objeto que captura el concepto de hacer una transición.

Primero algo de notación, sea un conjunto X y $x \in X$, usaremos la siguiente notación:
$(\bullet = x) = \{y \in X \mid y = x\} = \{x\}$
$(\bullet \neq x) = \{y \in X \mid y \neq x\} = \neg (\bullet = x)$

Al primero se le conoce como predicado "singleton", y al segundo predicado "no-singleton". Con base en estos predicados se define la relación de transición:

8.2.4 Definición (Relación de Transición)
Sea $c : X \to F(X)$ una coálgebra de un funtor polinomial F. Sobre los estados $x, x' \in X$ se definirá la relación de transición por medio del operador $\bigcirc$, de la siguiente manera:

$$
\begin{aligned}
x \to x' &\iff x \in (\neg \bigcirc \neg)(\bullet = x') \\
&\iff x \notin \bigcirc (\bullet \neq x')
\end{aligned}
$$

esto quiere decir que hay una transición si y sólo si existe un estado sucesor de x que es igual a x'. Cabe notar que esta relación no es ni transitiva ni simétrica, pero esto se puede subsanar por medio de su clausura reflexiva-transitiva, que denotaremos como $x \to^* x'$.
Ahora veamos un lema que involucra estos predicados y nos ayudará a ver en otro resultado cómo se comportan los operadores temporales en UTS (Sistemas de transición sin marcas) los puntos 1 y 2 los hizo el autor del texto:

8.2.5 Lema

1. Para un predicado $P \subseteq X$, se cumple $P \subseteq (\bullet \neq x) \iff \neg P(x)$.
2. Para una función $f : Y \to X$, se cumple $f^{-1}(\bullet \neq x) = \bigcap_{y \in f^{-1}(x)} (\bullet \neq y)$.
3. Para un funtor polinomial F y un predicado $Q \subseteq F(X)$, se tiene
$\underleftarrow{Pred}(F)(Q) \iff \{x \in X \mid Q \nsubseteq Pred(F)(\bullet \neq x)\}$

Demostración

1. Lógicamente se tiene el resultado ya que si $(y \in P \Longrightarrow y \neq x) \iff (y = x \Longrightarrow y \notin P) \iff \neg P(x)$.
2. $y' \in \bigcap_{y \in f^{-1}(x)} (\bullet \neq y) \iff \forall y \in f^{-1}(\bullet = x).y' \in (\bullet \neq y) \iff \forall y \in f^{-1}(\bullet = x).y' \neq y \iff y' \in f^{-1}(\bullet \neq x)$
3. Usando la parte 1 y que $\underleftarrow{Pred}(F) \dashv Pred(F)$:

$$
\begin{aligned}
x \in \underleftarrow{Pred}(F)(Q) &\iff \underleftarrow{Pred}(F)(Q) \nsubseteq (\bullet \neq x) \\
&\iff Q \nsubseteq Pred(F)(\bullet \neq x) . \square
\end{aligned}
$$

8.2.6 Proposición
La relación de transición $\to$ de la definición 8.2.4 inducida por una coálgebra $X \to F(X)$ satisface para un predicado $P \subseteq X$:

1. $\bigcirc P = \{x \in X \mid \forall x'.x \to x' \Rightarrow P(x')\}$
2. $\square P = \{x \in X \mid \forall x'.x \to^* x' \Rightarrow P(x')\}$
3. $\lozenge P = \{x \in X \mid \exists x'.x \to^* x' \wedge P(x')\}$
4. $\underleftarrow{\bigcirc} P = \{x \in X \mid \exists y.y \to x \wedge P(y)\}$
5. $\underleftarrow{\lozenge} P = \{x \in X \mid \exists y.y \to^* x \wedge P(y)\}$
6. $\underleftarrow{\square} P = \{x \in X \mid \forall y.y \to^* x \Rightarrow P(y)\}$

Demostración
En esta prueba los puntos 2,4 y algunos detalles del punto 5 son aportes del autor

1. Se hace usando el lema 8.2.5

$$
\begin{aligned}
x \in \bigcirc P &\iff c(x) \in Pred(F)(P)\\
&\iff \{c(x)\} \subseteq Pred(F)(P)\\
&\iff \underleftarrow{Pred}(F)(\{c(x)\}) \subseteq P\\
&\iff \forall x'.x' \in \underleftarrow{Pred}(F)(\{c(x)\}) \Rightarrow P(x')\\
&\iff \forall x'.\,\{c(x)\} \nsubseteq Pred(F)\,(\bullet \neq x') \Rightarrow P(x')\\
&\iff \forall x'.c(x) \notin Pred(F)\,(\bullet \neq x') \Rightarrow P(x')\\
&\iff \forall x'.x \rightarrow x' \Rightarrow P(x')
\end{aligned}
$$

   La primera línea es la definición de $\bigcirc$, la tercera es por la adjunción, la quinta es por el lema 8.2.5 y la séptima definición de $x \rightarrow x'$.

2. Un predicado $Q \subseteq X$ es un invariante si y sólo si $Q \subseteq \bigcirc Q$ trivialmente, y como $\Box P$ es un invariante, cumple $\Box P \subseteq \bigcirc \Box P$, se tiene: $x \in \Box P \Rightarrow x' \in \Box P \Rightarrow x'' \in \Box P \Rightarrow \ldots \Rightarrow x^n \in \Box P$ reiterando la parte 1. Además como $\Box P \subseteq P$, se concluye que $x' \in P, x'' \in P, \ldots, x^n \in P$. Es decir, que aplicando varias veces la parte 1 se tiene $\subseteq$.
   Para probar $\supseteq$ notemos que $\{x \in X \mid \forall x'.x \rightarrow^* x' \Rightarrow P(x')\} \subseteq \bigcirc\,\{x \in X \mid \forall x'.x \rightarrow^* x' \Rightarrow P(x')\}$ claramente, por lo tanto ese conjunto es un invariante, además contiene a P, entonces debe estar contenido en el mayor invariante de todos $\Box P$.

3. Sigue fácilmente luego de aplicar la definición:$\Diamond P = \neg\Box\neg P$ y la parte 2.

4. Análogo a 1.

$$
\begin{aligned}
\underleftarrow{\bigcirc} P &= \underleftarrow{Pred}(F)(\{c(x) \mid y \in P\})\\
&= \underleftarrow{Pred}(F)(\bigcup_{y \in P} \{c(y)\})\\
&= \bigcup_{y \in P} \underleftarrow{Pred}(F)(\{c(y)\})\\
&= \Big\{x \in X \mid \exists y \in P.x \in \underleftarrow{\bigcirc}\,(\bullet = y)\Big\}\\
&= \{x \in X \mid \exists y.y \rightarrow x \wedge P(y)\}
\end{aligned}
$$

   La primera línea es la definición, la tercera es porque la adjunta a izquierda preserva uniones, veamos que esto es cierto:

$$
\begin{aligned}
\underleftarrow{Pred}(F)(\cup X_i) \subseteq Y &\Leftrightarrow \cup X_i \subseteq Pred(F)(Y)\\
&\Leftrightarrow \forall i.X_i \subseteq Pred(F)(Y)\\
&\Leftrightarrow \forall i.\underleftarrow{Pred}(F)(X_i) \subseteq Y\\
&\Leftrightarrow \cup \underleftarrow{Pred}(F)(X_i) \subseteq Y
\end{aligned}
$$

   Esto junto con el hecho que $\cup \underleftarrow{Pred}(F)(X_i) \subseteq \underleftarrow{Pred}(F)(\cup X_i)$ porque preserva orden, hace que

$\underleftarrow{Pred}(F)(\cup X_i)$ sea la menor cota superior de $\cup\underleftarrow{Pred}(F)(X_i)$, por tanto $\cup\underleftarrow{Pred}(F)(X_i) = \underleftarrow{Pred}(F)(\cup X_i)$.

5. Sea $Q = \{x \in X \mid \exists y.y \rightarrow^* x \wedge P(y)\}$, $P \subseteq Q$ si suponemos que no se hace transición, además si $x \in Q$ entonces existe y tal que $y \rightarrow^* x$ y $P(y)$, si además suponemos que $x \rightarrow x'$ entonces por transitividad se tiene que $y \rightarrow^* x'$ y $x' \in Q$, esto significa que Q es un invariante. Si hay otro invariante Q' que contiene a P, si $x \in Q$ con $y \rightarrow^* x$ y $P(y)$, entonces $y \in Q'$ y por ser invariante $x \in Q'$. De esta manera se prueba que $Q \subseteq Q'$ y Q es el menor invariante que tiene P, es decir: $\underleftarrow{\Diamond} P = Q$.

6. Sigue de la definición $\underleftarrow{\Box} P = \neg \underleftarrow{\Diamond} \neg P$ y el punto 5.$\Box$

Esto finalmente demuestra que los predicados hacen lo que queremos que se haga, como este ejemplo con relaciones de transición muestra, y es seleccionar unos estados del presente que van a tener un comportamiento inmediatamente, en algún momento o eventualmente.
Finalmente se demostrará un resultado realizado por el autor de este documento que se usará en la tercera parte.

8.2.7 Proposición
Sea P un predicado sobre el espacio de estados de una coálgebra de un funtor polinomial entonces $\underleftarrow{\Diamond} P = \bigcup_{n\in\mathbb{N}} \underleftarrow{\bigcirc}^n P$. Donde $\underleftarrow{\bigcirc}^0 P = P$ y $\underleftarrow{\bigcirc}^{n+1} P = \bigcirc\underleftarrow{\bigcirc}^n P$.
Desmostración

$$\begin{aligned}
x \in \underleftarrow{\Diamond} P &\iff \exists y.y \rightarrow^* x \wedge P(y)\\
&\iff \exists n \in \mathbb{N}, \exists y', \ldots, y^n, y \in P \; tal\; que\; (y \rightarrow y^n) \wedge \ldots \wedge (y' \rightarrow x) \wedge P(y)\\
&\iff \exists n \in \mathbb{N}, \exists y' \in P.y' \rightarrow x \wedge \underleftarrow{\bigcirc}^{n-1} P(y')\\
&\iff \exists n \in \mathbb{N}.x \in \bigcirc\underleftarrow{\bigcirc}^{n-1} P(x)\\
&\iff x \in \bigcup_{n\in\mathbb{N}} \underleftarrow{\bigcirc}^n P
\end{aligned}$$

Parte III

Coálgebras Finales en Funtores Polinomiales

En las dos partes anteriores del trabajo se hizo una introducción a los conceptos y su importancia en las coálgebras, no se intentó demostrar muchos resultados acerca de las propiedades de los conceptos involucrados como bisimulación o invariantes para no desviar mucho la atención, en cambio se hizo incapié en los conceptos mediante ejemplos, además de demostrar diferentes formulaciones para tener un panorama un poco más amplio. Esta parte del trabajo se encargará de probar algunas proposiciones para poder demostrar el resultado principal, que trata sobre la existencia de coálgebras finales para Funtores Polinomiales Kripke.

9. Propiedades de $Rel(F)$ y las bisimulaciones

El levantamiento de relación cumple propiedades muy especiales, sólo considerando su definición se nota que pretende preservar muchas propiedades y comportarse estructuralmente bien, el siguiente lema resume algunas de éstas:

9.1 Lema
Sea $F : Sets \to Sets$ un funtor polinomial y $h : X \to Z$ y $m : Y \to W$ funciones arbitrarias. $Rel(F)$ satisface las siguientes propiedades:

1. Preserva la relación de igualdad: $Rel(F)(Eq(X)) = Eq(F(X))$.
2. Conmuta con relaciones inversas: $Rel(F)(R^{-1}) = (Rel(F)(R))^{-1}$.
3. Preserva la composición de relaciones: $Rel(F)(R \circ S) = Rel(F)(R) \circ Rel(F)(S)$.
4. Es monótona: $R \subseteq S \Rightarrow Rel(F)(R) \subseteq Rel(F)(S)$
5. Conmuta con imágenes inversas: para $R \subseteq Z \times W$:
$$Rel(F)((h \times m)^{-1}(R)) = (F(h) \times F(m))^{-1} Rel(F)(R)$$
6. Conmuta con imágenes directas: para $R \subseteq X \times Y$,
$$Rel(F)(\coprod_{h \times m}(R)) = \coprod_{F(h) \times F(m)} Rel(F)(R)$$
7. Preserva relaciones kernel, para una función $f : X \to Y$:
$$Rel(F)(Ker(f)) = Ker(F(f))$$
8. Preserva la relación $Graph(f) = (f \times id_Y)^{-1}(Eq(Y))$, para una función $f : X \to Y$.
$$Rel(F)(Graph(f)) = Graph(F(f))$$
9. Preserva la relación pullback $Pb(f, g) = (f \times g)^{-1}(Eq(Z))$ para $f : X \to Z$ y $g : Y \to Z$.
$$Rel(F)(Pb(f, g)) = Pb(F(f), F(g))$$

10. Preserva la relación imagen $Im(\langle f,g\rangle) = \coprod_{f\times g}(Eq(Z))$ para $f : Z \to X$ y $g : Z \to Y$

$$Rel(F)(Im(\langle f,g\rangle)) = Im(\langle F(f), F(g)\rangle)$$

Demostración:
Las pruebas se harán por inducción sobre la estructura del funtor, se hará para el caso especial del funtor $F = G^A$ ya que se debe tener cuidado en la parte 3 por la existencia de una función y por lo tanto usar el axioma de elección. El resto de casos son análogos.

1. Supongamos que la propiedad se cumple para el funtor G, entonces:

$$\begin{array}{rcl} (f,f) \in Eq(G^A(X)) & \Leftrightarrow & f \in G^A(X) \\ & \Leftrightarrow & \forall a \in A. f(a) \in G(X) \\ & \Leftrightarrow & \forall a \in A.(f(a), f(a)) \in Eq(G(X)) \\ & \Leftrightarrow & \forall a \in A.(f(a), f(a)) \in Rel(G)(Eq(X)) \\ & \Leftrightarrow & (f,f) \in Rel(G^A(Eq(X))) \end{array}$$

   Todas las líneas son definiciones, excepto la cuarta que es el paso inductivo.

2. Se razona igual:

$$\begin{array}{rcl} (f,g) \in Rel(G^A(R^{-1})) & \Leftrightarrow & \forall a \in A.(f(a), g(a)) \in Rel(G)(R^{-1}) \\ & \Leftrightarrow & \forall a \in A.(f(a), g(a)) \in Rel(G)(R)^{-1} \\ & \Leftrightarrow & \forall a \in A.(g(a), f(a)) \in Rel(G)(R) \\ & \Leftrightarrow & (g,f) \in Rel(G^A(R)) \\ & \Leftrightarrow & (f,g) \in Rel(G^A(R))^{-1} \end{array}$$

   El paso inductivo es la segunda línea.

3. En esta prueba se usará el axioma de elección en la cuarta línea, y el paso inductivo en la segunda:

$$\begin{array}{rcl} (f,g) \in Rel(G^A)(R \circ S) & \Leftrightarrow & \forall a \in A.(f(a), g(a)) \in Rel(G)(R \circ S) \\ & \Leftrightarrow & \forall a \in A.(f(a), g(a)) \in Rel(G)(R) \circ Rel(G)(S) \\ & \Leftrightarrow & \forall a \in A. \exists b \in G.(f(a), b) \in Rel(G)(R) \wedge (b, g(a)) \in Rel(G)(S) \\ & \Leftrightarrow & \exists h \in G^A. \forall a \in A.(f(a), h(a)) \in Rel(G)(R) \wedge (h(a), g(a)) \in Rel(G)(S) \\ & \Leftrightarrow & \exists h \in G^A.(f,h) \in Rel(G^A)(R) \wedge (h,g) \in Rel(G^A)(S) \\ & \Leftrightarrow & (f,g) \in Rel(G^A)(R) \circ Rel(G^A)(S) \end{array}$$

   La busqueda de estas propiedades es muy sugestiva, y claramente se está buscando que $Rel(F)$ preserve relaciones de equivalencia, esto es importante, ya que si definimos por medio de una relación que dos estados son equivalentes, esto nos permite asegurar que van a seguir siendo equivalentes en las sucesivas transiciones.

4. Suponiendo $R \subseteq S$ se prueba,

$$\begin{array}{rcl} (f,g) \in Rel(F)(R) & \Rightarrow & \forall a \in A.(f(a), g(a)) \in Rel(G)(R) \\ & \Rightarrow & \forall a \in A.(f(a), g(a)) \in Rel(G)(S) \\ & \Rightarrow & (f,g) \in Rel(F)(S) \end{array}$$

5. El paso inductivo está en la segunda línea, la cuarta línea es el funtor exponente aplicado a una función.

$$
\begin{aligned}
(f,g) \in Rel(F)((h\times m)^{-1}(R)) &\Leftrightarrow \forall a \in A.(f(a),g(a)) \in Rel(G)((h\times m)^{-1}(R))\\
&\Leftrightarrow \forall a \in A.(f(a),g(a)) \in (G(h)\times G(m))^{-1}Rel(G)(R)\\
&\Leftrightarrow \forall a \in A.((G(h)\circ f)(a),(G(m)\circ g)(a)) \in Rel(G)(R)\\
&\Leftrightarrow (G(h)^A \circ f, G(m)^A \circ g) \in Rel(G^A)(R)\\
&\Leftrightarrow (f,g) \in (F(h),F(m))^{-1}Rel(F)(R)
\end{aligned}
$$

6. En esta prueba nuevamente es necesario usar axioma de elección en la cuarta línea e inducción en la segunda.

$$
\begin{aligned}
(f,g) \in Rel(F)(\coprod_{h\times m}(R)) &\Leftrightarrow \forall a \in A.(f(a),g(a)) \in Rel(G)(\coprod_{h\times m}(R))\\
&\Leftrightarrow \forall a \in A.(f(a),g(a)) \in \coprod_{G(h)\times G(m)} Rel(G)(R)\\
&\Leftrightarrow \forall a \in A.\exists u,v \in Rel(G)(R).(f(a),g(a)) = (G(h)(u),G(m)(v))\\
&\Leftrightarrow \exists k,l \in Rel(G)(R)^A.\forall a \in A.(f(a),g(a)) = (G(h)(k(a)),G(m)(l(a)))\\
&\Leftrightarrow \exists k,l.(f,g) = (G^A(h)(k),G^A(m)(l))\\
&\Leftrightarrow (f,g) \in \coprod_{F(h)\times F(m)} Rel(F)(R)
\end{aligned}
$$

7. En esta prueba no hace falta hacer inducción ya que se probaron los resultados previos en los puntos anteriores de este mismo lema. La segunda línea es por el punto 5, y la tercera línea es por el punto 1.

$$
\begin{aligned}
Rel(F)(Ker(f)) &= Rel(F)((f\times f)^{-1}Eq(Y))\\
&= (F(f)\times F(f))^{-1}Rel(F)(Eq(Y))\\
&= (F(f)\times F(f))^{-1}Eq(F(Y))\\
&= Ker(F(f))
\end{aligned}
$$

8. Análoga al punto 7, pero se usa $f \times id_Y$ en lugar de $f \times f$

9. Análoga al punto 7, pero se usa $f \times g$ en lugar de $f \times f$.

10. En este punto se usa que Rel conmuta con las imágenes directas punto 6,

$$
\begin{aligned}
Rel(F)(Im(\langle f,g\rangle)) &= Rel(F)(\coprod_{f\times g}(Eq(Z)))\\
&= \coprod_{F(f)\times F(g)} Rel(F)(Eq(Z))\\
&= \coprod_{F(f)\times F(g)} Eq(F(Z))\\
&= Im(\langle F(f),F(g)\rangle)
\end{aligned}
$$

En la tercer línea se usó el punto 1.□

El levantamiento de relación fue la base sobre la que se fundó la noción de bisimulación, ya que probamos sus buenas propiedades vamos a aplicarlas a las bisimulaciones que igualmente goza de buenas propiedades como ser cerradas para: inversión, composición, uniones arbitrarias, imágenes inversas, imágenes directas. En el siguiente lema se demostrará lo que se usará posteriormente, las ideas generales de las demostraciones se tomaron de [2].

Antes del Lema se introducirá un concepto muy importante en coálgebras y es la relación de bisimilaridad, la importancia radica en que es la bisimulación más grande de todas, y ayuda para formalizar la idea que observacionalmente dos coálgebras sean indistinguibles. Esto se hace con el principio de prueba coinductiva que dice que dos estados tienen el mismo comportamiento si y sólo si hay una bisimulación que los contiene es decir si y solo si esos estados son bisimilares. Resultado que se demostrará en la sección 15.

9.2 Definición (Bisimilaridad)
Sean $c : X \to F(X)$ y $d : Y \to F(Y)$ dos coálgebras de un funtor polinomial. La relación de bisimilaridad es la unión de todas las bisimulaciones:

$$x \leftrightarrow y \Longleftrightarrow \exists R \subseteq X \times Y. \, R \, es \, una \, bisimulación \, para \, c \, y \, d, \; y \; R(x, y)$$

Se ha dicho que la relación de bisimilaridad es una bisimulación, esto se puede demostrar comprobando que la unión arbitraria de bisimulaciones es bisimulación. Es un hecho que se probará en el siguiente lema.

9.3 Lema

1. Sean $c : X \to F(X)$, $c' : X' \to F(X')$ y $d : Y \to F(Y)$ coálgebras de un funtor polinomial F. Las bisimulaciones son cerradas bajo:
   a) Inversión: si $R \subseteq X \times Y$ es una bisimulación, entonces también lo es $R^{-1} \subseteq Y \times X$.
   b) Composición: si $R \subseteq X \times X'$ y $S \subseteq X' \times Y$ son bisimulaciones, entonces también lo es $S \circ R \subseteq X \times Y$.
   c) Uniones Arbitrarias: si $R_i \subseteq X \times Y$ es una bisimulación para cada $i \in I$, entonces también lo es $\bigcup_{i \in I} R_i \subseteq X \times Y$
2. $f : X \to Y$ es un homomorfismo de coálgebras si y solo si la relación $Graph(f) = (f \times id_Y)^{-1}(Eq(Y))$ es una bisimulación.
3. Si $f : X \to Y$ es un homomorfismo entonces $Ker(f)$ es una bisimulación de equivalencia.
4. Para homomorfismos $f : X \to Z$, $g : Y \to Z$ la relación pullback $Pb(f, g)$ es una bisimulación.
5. $(c \leftrightarrow d)^{-1} \subseteq d \leftrightarrow c$ y $(c \leftrightarrow c') \circ (c' \leftrightarrow d) \subseteq c \leftrightarrow d$.
6. La relación de bisimilaridad $c \leftrightarrow c \subseteq X \times X$ para una coálgebra es una bisimulación de equivalencia.
7. Para homomorfismos $f : Z \to X$, $g : Z \to Y$ la relación imagen $Im(\langle f, g \rangle)$ es una bisimulación.

Demostración

1. Sea R una bisimulación entonces

a) $R \subseteq (c \times d)^{-1}(Rel(F)(R))$, por lo tanto,

$$\begin{array}{rcl} R^{-1} & \subseteq & \left((c \times d)^{-1}(Rel(F)(R))\right)^{-1} \\ & = & (d \times c)^{-1}\left(Rel(F)(R)^{-1}\right) \\ & = & (d \times c)^{-1}(Rel(F)(R^{-1})) \end{array}$$

La tercera línea es por la parte 2 del lema 9.1

b) Si $(x, y) \in (S \circ R)$ con $(x, x') \in R$ y $(x', y) \in S$, entonces como R y S son bisimulaciones se tiene que $(c(x), c'(x')) \in Rel(F)(R)$ y $(c'(x'), d(y)) \in Rel(F)(S)$. Entonces por el punto 3 del lema 9.1 se tiene que $(c(x), d(y)) \in Rel(F)(S \circ R)$, lo que quiere decir que $S \circ R$ es bisimulación.

c) Sea R_i bisimulación, entonces

$$\begin{array}{rcl} \bigcup_{i \in I} R_i & \subseteq & \bigcup_{i \in I} (c \times d)^{-1}(Rel(F)(R_i)) \\ & = & (c \times d)^{-1}(\bigcup_{i \in I} Rel(F)(R_i)) \\ & \subseteq & (c \times d)^{-1}(Rel(F)(\bigcup_{i \in I} R_i)) \end{array}$$

La primera línea es la definición de bisimulación para R_i la segunda es porque las imágenes inversas preservan la unión, y la tercera es por el punto 4 del Lema 9.1, es decir, porque $Rel(F)$ es monótona. Este resultado tiene como consecuencia que la relación de bisimilaridad es una bisimulación.

2. $Graph(f) = (f \times id_Y)^{-1}(Eq(Y))$ es una bisimulación si y sólo si:

$$\begin{array}{rcl} Graph(f) \subseteq (c \times d)^{-1} Graph(F(f)) & \Longleftrightarrow & \forall x, y. f(x) = y \Rightarrow F(f)(c(x)) = d(y) \\ & \Longleftrightarrow & \forall x. F(f)(c(x)) = d(f(x)) \\ & \Longleftrightarrow & f\ es\ homomorfismo \end{array}$$

La primer línea se tiene por el punto 8 del lema 9.1, el resto son simplemente las definiciones de *Graph* y de homomorfismo

3. Como el $Ker(f) = Graph(f) \circ Graph(f)^{-1}$ se tiene el resultado por el punto 2 y el punto 1b.

4. Usando el punto 9 del lema 9.1 se necesita probar que $Pb(f, g) \subseteq (c \times d)^{-1}(Pb(F(f), F(g)))$, es decir que $f(x) = g(y)$ implica $F(f)(c(x)) = F(g)(d(y))$ pero esto se tiene ya que f y g son homomorfismos.

5. Se tiene trivialmente porque la bisimilaridad es una bisimulación por el punto 1c y los resultados por los puntos 1a y 1b.

6. La simetría y la transitividad siguen del punto 5, la propiedad reflexiva se tiene si se toma el $Ker(id_X)$, y como vimos en el punto 3 también es bisimulación. De esta manera se tiene la relación de equivalencia.

7. Se necesita probar la inclusión $Im(\langle f,g\rangle) \subseteq (c \times d)^{-1}(Im(\langle F(f), F(g)\rangle))$ por punto 10 lema 9.1. Esto quiere decir que tenemos que probar que para cada $z \in Z$ hay un $w \in F(Z)$ con $c(f(z)) = F(f)(w)$ y $d(g(z)) = F(g)(w)$, pero como f, g son homomorfismos se puede tomar $w = e(z)$ donde $e : Z \to F(Z)$.$\square$

10. Coálgebras observables

En el ejemplo 2.1 de la parte I del documento se muestra explícitamente una coálgebra final. Como se vio, existe una función que llamamos comportamiento y denotamos por $beh_c : X \to A^\infty$ que asocia a cada estado una representación en la coálgebra final, y significa a grandes rasgos cómo se ve el estado, es decir, su observación. Una situación muy particular sucede cuando dos estados tienen el mismo comportamiento sólo si son iguales estos dos estados, esto matemáticamente está expresando que su función comportamiento es inyectiva y en la práctica significa que estamos viendo una representación de la máquina en la coálgebra final. Cuando se trata de Autómatas Deterministas estos se llaman *observables* si cumplen la condición de inyectividad.

Este capítulo tratará varios resultados relacionados con la noción de observabilidad definida en el contexto más general de las coálgebras.

10.1 Definición (Observable)
Una coálgebra $c : X \to F(X)$ se llama observable si su relación de bisimilaridad $c \leftrightarrow c$ es la igualdad sobre X.

Anteriormente se ha dicho que un autómata determinista es observable si su función comportamiento es inyectiva, esto tiene sentido sólo si el funtor asociado $F(X) = X^A \times B$ tiene coálgebra final, como se dijo al final de la sección 4 la coálgebra final de este funtor existe y es de la forma $B^{A^*} \to (B^{A^*})^A \times B$. Para relacionar esta definición con la presentada en 10.1, se emplea un resultado muy importante que se probará en la IV parte del documento y dice que dos estados son bisimilares si y sólo si tienen el mismo comportamiento. Como una coálgebra es observable si y sólo si su relación de bisimilaridad es la igualdad, se tiene que dos estados bisimilares son iguales, y tienen el mismo comportamiento, así la función comportamiento es inyectiva y la definición coincide en el caso de los autómatas deterministas. La presente sección es una adaptación de [2] haciendo énfasis en los detalles que se necesitan para resultados futuros.

A continuación veremos una manera de volver una coálgebra en coálgebra observable haciendo un cociente con su relación de bisimilaridad, muchos detalles de la prueba fueron hechos por el autor.

10.2 Proposición
Sea $F : Sets \to Sets$ un funtor polinomial. Para una coálgebra $c : X \to F(X)$ y una bisimulación de equivalencia $R \subseteq X \times X$, el conjunto cociente X/R es subyacente de una única estructura de coálgebra, $c/R : X/R \to F(X/R)$, haciendo de la aplicación canónica $[-]_R : X \to X/R$ un homomorfismo de coálgebras:

$$\begin{tikzcd} F(X) \arrow[r, "F([-]_R)"] & F(X/R) \\ X \arrow[u, "c"] \arrow[r, "{[-]_R}"'] & X/R \arrow[u, "c/R"'] \end{tikzcd}$$

Demostración
La coálgebra c/R se define como $F([-]_R) \circ c$ sobre los elementos de X/R, mostremos que está bien definida de esta manera, es decir, que $(x, y) \in R$ implica $F([-]_R)(c(x)) = F([-]_R)(c(y))$, para esto se intentará mostrar que $R \subseteq Ker(F([-]_R) \circ c)$.

$$\begin{aligned} R &\subseteq (c \times c)^{-1}(Rel(F)(R)) \\ &= (c \times c)^{-1}(Rel(F)(Ker([-]_R))) \\ &= (c \times c)^{-1} Ker(F([-]_R)) \\ &= Ker(F([-]_R \circ c)) \end{aligned}$$

La primera línea es la definición de bisimulación, la segunda es por ser R una bisimulación de equivalencia, la tercera por el punto 7 del lema 9.1, y la cuarta es la misma tercera pero viendo el Kernel desde X.$\Box$.

Ahora como sabemos que la bisimilaridad es de equivalencia (punto 6 Lema 9.3), podemos usarla para definir una coálgebra que además tiene la propiedad de ser observable. La siguiente proposición es hecha por el autor.

10.3 Proposición
Si $F : Sets \to Sets$ un funtor polinomial, $c : X \to F(X)$ una coálgebra y $\leftrightarrow$ su relación de bisimilaridad, entonces $c/\leftrightarrow : (X/\leftrightarrow) \to F(X/\leftrightarrow)$ es una coálgebra observable.
Demostración
La coálgebra existe por la proposición 10.2 y por el punto 6 del lema 9.3 que dice que la bisimilaridad es de equivalencia, además es observable porque si $a, b \in X/\leftrightarrow$ entonces ellos están relacionados si y sólo si pertenecen a la misma clase de equivalencia, pero como a, b son clases por pertenecer a $X/\leftrightarrow$ entonces deben ser iguales, esto es,

$$a \leftrightarrow b \Leftrightarrow [a] = [b] \Leftrightarrow a = b$$

Por lo tanto la coálgebra es observable.$\Box$

Como observación importante cabe notar que como $[-] : X \to X/\leftrightarrow$ es un homomorfismo entonces $Graph([-])$ es una bisimulación (punto 2 lema 9.3). Así, un estado $x \in X$ es bisimilar a su clase de equivalencia $[x] \in X/\leftrightarrow$, esto significa que volver una coálgebra observable no cambia su comportamiento.

11. Algunos resultados sobre $CoAlg(F)$

En esta sección se darán algunas definiciones y resultados acerca de la estructura de la categoría $CoAlg(F)$ para funtores polinomiales definidos sobre conjuntos que serán de utilidad luego.
Algunos fueron tomados de [10] y otros tantos como detalles fueron aportes del autor de este docu-

mento.

11.1 Definición (Coproducto Finito)
Sea $\mathbb{C}$ una categoría, se dice que tiene coproductos finitos si tiene coproducto binario y tiene elemento inicial.

Aprovechando la definición podemos mostrar la construcción de coproducto finito para la categoría $CoAlg(F)$ que luego nos será de utilidad.
La categoría $Sets$ tiene coproductos finitos, ya que su elemento inicial es el conjunto vacío ϕ porque de él sale sólo una función a cada elemento (función vacío), además el coproducto binario se define por la unión disjunta, es decir, si X y Y son conjuntos su coproducto está formado por
$X+Y = \{(x,1)\,|x\in X\}\cup\{(y,2)\,|y\in Y\}$, donde en vez de proyecciones como en productos tiene coproyecciones, definidas así: $\kappa_1(x)=(x,1)$ y $\kappa_2(y)=(y,2)$.

Análogamente se puede definir un coproducto para la categoría $CoAlg(F)$, donde la coálgebra inicial se puede tomar como $i:\phi\to F(\phi)$, en el siguiente diagrama el morfismo Φ es único por definición y conmuta en el caso de funtores polinomiales.

$$\begin{tikzcd} F(\phi) \arrow[r, "F(\Phi)"] & F(X) \\ \phi \arrow[u, "i"] \arrow[r, "\Phi"'] & X \arrow[u, "c"'] \end{tikzcd}$$

Y el coproducto binario se puede definir de la siguiente manera, si $c:X\to F(X)$ y $d:Y\to F(Y)$ son dos coálgebras $[F(\kappa_1)\circ c, F(\kappa_2)\circ d]: X+Y\to F(X+Y)$ es su coproducto, como se ve en el siguiente diagrama.

$$\begin{tikzcd} F(X) \arrow[r, "F(\kappa_1)"] & F(X+Y) & F(Y) \arrow[l, "F(\kappa_2)"'] \\ X \arrow[u, "c"] \arrow[r, "\kappa_1"'] & X+Y \arrow[u] & Y \arrow[u, "d"'] \arrow[l, "\kappa_2"] \end{tikzcd}$$

Y existe ya que el coproducto en $Sets$ existe y lo único que hemos hecho es levantarlo a $CoAlg(F)$, además cumple la propiedad universal; si hay otra coálgebra $Z\to F(Z)$ tal que tiene dos coproyecciones p_1,p_2 a $c:X\to F(X)$ y $d:Y\to F(Y)$ respectivamente, entonces existe un único homomorfismo de $X+Y\to F(X+Y)$ a $Z\to F(Z)$. Todo lo que se ha hecho es heredar las propiedades de $Sets$.

11.2 Definición (Coecualizador)
Sea $\mathbb{C}$ una categoría y dos morfismos "paralelos" $f,g:X\to Y$, un coequalizador de f,g es un objeto Q junto con un morfismo $q:Y\to Q$ tal que $q\circ f=q\circ g$. Más aún, el par (Q,q) debe ser universal en el sentido que dado cualquier otro par (Q',q') que cumple $q'\circ f=q'\circ g$ existe un único morfismo $u:Q\to Q'$ tal que el siguiente diagrama conmuta:

$$\begin{tikzcd} X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \arrow[r, "q"] \arrow[dr, "q'"'] & Q \arrow[d, "u"] \\ & & Q' \end{tikzcd}$$

Igual que en el caso de los coproductos los coecualizadores también se pueden construir en $CoAlg(F)$, consideremos dos homomorfismos $f : (X, c) \to (Y, d)$ y $g : (X, c) \to (Y, d)$, por lo tanto debemos encontrar una coálgebra (Q, e) y un homomorfismo $q : (Y, d) \to (Q, e)$ tal que $q \circ f = q \circ g$.
Como por definición $f : X \to Y$ y $g : X \to Y$ son funciones en $Sets$ existe un coecualizador $q : Y \to Q$ en $Sets$. Y teniendo en cuenta el siguiente diagrama:

$$\begin{tikzcd}
F(X) \arrow[r, shift left, "F(f)"] \arrow[r, shift right, "F(g)"'] & F(Y) \arrow[r, "F(q)"] & F(Q) \\
X \arrow[u, "c"] \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \arrow[u, "d"'] \arrow[r, "q"'] & Q \arrow[u, "e"']
\end{tikzcd}$$

$F(q) \circ d : Y \to F(Q)$ coequaliza ya que

$$\begin{aligned}
F(q) \circ d \circ f &= F(q) \circ F(f) \circ c \\
&= F(q \circ f) \circ c \\
&= F(q \circ g) \circ c \\
&= F(q) \circ F(g) \circ c \\
&= F(q) \circ d \circ g
\end{aligned}$$

Además, $q : Y \to Q$ es coecualizador por lo tanto hay una coálgebra única $e : Q \to F(Q)$ que hace el diagrama conmutar, de esta manera (Q, e) es una coálgebra y $q : (Y, d) \to (Q, e)$ un homomorfismo. La propiedad universal se cumple sin complicaciones.

Los resultados presentados en los últimos párrafos se pueden resumir en el siguiente Teorema.

11.3 Teorema

1. Sea F un funtor polinomial, la categoría $CoAlg(F)$ tiene sus coequalizadores y todos sus coproductos.
2. La categoría $CoAlg(F)$ es cocompleta.$\square$

Demostración

1. Los coequalizadores y coproductos binarios ya están, los coproductos de una familia indexada se pueden generalizar de la misma manera ya que estos también existen en $Sets$.
2. Usando el teorema de existencia para colímites se tiene el resultado, la demostración del dual de este teorema es decir para límites se puede encontrar en [1] Pag 109.$\square$

Otro resultado que se usará luego, es uno que tiene que ver con una generalización del concepto de sobreyectividad de una función llamado epimorfismo, así que se definirá y se relacionará con el concepto de coecualizadores que se acabó de presentar.

11.4 Definición (Epimorfismo)
Un epimorfismo es un morfismo $f : X \to Y$ tal que para todos los morfismos $g_1, g_2 : Y \to Z$ cumple que $g_1 \circ f = g_2 \circ f$ implica que $g_1 = g_2$.

Como ilustración examinemos los epimorfismos en la categoría $Sets$, para ver que corresponden a las funciones sobreyectivas.
Si $f : X \to Y$ es una función sobreyectiva y $g_1, g_2 : Y \to Z$ dos funciones que cumplen $g_1 \circ f = g_2 \circ f$. Entonces para cada elemento $y \in Y$ podemos encontrar un elemento $x \in X$ tal que $f(x) = y$, además $g_1(y) = g_1(f(x)) = g_2(f(x)) = g_2(y)$, por lo tanto $g_1 = g_2$.

En el otro sentido, si $f : X \to Y$ es un epimorfismo considere el conjunto $\{0, 1\}$ y las siguientes funciones $g_1, g_2 : Y \to \{0, 1\}$, definidas así:

$$\begin{aligned} g_1(y) &= 1 \; si \; y \in f(X) \\ g_1(y) &= 0 \; si \; y \notin f(X) \\ g_2(y) &= 1 \; \forall y \in Y \end{aligned}$$

Entonces $g_1 \circ f = g_2 \circ f = 1$ por lo tanto $g_1 = g_2$ y $f(X) = Y$, es decir, f es sobreyectiva.

De esta manera, como los epimorfismos $f : X \to Y$ corresponden a funciones sobreyectivas en $Sets$, se tiene que $Y = X/\sim$, donde $\sim$ es una relación de equivalencia definida $x \sim x'$ cuando $f(x) = f(x')$, como precisamente $(x, x') \in Ker(f)$ entonces f está coecualizando las proyecciones del Kernel sobre X. Sintetizando, si f es un epimorfismo es el coecualizador de su propio Kernel.

A continuación se demostrará el resultado para la categoría $CoAlg(F)$ el cual es un aporte del autor del documento.

11.5 Proposición
Si $f : X \to Y$ es un epimorfismo en una categoría $CoAlg(F)$ para un funtor polinomial sobre los conjuntos, entonces f es el coecualizador de su propio par kernel $p_1, p_2 : Ker(f) \to X$.
Demostración
La prueba es análoga a la construcción del coecualizador en $Coalg(F)$, sólo que el diagrama es de la siguiente forma:

$$\begin{tikzcd} Ker(F(f)) \arrow[r, shift left, "F(p_1)"] \arrow[r, shift right, "F(p_2)"'] & F(X) \arrow[r, "F(f)"] & F(X/\sim) \\ Ker(f) \arrow[u, "i"] \arrow[r, shift left, "p_1"] \arrow[r, shift right, "p_2"'] & X \arrow[u, "c"] \arrow[r, "f"'] & X/\sim \arrow[u, "c/\sim"'] \end{tikzcd}$$

donde la relación de equivalencia está definida como se dijo en el párrafo anterior, esto se debe a que f es un epimorfismo. Además, se puede probar igual como se hizo luego de la definición 11.2 que $F(f) \circ c$ también es un coecualizador de p_1 y p_2, así como lo es f por ser el caso de la categoría $Sets$. Sólo faltaría probar la propiedad universal pero esta se tiene de inmediato por la proposición 10.2 ya que $\sim$ es de equivalencia, y define una única estructura de coálgebra $c/\sim: X/\sim \to F(X/\sim)$.□

12. Existencia de coálgebra final para KPF

El resultado que se probará en esta sección es el principal del documento, aunque nuestro objetivo es garantizar la existencia de coálgebras finales para funtores polinomiales Kripke finitos, el resultado es válido para una gama más amplia de funtores llamados ω-accesibles también llamados finitarios. Esta versión de la prueba fue reconstruida con base en [2] que a su vez se apoyó en [11].

La prueba es técnica, pero la idea general es primero encontrar una candidata a coálgebra final, para esto se construirá una que es observable, por medio de un cociente con la relación de bisimilaridad. Lo segundo es demostrar que esa coálgebra es final, es decir, que desde una arbitraria X existe un único homomorfismo que llega a ella, esto se logra por medio del coproducto de subcoálgebras de X, estas subcoálgebras se definen de manera que su conjunto subyacente sea un subconjunto de $\mathbb{N}$ isomorfo a un invariante de X, en particular este invariante es contable para funtores ω-accesibles, lo que facilita la conexión con los Funtores Polinomiales Kripke (KPF).

12.1 Definición (Funtor ω-accesible)
Un funtor $F : Sets \to Sets$ es ω-accesible si satisface para cada conjunto X,

$$F(X) = \bigcup_{U \in P_{fin}(X)} F(U) = \bigcup \{F(U) \mid U \subseteq X \; es \; finito\}$$

Por lo tanto nos queda mostrar que los funtores polinomiales Kripke finitos cumplen esta definición.

12.2 Lema
Cada funtor polinomial finito $F : Sets \to Sets$ es ω-accesible
Demostración
Los funtores polinomiales preservan inclusiones, hecho que se demuestra fácilmente por inducción. Por ejemplo si $F = id$ y $U \subseteq X$ entonces $F(U) = U \subseteq X = F(X)$. Si $F = F_1 \times F_2$ es el funtor producto, $U \subseteq X$ y F_1, F_2 preservan orden entonces $F(U) = F_1(U) \times F_2(U) \subseteq F_1(X) \times F_2(X) = F(X)$. El resto de casos se hacen de manera análoga.
Teniendo esto en cuenta como $U \subseteq X$, se tiene que $F(U) \subseteq F(X)$ para todo U finito, por lo tanto también $\bigcup F(U) \subseteq F(X)$
Para demostrar $\bigcup F(U) \supseteq F(X)$ se hace igualmente por inducción, por ejemplo cuando $F = G^*$ y $\langle y_0, \ldots, y_n \rangle \in F(X) = G(X)^*$ entonces por hipótesis inductiva hay subconjuntos finitos $U_i \subseteq X$ con $y_i \in G(U_i)$, por lo tanto su unión U es un conjunto finito con $y_i \in G(U)$ para cada i, entonces $\langle y_0, \ldots, y_n \rangle \in F(U) = G(U)^*$ como se buscaba. Los otros casos se demuestran similarmente. Es remarcable que aquí es donde se usa que el funtor potencia sea finito, por lo tanto el funtor potencia general no es ω-accesible.$\square$

En este momento vamos a retomar un operador temporal que se introdujo en la sección 8.2 y es el operador "de aquí hacia atrás" $\underleftarrow{\Diamond} P$, pero en el caso que $P \equiv (\bullet = x)$. Este operador nos servirá para hallar una subcoálgebra tal que su conjunto es contable.

12.3 Lema
Para cada coálgebra $c : X \to F(X)$ de un funtor polinomial finito, y para cada uno de sus estados $x \in X$, el predicado invariante de estados sucesores de x, $\underleftarrow{\Diamond}(x) = \{y \in X \mid x \to^* y\}$ es contable .
Demostración
Primero recordemos que $\underleftarrow{\Diamond} P = \{y \in X \mid \exists z. z \to^* y \wedge P(z)\}$ como en este caso $P \equiv (\bullet = x)$, el operador se transforma en $\underleftarrow{\Diamond}(x) = \{y \in X \mid \exists z. z \to^* y \wedge z = x\}$ de esta manera queda con la apariencia que necesitamos en el lema, es curioso además que un operador hacia atrás está dando cuenta de estados futuros.
Para demostrar que es contable retomemos la proposición 8.2.7 que dice que $\underleftarrow{\Diamond}(x) = \bigcup_{n \in \mathbb{N}} \underleftarrow{\bigcirc}^n (\bullet = x)$, de esta manera sólo tenemos que mostrar que para cada $x \in X$ el conjunto de sucesores inmediatos

$\underleftarrow{\bigcirc}(x) = \{y \mid x \to y\} = \underleftarrow{Pred}(F)(\{c(x)\})$ es finito. Como F es ω-accesible por el lema 12.2, el levantamiento de predicado es una aplicación funtorial por el lema 7.3, y además como $\underleftarrow{Pred}(F) \dashv Pred(F)$, se tiene,

$$\begin{aligned} c(x) \in F(X) = \bigcup_{U \in P_{fin}(X)} F(U) = \bigcup_{U \in P_{fin}(X)} Pred(F)(U) \quad &\Rightarrow \quad \{c(x)\} \subseteq Pred(F)(U) \\ &\Rightarrow \quad \underleftarrow{Pred}(F)(\{c(x)\}) \subseteq U \end{aligned}$$

Para algún U finito por lo tanto $\underleftarrow{\bigcirc}(x)$ es finito y $\underleftarrow{\Diamond}(x)$ es contable.$\square$

12.4 Teorema (Existencia coálgebra final)
Cada funtor polinomial Kripke finito $F : Sets \to Sets$ tiene una coálgebra final.
Demostración
Formemos la colección de todos los estados y F-coálgebras sobre subconjuntos de $\mathbb{N}$.

$$W = \{(n, N, d : N \to F(N)) \mid n \in \mathbb{N} \wedge N \subseteq \mathbb{N}\}$$

se puede definir una estructura coalgebraica asociada a W por medio de la coproyección $\kappa_{(N,d)} : N \to W$ donde $n \mapsto (n, N, d)$ para cada coálgebra $d : N \to F(N)$, esta coproyección es un homomorfismo, basta tomar $\xi(n, N, d) = (F(\kappa_{N,d}) \circ d)(n)$, por lo tanto $\xi : W \to F(W)$

$$\begin{tikzcd} F(N) \arrow[r, "F(\kappa_{N,d})"] & F(W) \\ N \arrow[u, "d"] \arrow[r, "\kappa_{N,d}"'] & W \arrow[u, "\xi"'] \end{tikzcd}$$

Ahora por la proposición 10.3 si tomamos $Z = W/\leftrightarrow$ el cociente por la relación de bisimilaridad obliga a la coálgebra $\zeta : Z \to F(Z)$ a ser observable, es decir, que la bisimilaridad sobre Z es la igualdad.
A continuación se probará que ζ es final.
Por el lema 12.3 se tiene que $\underleftarrow{\Diamond}(x)$ es contable, por lo tanto es isomorfo a un subconjunto de $\mathbb{N}$, por medio de $\varphi_x : N_x \underset{\rightarrow\leftarrow}{\cong} \underleftarrow{\Diamond}(x)$. Por teorema 7.5 el invariante tiene una estructura de coálgebra $c_x : \underleftarrow{\Diamond}(x) \to F(\underleftarrow{\Diamond}(x))$, por lo tanto llamaremos $c_x^N : N_x \to F(N_x)$ a la coálgebra inducida por el isomorfismo sobre N_x.

$$\begin{tikzcd} F(X) & F(\underleftarrow{\Diamond}(x)) \arrow[l] & F(N_x) \arrow[l, "\cong", "F(\varphi_x)"'] \arrow[r, "F(\kappa_{(N_x, c_x^N)})"'] & F(W) \arrow[r] & F(W/\leftrightarrow) \\ X \arrow[u, "c"] & \underleftarrow{\Diamond}(x) \arrow[l] \arrow[u, "c_x"] & N_x \arrow[l, "\cong", "\varphi_x"'] \arrow[u, "c_x^N"] \arrow[r, "\kappa_{(N_x, c_x^N)}"'] & W \arrow[u, "\xi"] \arrow[r] & W/\leftrightarrow \arrow[u, "\zeta"] \end{tikzcd}$$

Como c_x^N depende de $x \in X$ por teorema 11.3 podemos formar su coálgebra coproducto cuyo espacio de estados es $\coprod_{x \in X} N_x$, además existe para cada $x \in X$ una componente de la cotupla que se puede definir del anterior diagrama, entonces $\pi : \coprod_{x \in X} N_x \to X$ es epimorfismo ya que x recorre todo X. De esto tenemos el siguiente diagrama:

$$\begin{tikzcd} \coprod_{x \in X} N_x \arrow[rr, "\pi"] \arrow[dr] & & X \arrow[d, "f"] \\ & W \arrow[r] & Z \end{tikzcd}$$

Como π es un epimorfismo y por la proposición 11.5 entonces es coecualizador de su propio Kernel $Ker(\pi)$. El Kernel de un homomorfismo es una bisimulación por el punto 3 del lema 9.3, entonces como es bisimulación el morfismo $\coprod_{x\in X} N_x \to W \to Z$ es constante para valores $(u,v) \in Ker(\pi)$, esto hace de él también un coecualizador de $Ker(\pi)$. Por universalidad del coecualizador existe un homomorfismo $f : X \to Z = W/\leftrightarrow$.

Si hay dos homomorfismos $f, g : X \to Z$, su imagen $Im(\langle f, g\rangle) \subseteq Z \times Z$ es una bisimulación por lema 9.3 punto 7, y por lo tanto contenida en la relación de igualdad sobre Z porque $\zeta : Z \to F(Z)$ es observable por construcción. Así, $(f(x), g(x)) \in Im(\langle f, g\rangle) \subseteq Eq(Z)$ y $f(x) = g(x)$ para toda $x \in X$. Entonces f es único y $\zeta : Z \to F(Z)$ es coálgebra final.$\square$

Parte IV

Consecuencias del Teorema de Existencia de Coálgebra Final.

Usando el teorema de la sección anterior se pueden demostrar otros resultados de mucha importancia para la estructura de la categoría $CoAlg(F)$. En particular, en esta parte del documento se mostrará la importancia de la existencia de una coálgebra final para los siguientes resultados; construcción de coálgebras colibres, completez de la categoría $CoAlg(F)$ y teorema que relaciona la bisimilaridad de estados con el comportamiento.

13. Coálgebras Colibres

Como acercamiento intuitivo a las coálgebras colibres se puede considerar su dual: las álgebras libres. Por ejemplo, un grupo G se dice que es libre si hay un subconjunto S tal que todo elemento de G puede escribirse de forma única como producto finito de elementos de S y sus inversos, de esta manera se trata del mínimo conjunto tal que con las operaciones se puede formar un grupo. Así, estamos formando un grupo desde un conjunto de una manera canónica. Esta construcción no está restringida a los grupos, podemos ampliarla a las álgebras, y a partir de un conjunto construir su álgebra libre asociada por medio de un funtor $Sets \to Alg(F)$. Resulta que este funtor es el adjunto a izquierda del funtor olvido $Alg(F) \to Sets$, y como característica los funtores adjuntos a izquierda realizan construcciones mínimas como lo vimos en el caso de la categoría de los grupos.

Estos resultados tienen su contraparte dual en las coálgebras, por lo tanto se puede construir de una manera natural a partir de un conjunto una coálgebra por medio del adjunto a derecha del funtor olvido $CoAlg(F) \to Sets$, además, esta construcción tiene la característica de ser la máxima posible. El siguiente resultado formaliza la anterior discusión, la idea general se tomó de [2], aunque aquí se completan algunos detalles que hacían falta en la referencia como la preservación de identidades y composiciones, y también el requerimiento de naturalidad.

13.1 Teorema
Para un funtor polinomial Kripke finito $F : Sets \to Sets$, el funtor olvido $U : CoAlg(F) \to Sets$ tiene adjunta a derecha.
Demostración
Dado F se considera el funtor $F' : Sets \to Sets$ dado por $X \mapsto A \times F(X)$. Por lo tanto F' también es polinomial Kripke finito. Entonces por el teorema 12.4 tiene coálgebra final

$$\zeta^A = \langle \zeta_1^A, \zeta_2^A \rangle : \hat{A} \to A \times F(\hat{A})$$

la cual se usará para definir una $F - coálgebra$ con la segunda componente.

$$G(A) \equiv (\hat{A} \xrightarrow{\zeta_2^A} F(\hat{A}))$$

Primero se mostrará que G se puede extender a un funtor de $Sets \to CoAlg(F)$, es decir, que si $f : A \to B$ es una función arbitraria, entonces $G(f)$ es un homomorfismo de coálgebras $G(A) \to G(B)$. Se definirá G sobre f por finalidad como una función $G(f) : \hat{A} \to \hat{B}$, entonces:

$$
\begin{tikzcd}
B \times F(\hat{A}) \arrow[r, "id_B \times F(G(f))"] & B \times F(\hat{B}) \\
A \times F(\hat{A}) \arrow[u, "f \times id_{F(\hat{A})}"] & \\
\hat{A} \arrow[u, "\zeta^A"] \arrow[r, "G(f)"'] & \hat{B} \arrow[uu, "\zeta^B"']
\end{tikzcd}
$$

Por lo tanto $\zeta_2^B \circ G(f) = F(G(f)) \circ \zeta_2^A$, es decir, $G(f)$ es un homomorfismo de coálgebras $G(A) \to G(B)$ como se quería. Pero aún para ser una aplicación funtorial debe preservar identidades y composiciones.

Preserva identidades ya que

$$
\begin{tikzcd}
F(\hat{A}) \arrow[r, "F(G(id_A))"] & F(\hat{A}) \\
\hat{A} \arrow[u, "\zeta_2^A"] \arrow[r, "G(id_A)"'] & \hat{A} \arrow[u, "\zeta_2^A"']
\end{tikzcd}
$$

además, el morfismo $id_{G(A)}$ también hace conmutar el diagrama porque ambas coálgebras son $G(A)$, entonces por finalidad se tiene que ambos morfismos deben ser iguales por lo tanto, $G(id_A) = id_{G(A)}$
De manera análoga se procede para mostrar que preserva composiciones. Sean $f : A \to B$ y $g : B \to C$ dos morfismos (funciones) en $Sets$, entonces $g \circ f : A \to C$, y de acuerdo al siguiente diagrama

$$
\begin{tikzcd}
 & C \times F(\hat{B}) \arrow[r, "id_C \times F(G(g))"] & C \times F(\hat{C}) \\
B \times F(\hat{A}) \arrow[r, "id_B \times F(G(f))"] & B \times F(\hat{B}) \arrow[u, "g \times id_{F(\hat{B})}"'] & \\
A \times F(\hat{A}) \arrow[u, "f \times id_{F(\hat{A})}"] & & \\
\hat{A} \arrow[u, "\zeta^A"] \arrow[r, "G(f)"'] & \hat{B} \arrow[uu, "\zeta^B"'] \arrow[r, "G(g)"'] & \hat{C} \arrow[uuu, "\zeta^C"']
\end{tikzcd}
$$

Se cumple que $\zeta_2^C \circ (G(g) \circ G(f)) = F(G(g)) \circ F(G(f)) \circ \zeta_2^A = F(G(g) \circ G(f)) \circ \zeta_2^A$. Donde la última igualdad se tiene porque F es funtor. Esto nos dice que $G(g) \circ G(f)$ es un homomorfismo de las coálgebras $G(A)$ y $G(C)$, pero naturalmente $G(g \circ f)$ también lo es, entonces por finalidad de ζ^C, estos dos morfismos deben ser iguales, de esta manera $G(g) \circ G(f) = G(g \circ f)$. Hasta ahora se ha mostrado que G es una aplicación funtorial de $Sets \to CoAlg(F)$, a continuación se mostrará que efectivamente es una adjunción, para esto necesitamos probar la biyección y el requerimiento de naturalidad. Sea $c : X \to F(X)$ una coálgebra entonces probaremos que

$$U(c) \xrightarrow{g} A \Leftrightarrow c \xrightarrow{h} G(A)$$

Dada una función $g : X \to A$, se puede definir $\overline{g} : X \to \hat{A}$ por finalidad de acuerdo al siguiente diagrama

$$\begin{tikzcd} A \times F(X) \arrow[r, "id_A \times F(\overline{g})"] & A \times F(\hat{A}) \\ X \arrow[u, "\langle g,c \rangle"] \arrow[r, "\overline{g}"'] & \hat{A} \arrow[u, "\zeta^A"'] \end{tikzcd}$$

se tiene que $\zeta_2^A \circ \overline{g} = F(\overline{g}) \circ c$, entonces $\overline{g}$ es un homomorfismo de coálgebras $c \to G(A)$ como se quería. Dado un homomorfismo de coálgebras $h : c \to G(A)$, entonces la función que le corresponde a los conjuntos es la de sus conjuntos subyacentes, $\overline{h} = \zeta_1^A \circ h : X \to A$.
Para probar biyectividad se mostrará que $\overline{\overline{h}} = h$ y $\overline{\overline{g}} = g$.

$$\overline{\overline{g}} = \zeta_1^A \circ \overline{g} = \pi_1 \circ \zeta^A \circ \overline{g} = \pi_1 \circ (id_A \times F(\overline{g})) \circ \langle g, c \rangle = \pi_1 \langle g, F(\overline{g}) \circ c \rangle = g$$

Por construcción $\overline{\overline{h}}$ es el único homomorfismo tal que $\zeta^A \circ \overline{\overline{h}} = (id_A \times F(\overline{\overline{h}})) \circ \langle \overline{h}, c \rangle$, es decir, $\left\langle \zeta_1^A \circ \overline{\overline{h}}, \zeta_2^A \circ \overline{\overline{h}} \right\rangle = \left\langle \zeta_1^A \circ h, F(\overline{\overline{h}}) \circ c \right\rangle$. Pero h también cumple esta propiedad por ser homomorfismo de coálgebras, entonces por unicidad $\overline{\overline{h}} = h$.

Para mostrar naturalidad sea $f : c \to d$ un homomorfismo entre las coálgebras $c : X \to F(X)$ y $d : Y \to F(Y)$. Sea $m : A \to B$ una función entre conjuntos, además consideremos el morfismo $g : U(d) \to A$. La aplicación $\psi : Sets(U(c), A) \to CoAlg(F)(c, G(A))$ relaciona los homomorfismos de los conjuntos con homomorfismos en coálgebras, la forma canónica es como mostramos anteriormente. Para que ψ cumpla naturalidad se debe cumplir: $G(m) \circ \psi(g) \circ f = \psi(m \circ g \circ U(f))$. El lado izquierdo de la igualdad se puede representar por el siguiente diagrama

$$\begin{tikzcd} F(X) \arrow[r, "F(f)"] & F(Y) \arrow[r, "F(\psi(g))"] & F(\hat{A}) \arrow[r, "F(G(m))"] & F(\hat{B}) \\ X \arrow[u, "c"] \arrow[r, "f"'] & Y \arrow[u, "d"'] \arrow[r, "\psi(g)"'] & \hat{A} \arrow[u, "\zeta_2^A"'] \arrow[r, "G(m)"'] & \hat{B} \arrow[u, "\zeta_2^B"'] \end{tikzcd}$$

donde $G(m) \circ \psi(g) \circ f : c \to G(B)$ es un homomorfismo entre las coálgebras $c : X \to F(X)$ y $G(B) \equiv (\hat{B} \xrightarrow{\zeta_2^B} F(\hat{B}))$.
Por otro lado, $m \circ g \circ U(f)$ actúa de la siguiente manera:

$$X \xrightarrow{U(f)} Y \xrightarrow{g} A \xrightarrow{m} B$$

por lo tanto es un homomorfismo entre los conjuntos $X \to B$. Como ψ relaciona los homomorfismos de $Sets$ con los homomorfismos de $CoAlg(F)$, entonces $\psi(X \to B) = (X \to \hat{B})$ como se vio al inicio de la página, de esta manera, $\psi(m \circ g \circ U(f))$ relaciona las coálgebras $c \to G(B)$, por finalidad se tiene entonces que $G(m) \circ \psi(g) \circ f = \psi(m \circ g \circ U(f))$.$\square$

Encontrar adjunciones como la anterior es una muy buena herramienta ya que nos permite ir de una categoría a otra y regresar, por ejemplo, podemos hacer la composición $\eta : X \to GU(X)$ y lo que resulta es un homomorfismo entre la coálgebra $X \to F(X)$ y la coálgebra colibre $\hat{X} \to F(\hat{X})$ de su conjunto subyacente. También podemos hacer la composición $\varepsilon : UG(Y) \to Y$ y el resultado es una función que transforma el conjunto subyacente de una coálgebra colibre $\hat{A}$ en su conjunto original A.

Estas transformaciones son muy importantes, η se llama unidad y ε se llama counidad. Estas coálgebras colibres nos serán de mucha utilidad a la hora de construir el producto de coálgebras, que es el tema de la siguiente sección.

14. La categoría $CoAlg(F)$ es bicompleta

Como se dijo en la sección 11, la categoría $CoAlg(F)$ es cocompleta, para que sea bicompleta falta además que sea completa. Una categoría se llama cocompleta cuando tiene todos sus colímites y completa cuando tiene todos sus límites. Los conceptos de límite y colímite son de un alto nivel de abstracción y su existencia es una categoría es un resultado muy fuerte para ser relevante. Una característica de los límites es que todos cumplen la propiedad universal, esto quiere decir que si hay otro objeto que cumpla con lo que cumple el límite entonces debe haber un único morfismo que relacione el objeto con el límite. Ejemplos de límites son: Objetos Finales, Productos, Pullback, Ecualizadores (en particular Kernel). Entre los colímites se pueden hallar sus duales: Objetos iniciales, Coproductos, Pushout, Coecualizadores (en particular Cokernel).

Un resultado muy importante en el tema de completez es el teorema de existencia de límites demostrado en [1] Pag 109, este teorema dice que si una categoría tiene ecualizadores para cada par de morfismos y si tiene todos sus productos indexados entonces existen todos los límites. Por lo tanto lo que necesitamos para demostrar que $CoAlg(F)$ es completa para un funtor polinomial, es demostrar que tiene ecualizadores para cada par de morfismos entre coálgebras y también que existe el producto de coálgebras.

Para demostrar este resultado se va a retomar la definición 8.1.2 que introduce el invariante más grande de todos contenido en un predicado P, este invariante se denotado por $\Box P$, se define como la unión de los invariantes contenidos en P. Antes de proceder demostraremos un lema necesario para los resultados posteriores.

14.1 Lema
Sean $c : X \to F(X)$ y $d : Y \to F(Y)$ coálgebras de un funtor polinomial F y sea $P \subseteq X$ un predicado arbitrario. El invariante más grande $\Box P \subseteq X$ tiene una estructura de subcoálgebra asociada, denotada por c_P,

$$\begin{tikzcd} F(\Box P) \arrow[r, "F(m)"] & F(X) \\ \Box P \arrow[u, "c_P"] \arrow[r, "m"'] & X \arrow[u, "c"'] \end{tikzcd}$$

además, cada homomorfismo $f : d \to c$ que satisface que $f(y) \in P$ para todo $y \in Y$, también satisface que $f'(y) \in \Box P$ para toda $y \in Y$. Donde $f' : d \to c_P$ con $m \circ f' = f$.
Demostración
El predicado $\Box P$ tiene una subcoálgebra asociada c_P por ser invariante y por el teorema 7.5. Primero notemos que las imágenes directas de homomorfismos preservan invariantes ya que

$$\begin{aligned}
\coprod_d \coprod_f P &= \coprod_{F(f)} \coprod_c P \\
&\subseteq \coprod_{F(f)} Pred(F)(P) \\
&= Pred(F)(\coprod_f P)
\end{aligned}$$

donde la primera línea es por ser f homomorfismo, la segunda es porque P es invariante y la tercera se puede probar por inducción en la estructura de F. Entonces en particular $Im(f) = \coprod_f(\top)$ es un invariante donde $\top = Y \subseteq Y$. Por lo tanto, como $f(y) \in P$ para todo $y \in Y$, es decir, $Im(f) \subseteq P$ pero como es invariante entonces $Im(f) \subseteq \Box P$. Teniendo esto en cuenta y como $m : \Box P \to X$ es una inclusión y $f' : Y \to \Box P$ entonces se tiene la siguiente factorización $f = m \circ f'$.

Para ver que esta factorización de homomorfismos es correcta basta probar que f' también es un homomorfismo de coálgebras. Para este resultado se va a usar que $F(m)$ es monomorfismo así como lo es m.

$$\begin{aligned}
F(m) \circ c_P \circ f' &= c \circ m \circ f' \\
&= c \circ f \\
&= F(f) \circ d \\
&= F(m \circ f') \circ d \\
&= F(m) \circ F(f') \circ d
\end{aligned}$$

Como $F(m)$ es monomorfismo entonces $c_P \circ f' = F(f') \circ d$, esto quiere decir que f' es un homomorfismo entre las coálgebras d y c_P.

Observación: Que $F(m)$ es monomorfismo, viene del hecho que m es un monomorfismo si y sólo si su diagrama pullback es

$$\begin{tikzcd}
\bullet \arrow[r, "k"] \arrow[d, "k"'] & P \arrow[d, "m"] \\
P \arrow[r, "m"'] & X
\end{tikzcd}$$

Como los funtores polinomiales preservan pullback (demostrable por inducción en la estructura) luego de aplicar F el diagrama pullback es similar, por lo tanto $F(m)$ es un monomorfismo.$\Box$

En la sección 11 se mostró que los coproductos y coecualizadores de coálgebras existen y son sencillos de calcular, ya que se calculan exactamente como se hace en conjuntos, más aún, el conjunto subyacente de estas coálgebras es el mismo que en los conjuntos. El caso es análogo con los productos y ecualizadores en las álgebras.

El caso de los límites para las coálgebras, los productos y ecualizadores no necesariamente tienen que existir, pero cuando existen, su conjunto subyacente es diferente del producto cartesiano de los conjuntos subyacentes. Como veremos, en el caso de los funtores polinomiales el conjunto en cuestión es el invariante más grande.

Tanto los ecualizadores como los productos de coálgebras fueron discutidos por primera vez en [12], luego se demostraron para categorías basadas en conjuntos y para funtores acotados en [13]. La demostración que se hará en este documento se tomó de [14] que a su vez se basó en [13]. El resultado es una prueba elemental respecto a los elementos que hemos introducido, pero técnica en su construcción. Antes de los teoremas se mostrarán las definiciones de lo que queremos mostrar.

14.2 Definición (Ecualizador)
Sea $\mathbb{C}$ una categoría y dos morfismos "paralelos" $f, g : X \to Y$, un ecualizador de f, g es un objeto Q junto con un morfismo $q : Q \to X$ tal que $f \circ q = g \circ q$. Más aún, el par (Q, q) debe ser universal en el sentido que dado cualquier otro par (Q', q') que cumple $f \circ q' = g \circ q'$ existe un único morfismo $u : Q' \to Q$ tal que el siguiente diagrama conmuta:

$$\begin{tikzcd}
Q \arrow[r, "q"] & X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
Q' \arrow[u, "u"] \arrow[ur, "q'"'] & &
\end{tikzcd}$$

14.3 Teorema (Ecualizador de Coálgebras)
Sean $c : X \to F(X)$ y $d : Y \to F(X)$ dos coálgebras con funtor polinomial y con homomorfismos $f, g : X \to Y$ entre ellas, entonces existe un ecualizador en $CoAlg(F)$.

$$\begin{tikzcd}
F(\Box Equal(f,g)) \arrow[r, "F(m)"'] & F(X) \arrow[r, shift left, "F(f)"] \arrow[r, shift right, "F(g)"'] & F(Y) \\
\Box Equal(f,g) \arrow[u] \arrow[r, "m"'] & X \arrow[u, "c"] \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \arrow[u, "d"']
\end{tikzcd}$$

donde $Equal(f, g)$ es el ecualizador en los conjuntos.
Demostración
La estructura de coálgebra $d : \Box Equal(f, g) \to F(\Box Equal(f, g))$ existe por el lema 14.1, además con el embebimiento canónico m se tiene que $f \circ m = g \circ m$ ya que $\Box Equal(f, g) \subseteq Equal(f, g)$.
Por lo tanto se mostrará que cumple la propiedad universal. Para cada coálgebra $e : Z \to F(Z)$ con homomorfismo $h : Z \to X$ que cumpla la propiedad de ecualizador es decir, $f \circ h = g \circ h$ se tiene que $h(z) \in Equal(f, g)$ para todo $z \in Z$ ya que es la propiedad sobre los conjuntos, pero por el lema 14.1 $h'(z) \in \Box Equal(f, g)$ para todo $z \in Z$. Esto quiere decir que hay un único homomorfismo de coálgebras $Z \to \Box Equal(f, g)$.$\Box$

Como vemos en la demostración el paso importante es que $\Box Equal(f, g)$ tiene estructura de coálgebra, por lo tanto esta es la que se emplea como candidata. Esta idea se usará también en la siguiente demostración, donde la candidata a coálgebra producto es el máximo invariante contenido en el ecualizador de un producto de conjuntos, como lo veremos a continuación. En la referencia [14] el teorema se hace para el producto de dos coálgebras, el autor lo generalizó para un productos arbitrarios.

14.4 Definición (Producto)
Sea $\mathbb{C}$ una categoría. El producto de dos objetos $X, Y \in \mathbb{C}$ es un nuevo objeto $X \times Y \in \mathbb{C}$ con dos morfismos proyecciones π_1, π_2 que son universales; para cada par de aplicaciones $f : Z \to X$, y $g : Z \to Y$ en $\mathbb{C}$ existe un único morfismo $\langle f, g \rangle : Z \to X \times Y$ en $\mathbb{C}$, tal que el siguiente diagrama conmuta

$$\begin{tikzcd}
X & X \times Y \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y \\
& Z \arrow[ul, "f"] \arrow[u, "\langle f,g\rangle"'] \arrow[ur, "g"'] &
\end{tikzcd}$$

14.5 Teorema (Producto de Coálgebras)
Para un funtor polinomial $F : Sets \to Sets$, la categoría $CoAlg(F)$ tiene productos arbitrarios.
Demostración
En este teorema se construirá el producto de una familia de coálgebras $c_i : X_i \to F(X_i)$. Lo primero que se hará será construir el objeto producto, luego definir las proyecciones y por último probar la propiedad universal, es decir, que si hay otro objeto con las mismas características del producto, entonces debe haber un único morfismo de ese objeto al producto.

Lo natural sería empezar por el producto en conjuntos y formar una coálgebra a partir de este conjunto, anteriormente vimos cómo hacerlo y es por medio de la coálgebra colibre. A partir del producto de los subconjuntos subyacentes $\prod_i X_i$ se puede formar su coálgebra colibre asociada como se indica en el teorema 13.1. Como $U_F G(\prod_i X_i)$ es el conjunto subyacente de la coálgebra colibre, esta se puede expresar $e : U_F G(\prod_i X_i) \to F(U_F G(\prod_i X_i))$. Recordemos que a partir de $U_F G(\prod_i X_i)$ podemos obtener nuevamente $\prod_i X_i$ por medio de la counidad, $\varepsilon : U_F G(\prod_i X_i) \to \prod_i X_i$. También podemos expresar la counidad por medio de sus componentes pasándola por la proyección adecuada, es decir, $\varepsilon_i = \pi_i \circ \varepsilon : U_F G(\prod_i X_i) \to X_i$. A continuación formamos el siguiente ecualizador en $Sets$, este es un primer acercamiento para obtener una candidata de coálgebra producto pero sin estructura.

$$E = \left\{ u \in U_F G(\prod_i X_i) \mid (c_i \circ \varepsilon_i)(u) = (F(\varepsilon_i) \circ e)(u) \right\}$$

La idea de formar este ecualizador es para seleccionar los elementos del producto que permitan que haya un homomorfismo entre la coálgebra colibre y cada una de las coálgebras de la familia, en cierta forma se está intentando que las proyecciones de la counidad sean homomorfismos.

$$\begin{tikzcd}
F(U_F G(\prod_i X_i)) \arrow[r, "F(\varepsilon_i)"] & F(X_i) \\
U_F G(\prod_i X_i) \arrow[u, "e"] \arrow[r, "\varepsilon_i"'] & X_i \arrow[u, "c_i"']
\end{tikzcd}$$

E es un buen candidato para coálgebra producto, el único inconveniente es que no tiene estructura, para esto usamos el lema 14.1, a esta estructura la denotaremos sugestivamente como
$\bigodot_i c_i : \Box E \to F(\Box E)$. Hasta el momento los anteriores elementos los podemos relacionar por medio del siguiente diagrama

$$\begin{tikzcd}
& & & F(U_F G(\prod_i X_i)) \arrow[dr, "\langle F(\varepsilon_1), F(\varepsilon_2), \ldots\rangle"] & \\
\Box E \arrow[r, "n"] & E \arrow[r, "m"] & U_F G(\prod_i X_i) \arrow[ur, "e"] \arrow[dr, "\varepsilon"'] & & \prod_i F(X_i) \\
& & & \prod_i X_i \arrow[ur, "\bigotimes_i c_i"'] &
\end{tikzcd}$$

Además, como se dijo anteriormente $\bigodot_i c_i$ es la candidata a coálgebra producto y está relacionada mediante el siguiente diagrama con la coálgebra e

$$\begin{tikzcd} F(\Box E) \arrow[r, "F(m \circ n)"] & F(U_F G(\prod_i X_i)) \\ \Box E \arrow[u, "\bigodot_i c_i"] \arrow[r, "m \circ n"'] & U_F G(\prod_i X_i) \arrow[u, "e"'] \end{tikzcd}$$

a continuación se hará la descripción categórica del producto, para ello se definirán las proyecciones como: $p_i \equiv \varepsilon_i \circ m \circ n{:}\Box E \to X_i$, y se mostrará que son homomorfismos entre las coálgebras $\bigodot_i c_i$ y c_i

$$\begin{aligned} F(p_i) \circ \bigodot_i c_i &= F(\varepsilon_i) \circ F(m \circ n) \circ \bigodot_i c_i \\ &= F(\varepsilon_i) \circ e \circ m \circ n \\ &= \pi_i \circ \bigotimes_i c_i \circ \varepsilon \circ m \circ n \\ &= c_i \circ \pi_i \circ \varepsilon \circ m \circ n \\ &= c_i \circ p_i \end{aligned}$$

La primera igualdad es porque F es un funtor, la segunda es porque $m \circ n$ es homomorfismo (Ver anterior diagrama), la tercera es porque m es ecualizador, la cuarta es por propiedad de las proyecciones y la quinta es la definición de p_i.

Con el objeto construido ahora tenemos que probar su propiedad universal, para ello vamos a suponer que hay otra coálgebra con homomorfismos de ella a las coálgebras c_i, esto es,
$f_i : (d : Y \to F(Y)) \to (c_i : X_i \to F(X_i))$. Así que se debe construir un único morfismo
$d \to \bigodot_i c_i$, podemos empezar por el morfismo correspondiente en conjuntos, es decir
$\langle f_1, f_2, \ldots \rangle : Y \to \bigotimes_i X_i$. Por ser $U_F G(\prod_i X_i)$ el conjunto de una coálgebra libre existe un única función $g : Y \to U_F G(\prod_i X_i)$ que forma un homomorfismo de coálgebras $d \to e$, con $\varepsilon \circ g = \langle f_1, f_2, \ldots \rangle$. Y de acuerdo al siguiente cálculo se concluye que g también ecualiza las mismas funciones que m.

$$\begin{aligned} \langle F(\varepsilon_1), F(\varepsilon_2), \ldots \rangle \circ e \circ g &= \langle F(\pi_1 \circ \varepsilon), F(\pi_2 \circ \varepsilon), \ldots \rangle \circ F(g) \circ d \\ &= \langle F(f_1) \circ d, F(f_2) \circ d, \ldots \rangle \\ &= \langle c_1 \circ f_1, c_2 \circ f_2, \ldots \rangle \\ &= \langle c_1 \circ \pi_1 \circ \varepsilon \circ g, c_2 \circ \pi_2 \circ \varepsilon \circ g, \ldots \rangle \\ &= \bigotimes_i c_i \circ \varepsilon \circ g \end{aligned}$$

Donde la primera igualdad es porque $g : d \to e$ es homomorfismo, la segunda por $\varepsilon \circ g = \langle f_1, f_2, \ldots \rangle$ y porque F es funtor, la tercera es porque f_i es homomorfismo, y la cuarta y la quinta es otra vez por $\varepsilon \circ g = \langle f_1, f_2, \ldots \rangle$.

Por universalidad del ecualizador en $Sets$ se tiene que debe existir un único g' tal que $g = m \circ g'$. Esto quiere decir que $g(y) \in E$ para todo $y \in Y$, entonces por el lema 14.1 se tiene que $g'(y) \in \Box E$ para todo $y \in Y$. Como $g' : Y \to E$ entonces debe existir una única aplicación $\langle\langle f_1, f_2, \ldots \rangle\rangle$ que hace la labor de inclusión tal que $n \circ \langle\langle f_1, f_2, \ldots \rangle\rangle = g'$.

Este homomorfismo $\langle\langle f_1, f_2, \ldots \rangle\rangle : d \to \bigodot_i c_i$ es el candidato al único homomorfismo que relaciona las dos coálgebras producto. Probemos que satisface la propiedad $p_i \circ \langle\langle f_1, f_2, \ldots \rangle\rangle = f_i$ y además que es el único que la satisface.

$$\begin{aligned} p_i \circ \langle\langle f_1, f_2, \ldots \rangle\rangle &= \pi_i \circ \varepsilon \circ m \circ n \circ \langle\langle f_1, f_2, \ldots \rangle\rangle \\ &= \pi_i \circ \varepsilon \circ m \circ g' \\ &= \pi_i \circ \varepsilon \circ g \\ &= \pi_i \circ \langle f_1, f_2, \ldots \rangle \\ &= f_i \end{aligned}$$

La primera línea es la definición de p_i, la segunda porque $n \circ \langle\langle f_1, f_2, \ldots \rangle\rangle = g'$, la tercera porque $g = m \circ g'$, la cuarta porque $\varepsilon \circ g = \langle f_1, f_2, \ldots \rangle$, y la quinta es una proyección.

Además, es el único homomorfismo que lo cumple porque si $h : d \to \bigodot_i c_i$ es otro homomorfismo que cumple que $p_i \circ h = f_i$ entonces $m \circ n \circ h$ es una aplicación de $d \to e$ que cumple

$$\begin{aligned} \varepsilon \circ m \circ n \circ h &= \langle \pi_1 \circ \varepsilon \circ m \circ n \circ h, \pi_2 \circ \varepsilon \circ m \circ n \circ h \rangle \\ &= \langle p_1 \circ h, p_2 \circ h, \ldots \rangle \\ &= \langle f_1, f_2, \ldots \rangle \end{aligned}$$

Donde la segunda línea es por la definición de p_i. Como $\varepsilon \circ g = \langle f_1, f_2, \ldots \rangle$ entonces $m \circ n \circ h = g$, por lo tanto $m \circ n \circ h = m \circ g'$ y como m es mono entonces $n \circ h = g'$ y como $n \circ \langle\langle f_1, f_2, \ldots \rangle\rangle = g'$ se tiene que $h = \langle\langle f_1, f_2, \ldots \rangle\rangle$.$\square$

Esta teorema junto con el teorema 14.3 y el teorema 11.3 sirven para deducir el siguiente corolario importante.

14.4 Corolario
La categoría $CoAlg(F)$ para un funtor polinomial Kripke finito es bicompleta.

15. Principio de prueba coinductiva

A lo largo del documento se buscó la coálgebra final como un lugar para ver reflejados los comportamientos de los estados de una coálgebra. Así, en el ejemplo 2.1 vimos que cualquier número perteneciente al intervalo $[0, 1)$ podíamos encontrarle una representación en sistema decimal por medio de la función comportamiento. De esta manera si vemos los estados de la coálgebra final estos son exactamente comportamientos. Por otro lado, también se vio que dos estados eran bisimilares si ellos pertenecían a una bisimulación, y en el ejemplos posteriores a la definición 5.2, se mostró que en el caso de los autómatas estar en un bisimulación implicaba que las observaciones eran iguales, es decir, eran indistinguibles observacionalmente. Por lo tanto tomando en cuenta estos dos aspectos, el siguiente resultado es muy intuitivo. Este dice que si dos estados son bisimilares entonces tienen el mismo comportamiento, en otras palabras se vuelven iguales cuando se miran en la coálgebra final. Con esto finaliza la caracterización de comportamiento de los estados de una máquina, lo único que hay que hacer para asegurar que dos estados se comporten igual es encontrar una bisimulación que los contenga, aunque por lo general no sea sencillo encontrar esa bisimulación. Este es un resultado fundamental demostrado en la página 42 teorema 9.3 en [11].

15.1 Teorema
Sea $F : Sets \to Sets$ un funtor polinomial finito con coálgebra final $\zeta : Z \to F(Z)$ (Teorema 12.4). Sean $c : X \to F(X)$ y $d : Y \to F(Y)$ dos coálgebras con homomorfismos $beh_c : X \to Z$ y $beh_d : Y \to Z$

dados por finalidad. Entonces dos estados $x \in X$ y $y \in Y$ son bisimilares si y sólo si tienen el mismo comportamiento, es decir

$$x \leftrightarrow y \Longleftrightarrow beh_c(x) = beh_d(y)$$

Demostración
Supongamos que dos estados tienen el mismo comportamiento, entonces por el punto 4 del lema 9.3 se tiene que la relación pullback $Pb(beh_c, beh_d) = \{(x, y) \mid beh_c(x) = beh_d(y)\}$ es una bisimulación, por lo tanto está incluida en la bisimilaridad que es la bisimulación más grande y los dos estados están relacionados con bisimilaridad.

Ahora supongamos que dos estados son bisimilares, entonces ellos pertenecen a la relación de bisimilaridad que es una bisimulación. Esta bisimulación por el teorema 6.4 es una bisimulación de Aczel-Mendler, por lo tanto tiene una estructura de coálgebra $e : (\leftrightarrow) \to F(\leftrightarrow)$ junto con dos homomorfismos $\langle r_1, r_2 \rangle : (\leftrightarrow) \to X \times Y$. Por lo tanto existen dos morfismos $beh_c \circ r_1$ y $beh_d \circ r_2$ entre las coálgebras $(\leftrightarrow)$ y ζ. Por finalidad se tiene que $beh_c \circ r_1 = beh_d \circ r_2$, de esta manera los dos estados tienen el mismo comportamiento.$\square$

Como corolario tenemos el más conocido Principio de Prueba Coinductiva, es el mismo resultado pero expresado de una manera más práctica.

15.2 Corolario (Principio de prueba coinductiva)
Dos estados tienen el mismo comportamiento si y sólo si existe una bisimulación que los contiene.

Si consideramos en vez de las coálgebras c y d, la coálgebra ζ. El resultado en particular dice que en la coálgebra final, la bisimilaridad es la igualdad, como consecuencia cualquier bisimulación sobre la coálgebra final está contenida en la relación igualdad. El resultado es importante ya que como se sabe en las álgebras la igualdad es la clase de equivalencia más fina y por lo tanto está contenida en cualquier congruencia, aquí como resultado dual se tiene que cualquier bisimulación en la coálgebra final está contenida en la igualdad.

Conclusiones

- La importancia de la existencia de la coálgebra final se ve reflejada en el último capítulo, ésta por definición tienen muy buenas propiedades como la existencia del único homomorfismo y que ella misma es un isomorfismo. Además, permite que $CoAlg(F)$ tenga muy buenas propiedades como su completez, y la construcción de coálgebras libres.
- El estudio de los funtores polinomiales, definidos en la categoría de los conjuntos, a pesar de ser una clase restringida, contiene ejemplos muy interesantes como lo son los autómatas deterministas y no deterministas, sucesiones, árboles n-arios.
- El concepto de bisimilaridad y bisimulación capturan muy bien la igualdad de comportamientos, prueba de ello es el teorema 15.2.
- El enfoque de la bisimulación a partir del levantamiento de relación ayuda a percibir este concepto desde un punto de vista computacional, como una relación que se mantiene en cada transición que se hace. A diferencia de la bisimulación por lazos, donde el enfoque categórico hace perder de vista el enfoque computacional, pero es útil en algunos resultados teóricos como en el teorema 15.2 y la estructura de la categoría.
- La lógica temporal es una herramienta que permite transportarse hacia el futuro y el pasado en los diferentes estados de una coálgebra, ayudando en el análisis de propiedades generales con predicados muy puntuales.
- Es interesante notar que construcciones elementales como los invariantes y el cociente tienen una estructura de coálgebra, esto es de suma importancia al momento de poder construir coálgebras finales, productos y ecualizadores.

Bibliografía

.

[1] S. Mac Lane. Categories for the Working Mathematician. 1971

[2] B. Jacobs. Introduction to Coalgebra. Towards Mathematics of States and Observation, 2005

[3] Y. Venema. Algebras and Coalgebras, 2005

[4] Aczel, Mendler. A final coalgebra theorem, 1989

[5] P. Aczel. Non-well-founded sets , volume 14 of CSLI Lecture Notes. CSLI , 1988.

[6] Robin Milner. Communicating Systems and the Pi calculus

[7] M. Cederbaum , M. Roggenbach. On two diferent characterizations of bisimulation, 1996.

[8] http: //plato.stanford.edu/entries/logic-temporal/

[9] P. T. Johnstone. Stone Spaces, Cambridge Univ. Press, 1982

[10] F. Borceux. Handbook of Categorical Algebra 1 : Basic Category Theory, 1994

[11] J.J.M.M. Rutten. Universal coalgebra: a theory of systems, 2000

[12] J. Worrell. Toposes of coalgebras and hidden algebras. Coalgebraic Methods in Computer Science, number 11 , 1998

[13] H. P. Gumm, T. Schröder. Products of coalgebras, 2001

[14] Bart Jacobs, Comprehension for Coalgebras, 2002

[15] Davide Sangiorgi, On the origins of bisimulation, coinduction, and xed points, 2007