UN ALGORITMO PARA LA APLICACIÓN DE INGENIERIA REVERSA EN SISTEMAS BIOLÓGICOS

Por:
María del Pilar Orjuela Garavito

Tesis sometida en cumplimiento parcial de los requerimientos para el grado de

MAESTRÍA EN CIENCIAS
en
MATEMÁTICA APLICADA

Universidad de Puerto Rico
Recinto Universitario de Mayagüez

Mayo de 2009

Aprobado por:

Omar Colón-Reyes, Ph.D Presidente, Comité Graduado	Fecha
Dorothy Bollmam, Ph.D Miembro, Comité Graduado	Fecha
Luis Fernando Cáceres, Ph.D Miembro, Comité Graduado	Fecha
Jeannette Santos, Ph.D Rerpresentante Estudios Graduados	Fecha
Julio Quintana, Ph.D Director Departamento de Ciencias Matemáticas	Fecha

Resumen de Disertación Presentado a Escuela Graduada de la
Universidad de Puerto Rico como requisito parcial de los
Rquerimientos para el grado de Mestría en Ciencias

UN ALGORITMO PARA LA APLICACIÓN DE INGENIERIA REVERSA EN SISTEMAS BIOLÓGICOS

Por

María del Pilar Orjuela Garavito

Mayo de 2009

Consejero: Omar Colón Reyes

Departamento: Departamento de Ciencias Matemáticas

RESUMEN

Dado un conjunto de datos sobre un cuerpo finito F_q, queremos encontrar una función que interpole tales datos. Esta función se puede obtener por ingeniería reversa y es de la forma

$$f = (f_1, f_2, ..., f_n) : F_q^n \longrightarrow F_q^n$$

donde F_q^n es el n-producto cartesiano del cuerpo finito con q elementos y $f_i \in F_q[x_1, \ldots, x_n]$. Pero la pregunta que nos hacemos es, ¿podemos escribir f_i en términos de x_j?. Usando una versión del Algoritmo de Sasao desarrollado por D. Bollman y E. Orozco podemos producir una conjunto $\mathcal{X}$ con variables no redundantes. Luego, utilizando $\mathcal{X}$ y una versión del algoritmo del Teorema Chino de los residuos, una fórmula alternativa de interpolación presentada en [11], se puede encontrar una solución particular f_0 que se ajusta a la datos en términos de x_j y las variables de $\mathcal{X}$. Por último, se computa el ideal I de todas las soluciones particulares que se desvanecen en los datos y usando teoría de eliminación se obtendrá la reducción f de f_0 con respecto a $I \cap F_q[\mathcal{X}, x_j]$.

Abstract of Dissertation to the Graduate School of the
University of Puerto Rico in Partial Fulfillment of the
Requirements for the Degree of Master of Science

UN ALGORITMO PARA LA APLICACIÓN DE INGENIERIA REVERSA EN SISTEMAS BIOLÓGICOS

By

María del Pilar Orjuela Garavito

May 2009

Chair: Omar Colón Reyes

Major Departament: Department Of Mathematical Sciences

ABSTRAC

Given a set of data over a finite field F_q, we are interested in finding a function that interpolates this data. This function can be obtained using reverse engineering. Such a function is of the form: $f = (f_1, f_2, ..., f_n) : F_n^q \longrightarrow F_n^q$, where F_n^q is the n-fold Cartesian product of a finite field with q elements, and $f_i \in F_q[x_1, \ldots, x_n]$. In applying these methods, a key question arises: Can we write f_i in terms of x_j?

Using a version of Sasao's Algorithm developed by D. Bollman and E. Orozco we can produce a minimal basis $\mathcal{X}$ with no redundant variables. Using $\mathcal{X}$ and The Chinese Remainder Algorithm, an alternative to the interpolation formula presented in [11], we find a particular solution f_0 that interpolates the data in terms of x_j and the variables of $\mathcal{X}$. Later, we compute the ideal I of all solutions that vanish on the data, and, using elimination theory, we obtain the reduction f of f_0 with respect to $I \cap F_q[\mathcal{X}, x_j]$.

A mis padres y esposo que me motivan a seguir cada día.

Agradecimientos

Quiero agradecer primero a Dios por darme la oportunidad de estar aquí hasta donde he llegado.

A mis padresy hermanos por estar apoyándome y hacerme sentir como si estuviera en casa.

A mi esposo Jaime, por ser mi guia y darme fuerza para continuar con mis estudios.

A mi consejero, Omar Colón Reyes por su paciencia, disposición, dedicación y porque sin él este trabajo no hubiera sido posible.

A todos los profesores y personal administrativo del Departamento de Matemáticas, en especial a Nilsa y Luis Fernando por brindarme su ayuda y su amistad.

A mis amigos Cesar, Leo y Edwin quienes fueron mi familia durante el año que compartimos juntos y por los bueno momentos que pasamos.

A todos mis compañeros de la maestría, al personal de AFAMaC y todas aquellas personas que tuvieron que ver estos dos años conmigo y se olvidó nombrar.

A todos ellos, Muchas gracias!!!

Lista de Algoritmos

Índice general

Capítulo 1

Introducción

Las redes reguladoras de genes son aquellas que realizan el procesamiento de la información y mantienen el mecanismo de control en las células. Su función consiste en regular los genes que codifican las proteínas para diversas funciones sobre el organismo, creando una red compleja de interacciones. La inferencia y simulación de estas redes contribuye sustancialmente al conocimiento biológico como también en aplicaciones prácticas en el ámbito de los circuitos electrónicos, redes neuronales artificiales y otras áreas de la matemática e ingeniería.

La ingeniería reversa, también conocida como el problema inverso, propone inferir o encontrar el modelo de un sistema a partir de datos experimentales. En este trabajo, se producirá un modelo dinámico de una red reguladora de genes, el cual puede ser usado para simular y predecir las diferentes respuestas de los genes que se puedan generar. Las mediciones a ser usadas son expresiones de los genes que resultan, por ejemplo, de perturbaciones a la red a través del tiempo.

Para este trabajo, el modelo a encontrar será un Sistema Dinámico Polinomial (PDS)

multivariado sobre un cuerpo finito $\mathbb{F}$. Este va a ser de la forma

$$f = (f_1, f_2, \ldots, f_n) : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$$

donde $f_i \in \mathbb{F}[x_1, x_2, \ldots, x_n]$, n representa el número de genes que interfieren en la red, x_i es el gen i y q es el número de elementos de $\mathbb{F}$. Éstos elementos representan los estados o expresiones posibles de los genes, tal que si $s_0, s_1, \ldots, s_{m-1}$ son las m mediciones consecutivas de la red en m tiempos, entonces

$$f(s_0) = s_1, \quad f(s_1) = s_2, \ \ldots, f(s_{m-2}) = s_{m-1}$$

Ahora, supongamos que además de las mediciones que se obtuvieron del sistema, se posee un conocimiento previo que identifica la relación de un gen con otro, esto es, un gen i influye en el siguiente estado del gen j, entonces la pregunta que nos hacemos es ¿cómo escribir f_i en términos de x_j? Trataremos de resolver ésta utilizando el resultado de [11].

Dividiremos este trabajo en 5 capítulos. En el capítulo 1 se encuentra una breve introducción al problema por resolver. En el capítulo 2 se describe en forma general el porqué de la ingeniería reversa, su aplicación en los sistemas biológicos y en especial en las redes reguladoras de genes y por último se explica lo que es ingeniería reversa de sistemas polinomiales sobre cuerpos finitos.

En el capítulo 3 se dan algunos conceptos importantes para hacer álgebra computacional, entre los cuales se encuentran ideales, órdenes monomiales, reducción de polinomios, algoritmos de división en múltiples variables, bases de Gröbner para ideales, entre otros, incluyendo los algoritmos para encontrar tales conceptos. Por último, se explica el algoritmo desarrollado por Lubenbacher y Stigler en [9] para hacer ingeniería reversa de redes reguladoras de genes utilizando lo mencionado anteriormente.

La definición de base y bases mínimas para un polinomio multivariado f y el método desarrollado por Sasao y luego trabajado por Bollman y Orozco se explican en el capítulo 4.

Finalmente, en el capítulo 5 se dan las soluciones al problema planteado en este trabajo, haciendo uso de todo lo desarrollado en los capítulos anteriores. Junto a esto, se explican los algoritmos que fueron implementados en el programa computacional CoCoA que ayudan a simplificar los cálculos. Al final se dan algunas conclusiones y sugerencias para trabajos futuros.

Capítulo 2

Ingeniería reversa

Siendo la ingeniería el arte de aplicar todo conocimiento científico para cubrir las necesidades humanas, surge la ingeniería reversa como una de sus disciplinas cuyo propósito es descubrir o generar el diseño de un sistema a través del análisis de su estructura, su funcionamiento y operación. Su nombre viene del trabajo que hay que realizar para obtener lo que se deseea, pues funciona de manera inversa al trabajo normal de la ingeniería.

En muchos campos se utiliza la ingeniería reversa a nivel de dispositivos mecánicos, circuitos integrados, aplicaciones militares y en especial, en la ingeniería de software, donde para descubrir cómo funciona un programa, función o característica de cuyo código fuente no se dispone, se aplica ésta hasta el punto de poder modificar ese código o generar código propio que cumpla las mismas funciones.

Nuestro objetivo es encontrar un nuevo modelo para realizar ingeniería reversa a sistemas biológicos.

2.1. Ingeniería reversa en sistemas biológicos

Algunas técnicas de la matemática, estadística, ingeniería y ciencias de la computación han sido desarrolladas e implementadas para hacer ingeniería reversa a sistemas biológicos. La importancia de esto radica no sólo en la investigación sobre las propiedades dinámicas de los aspectos específicos de los organismos que pueden ser aplicado ampliamente, sino en crear modelos de precisión de las células y órganos, y descubrir los principios fundamentales, estructurales, ambientales y evolutivos detrás de los sistemas biológicos que definen el diseño del espacio posible de la vida.

Uno de los problemas importantes en sistemas biológicos son las redes bioquímicas y en particular, las redes reguladoras de genes o GRN (por sus siglas en inglés). Éstas son colecciones de segmentos de ADN que interactúan unas con otras y con sustancias de las células para realizar proteínas específicas que tienen propiedades estructurales, enzimáticas (que catalizan reacciones), o que sirven para activar otros genes que al encenderlos, inician la producción de nuevas proteínas.

Aplicar ingeniería reversa en las redes reguladoras de genes busca describir y modelar las relaciones existentes entre sus componentes (o topología de la red) expresándolas mediante grafos dirigidos, y encontrar las leyes que rigen su dinámica. Algunos autores como Friedman (ver [6]) y Hartemink (ver [8]) han propuesto reconstruir la topología de las redes, en particular, las bayesianas, que luego, éste último y Filkov (ver [5]) utilizaron para analizar y obtener el modelo de la red reguladora responsable para el control de los genes necesarios para el metabolismo de la lactosa.

Yeung en [13] se aproxima al modelo de la dinámica de las redes bioquímicas (en particular, del RNA y las proteínas), generando lo que ellos llaman "la primera versión"de la topología de la red que servirá de base para su futuro análisis.

Los métodos para realizar ingeniería reversa en sistemas biológicos pueden ser clasificados, en general, en métodos de tiempo discretos o continuos y determinísticos o estocásticos. Los modelos determinísticos se basan en el concepto en el que el comportamiento futuro se puede predecir a partir del comportamiento de los últimos tiempos ignorando la existencia de perturbaciones externas que pueden alterar los datos del futuro, mientras que en los estocásticos se tienen en cuenta otros factores exteriores que influyen sobre el sistema.

Conceptualmente, cualquier reconstrucción de un sistema a partir de ingeniería reversa, tiene 3 entidades que influyen en su procedimiento: (1) Un conjunto de datos que surgen de las mediciones del sistema. (2) Un modelo matemático con el cual vamos a simular el sistema, como por ejemplo, ecuaciones diferenciales o redes bayesianas, y un espacio de estados que sean consistentes con los datos. (3) Un método de búsqueda para escoger el modelo con mayor probabilidad que genere las mediciones dadas y que se ajuste posiblemente a un conocimiento previo. Para este trabajo, el modelo matemático a considerar es sistemas dinámicos polinomiales sobre cuerpos finitos.

2.2. Ingeniería reversa de PDS sobre cuerpos finitos

Un sistema dinámico es un sistema que presenta un cambio o evolución de su estado como función del tiempo. El tiempo puede medirse en forma discreta o en forma continua. En los sistemas dinámicos en tiempo discreto el tiempo se mide en pequeños lapsos y son modelados como relaciones recursivas. Por ejemplo, la ecuación logística

$$x_{t+1} = ax_t(1 - x_t)$$

es un modelo de sistema dinámico donde t representa el tiempo y x es la variable que cambia con respecto al tiempo.

Los sistemas dinámicos en tiempo continuo son expresados mediante ecuaciones diferenciales. Por ejemplo, el sistema linear de ecuaciones diferenciales ordinarias

$$\frac{dx_i}{dt} = -\lambda_i x_i(t) + \sum_{j=1}^{N} w_{ij}(t) + b_i(t) + \xi_i(t), \quad i = 1, \ldots, N$$

donde x_i son las concentraciones de RNA, λ_i son las razones de degradación, b_i los estímulos externos, ξ_i el ruido y las variables w_{ij} describe la fuerza y tipo de influencia que ejerce el gen j sobre i, es el sistema dinámico en tiempo continuo planteado por Yeung en [13].

Para nuestro propósito se considerarán sistemas dinámicos en tiempo discreto y a continuación definiremos formalmente lo que es un sistema dinámico finito.

Definición 2.2.1. *Un sistema dinámico finito (FDS) f es una función*

$$f : X \longrightarrow X$$

donde X es un conjunto finito.

Ejemplo 2.2.2. Sea $X = \{0, 1, 2, 3, 4\}$ y sea $f : X \longrightarrow X$ definida por $f(x) = (x + 1) mod\ 5$. Entonces f es un FDS.

Ejemplo 2.2.3. Un sistema dinámico finito booleano es una función $g : X \longrightarrow X$ donde $X = \{0, 1\}$, es decir, donde el conjunto de estados posee sólo 2 elementos.

El ejemplo anterior es natural generalizarlo a cuerpos finitos, es decir, un FDS $f : \mathbb{X} \longrightarrow \mathbb{X}$ donde $\mathbb{X}$ es un cuerpo finito. Aquí existen 2 tipos de modelos a considerar: los modelos univariados y los multivariados.

Definición 2.2.4. *Un FDS univariado es un modelo de la forma $f : \mathbb{F}_q \longrightarrow \mathbb{F}_q$ donde $\mathbb{F}_q$ es un cuerpo con q elementos, mientras que un multivariado es de la forma*

$f : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$. *En esta última,* f *puede escribirse en términos de funciones coordenadas* $f_i : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q$ *para* $i = 1, 2, \ldots, n$ *denominadas funciones de transición, esto es, si* $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in X^n$ *entonces* $f(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_n(\boldsymbol{x}))$.

En las redes reguladoras de genes, una red que consista de n genes es representada por el FDS $f : X^n \longrightarrow X^n$, en el cual, cada gen i es modelado por la variable x_i y X es el conjunto finito de estados posibles que pueden adquirir los genes. El siguiente ejemplo muestra una clara aplicación de ésto.

Ejemplo 2.2.5. Las redes booleanas pueden ser vistas como sistemas dinámicos finitos de la forma $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$ donde $\mathbb{F}_2$ es un cuerpo finito de dos elementos y $n \geq 1$. Estos dos elementos, denotados comúnmente $0/1$, pueden representar la inhibición(0) o activación(1) de un gen, el nivel de concentración (bajo(0) o alto (1)) de un compuesto químico, entre otras.

Una de las desventajas de modelar una red en forma booleana es la necesidad de discretizar los valores en 2 estados cuantitativos, el cual puede generar la pérdida de una gran cantidad de información valiosa para el análisis de la dinámica de la red. En respuesta a esto, han surgido los modelos multiestados y los modelos híbridos, que mejor muestran las características de las redes reguladoras.

El modelo multivariado se utiliza para obtener información de cómo un gen es afectado por otros genes en la red mientras que el univariado da información acerca del comportamiento de la red en general. Sin embargo, estos dos tipos de modelos, pueden ser considerados equivalentes en el sentido que si un sistema dinámico $f : \mathbb{X} \longrightarrow \mathbb{X}$ es equivalente a $g : \mathbb{Y} \longrightarrow \mathbb{Y}$ si existe un epimorfismo $\Phi : \mathbb{X} \longrightarrow \mathbb{Y}$ tal que $\Phi \circ f = g \circ \Phi$. Para profundizar más, ver [1].

Definición 2.2.6. *Un sistema dinámico finito polinomial (PDS)* $f : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ *es un caso especial de los sitemas dinámicos finitos donde cada función de transición* f_i *es un polinomio.*

Así cada función coordenada f_i está en el anillo cociente

$$R = \mathbb{F}[x_1, \ldots x_n] / \langle x_1^q - x_1, \ldots, x_n^q - x_n \rangle$$

Como resultado, ellos son polinomios en n variables con coeficientes en $\mathbb{F}$ y el grado de cada variable es a lo mas $q - 1$.

Ejemplo 2.2.7. Sea $f = (x_1^2 x_2, x_3 - x_2 + 2, x_1^2 x_3^2, x_2 + 2x_3 - 2) : \mathbb{Z}_3^4 \longrightarrow \mathbb{Z}_3^4$. Entonces las funciones coordenadas son: $f_1(x_1, x_2, x_3, x_4) = x_1^2 x_2$, $f_2(x_1, x_2, x_3, x_4) = x_3 - x_2 + 2$, $f_3(x_1, x_2, x_3, x_4) = x_1^2 x_3^2$, $f_4(x_1, x_2, x_3, x_4) = x_2 + 2x_3 - 2$

Ejemplo 2.2.8. En un PDS booleano las funciones coordenadas son monomios libres de cuadrados, en donde la multiplicación corresponde al "Y" lógico, la suma al lógico "Ó" y la negación es la suma del término constante 1, esto es:

$$x \wedge y = x \cdot y$$
$$x \vee y = x + y + x \cdot y$$
$$\neg x = x + 1$$

Existen varios tipos especiales de funciones multivariadas que son muy apetecidas para analizar la dinámica del sistema. Veamos algunos:

Definición 2.2.9. *Los PDS lineales, como dice el nombre, son aquellos en donde las funciones de transición son polinomios lineales.*

Ejemplo 2.2.10. Si $f : \mathbb{Z}_5^4 \longrightarrow \mathbb{Z}_5^4$ está dada por $f = (x_2 + 3x_4, x_4 + 5x_1 - 3x_2, x_1, x_4)$ entonces f es un PDS lineal.

Analizar la dinámica de un PDS lineal cuyas funciones de transición no poseen términos constantes se reduce a factorizar el polinomio característico de la matriz que representa el sistema lineal. Para ver más, ver [2, 3].

Definición 2.2.11. *Un PDS* $f : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ *se denomina monomial si sus funciones de transición son monomios de la forma* $\alpha x_1^{\epsilon_1} x_2^{\epsilon_2} \cdots x_n^{\epsilon_n}$ *donde* $\alpha \in F$ *y* $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \in \{1, 2, \ldots, q-1\}$.

Ejemplo 2.2.12. Sea $f = (x_1^2 x_2^2, 2x_3, x_1 x_2 x_3^2) : \mathbb{F}_3^3 \longrightarrow \mathbb{F}_3^3$. Entonces f es un sistema dinámico monomial.

Varios autores en [1, 2, 3] siguen estudiando cómo analizar la dinámica de un sistema dinámico, en particular de los sistemas dinámicos monomiales.

Capítulo 3

Álgebra computacional

Ya se ha mencionado antes que dado un conjunto de datos que representan mediciones de una red reguladora de genes, se desea conseguir el sistema dinámico polinomial que lo genere. Para cumplir con este objetivo se va a describir el método utilizado por R. Laubenbacher y B. Stigler en [9], sin embargo, antes se van a dar algunas definiciones importantes que son necesarias para entender su algoritmo.

3.1. Preliminares

3.1.1. Ideales

Definición 3.1.1. *Sea R un anillo y sea $I \subseteq R$, $I \neq \varnothing$. Se dice que I es ideal a derecha de R si:*

- $(I, +)$ *es subgrupo de* R
- *Para todo* $x \in I$ *y para todo* $r \in R$, $xr \in I$.

Similarmente, I se dice que es ideal a izquierda de R si:

- $(I, +)$ *es subgrupo de* R

- *Para todo $x \in I$ y para todo $r \in R$, $rx \in I$.*

Si I es ideal a izquierda e ideal a derecha de R, entonces I se dice que es ideal de R.

Ejemplo 3.1.2. Sea R cualquier anillo. $\{0\}$ y R son ideales de R, donde el 0 es la identidad para la suma. El primero es llamado *ideal trivial* y el segundo *ideal no propio.*

Ejemplo 3.1.3. $\mathbb{Z}$ es el anillo de los enteros. El conjunto $2\mathbb{Z}$, es decir, el conjunto de los números pares, es un ideal de $\mathbb{Z}$.

En particular, en el anillo de polinomios $\mathbb{F}[x_1, \ldots, x_n]$ también se pueden definir ideales que son útiles en nuestro estudio:

Definición 3.1.4. *Sean $f_1, \ldots, f_s$ elementos del anillo de polinomios $F[x_1, \ldots, x_n]$. Denotamos la colección $I = \langle f_1, \ldots, f_s \rangle$ al ideal del anillo de polinomios en n variables, en donde $f \in I$ si $f = p_1 f_1 + \cdots + p_s f_s$ con $p_i \in F[x_1, \ldots, x_s], i = 1, \ldots, s$; es decir, en donde el ideal consiste en el conjunto de todas las combinaciones lineales de los generadores $f_1, \ldots, f_s$ con coeficientes en el anillo.*

Ejemplo 3.1.5. Sea el anillo de polinomios $\mathbb{Q}[x, y]$ y el ideal $I \subseteq \mathbb{Q}[x, y]$ con $I = \langle x, y \rangle$. Entonces los elementos de I son de la forma $x(f_1) + y(f_2)$ donde $f_1, f_2 \in \mathbb{Q}[x, y]$.

Ejemplo 3.1.6. El polinomio $f = x^4 + x^3y + x^2 \in \langle x + y, x^2 \rangle$ porque $f = x^3(x + y) + 1(x^2)$ donde x^3 y $1 \in \mathbb{Q}[x, y]$.

Por el teorema básico de Hilbert, todo ideal en el anillo $F[x_1 \ldots, x_n]$ es finitamente generado, esto es, se puede encontrar un conjunto finito de elementos que lo genere, y en general el ideal $I = \langle f_1, \ldots, f_s \rangle$ puede ser generado por conjuntos de polinomios diferentes a los f_i, algo similar a lo que ocurre en los espacios vectoriales donde podemos encontrar diferentes bases; y por lo tanto un polinomio $f \in I$ se puede escribir como la combinación lineal de diferentes elementos de I.

Ejemplo 3.1.7. Sean los ideales $I = \langle x^2 + y,\ y \rangle$ y $J = \langle y,\ x^2 \rangle$ con $I,\ J \subseteq \mathbb{Q}[x, y]$. Entonces I y J son el mismo ideal[1] pero con diferentes conjuntos generadores. Además $f(x, y) = 3x^3 + 3xy + 16y$ pertenece a dicho ideal porque

$$f = 3x(x^2 + y) + 16(y) = 3x(x^2) + (3x + 16)(y)$$

Un tipo importante de ideales en anillos, es el que se definirá a continuación:

Definición 3.1.8. *Sea R un anillo y sea M un ideal del R. M se dice que es un ideal maximal si para todo ideal U de R tal que $M \subseteq U \subseteq R$ se tiene que $U = M$ o $U = R$.*

En particular, el anillo de polinomios $\mathbb{F}[x_1, \ldots, x_n]$ también posee ideales maximales.

3.1.2. Órdenes Monomiales

Para encontrar las bases de los ideales es importante establecer el algoritmo de división sobre polinomios de varias variables, motivo por el cual es importante determinar un orden sobre el conjunto de monomios del anillo.

Definición 3.1.9. *Sean $x_1, x_2, \ldots, x_n$ variables. Se denomina producto potencia a la expresión $x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n}$ donde $\alpha_i \in \mathbb{N}$ para todo $1 \leq i \leq n$. Tal expresión se simboliza $\boldsymbol{x}^{\alpha}$ -llamado monomio - con $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in \mathbb{N}^n$ -denominado α el vector potencia-.*

Definición 3.1.10. *Se define el grado u orden de $\boldsymbol{x}^{\alpha}$ como la suma de los exponentes de los términos del producto potencia, es decir,*

$$|\alpha| = \alpha_1 + \alpha_2 + \cdots + \alpha_n$$

Ejemplo 3.1.11. $x_1^3 x_2^6 x_3^7 x_5$ es un monomio cuyo vector potencia es (3,6,7,0,1) y su grado es 17.

[1] una manera para demostrar esto es hallando las bases de Gröbner reducidas de cada ideal y comparar que es la misma, procedimiento que no estamos profundizando en este trabajo

Definición 3.1.12. *Sea A un conjunto y $\prec$ una relación de orden sobre A. Se dice que $\prec$ es un orden total si para cada dos elementos $a, b \in A$ se tiene que $a \prec b$, $b \prec a$ o $a = b$.*

Definición 3.1.13. *Sea $\mathbb{F}[x_1, x_2, \ldots, x_n]$ el anillo de polinomios en n variables. Se define el ordenamiento monomial sobre $\mathbb{F}[x_1, x_2, \ldots, x_n]$ como una relación $\prec$ sobre el conjunto de monomios $\boldsymbol{x}^\alpha$ en $\mathbb{F}[x_1, x_2, \ldots, x_n]$ que satisface:*

- *$\prec$ es una ordenación total.*
- *$\prec$ es compatible con la multiplicación en $\mathbb{F}[x_1, x_2, \ldots, x_n]$, en el sentido que si $\boldsymbol{x}^\alpha \prec \boldsymbol{x}^\beta$ y $\boldsymbol{x}^\gamma$ es un monomio, entonces $\boldsymbol{x}^\alpha \boldsymbol{x}^\gamma = \boldsymbol{x}^{\alpha+\gamma} \prec \boldsymbol{x}^\beta \boldsymbol{x}^\gamma = \boldsymbol{x}^{\beta+\gamma}$ para todo $\boldsymbol{x}^\alpha, \boldsymbol{x}^\beta, \boldsymbol{x}^\gamma \in \mathbb{F}[x_1, x_2, \ldots, x_n]$.*
- *$1 \prec \boldsymbol{x}^\alpha$ para todo $\boldsymbol{x}^\alpha \neq 1$*
- *$\prec$ es un buen ordenamiento, es decir, cualquier subconjunto no vacio de monomios de $\mathbb{F}[x_1, x_2, \ldots, x_n]$ tiene un elemenento mínimo bajo $\prec$.*

Existen varios tipos de órdenes, los cuales se definen a continuación:

Definición 3.1.14. *Sean $\boldsymbol{x}^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n}$ y $\boldsymbol{x}^\beta = x_1^{\beta_1} x_2^{\beta_2} \ldots x_n^{\beta_n}$ dos monomios del anillo $\mathbb{F}[x_1, x_2, \ldots, x_n]$ sobre el cuerpo $\mathbb{F}$. Se dice que $\boldsymbol{x}^\alpha \prec_{lex} \boldsymbol{x}^\beta$ si existe $1 \leq i \leq n$ tal que $\alpha_j = \beta_j$ para $1 \leq j \leq i-1$ y $\alpha_i \prec \beta_i$. Este ordenamiento es llamado orden lexicográfico o léxico y denotado $\prec_{lex}$.*

Ejemplo 3.1.15. Sea $\mathbb{F}[x_1, x_2]$ el anillo de polinomios en dos variables sobre el cuerpo $\mathbb{F}$. Bajo este orden se tiene que $x_2 \prec_{lex} x_1$ porque el vector potencia de x_2 es (0,1) y el de x_1 es (1,0) y $(0, 1) \prec (1, 0)$ porque comparando la primera componente de ambos vectores se tiene que $0 < 1$.

Ejemplo 3.1.16. Dando una generalización del ejemplo anterior si $\mathbb{F}[x_1, x_2, \ldots, x_n]$ es un anillo de polinomios en n variables sobre el cuerpo $\mathbb{F}$, bajo el orden lexicográfico se tiene que $x_n \prec_{lex} \cdots \prec_{lex} x_2 \prec_{lex} x_1$.

Ejemplo 3.1.17. Sea $\mathbb{Q}[x_1, x_2, x_3]$ el anillo de polinomios en 3 variables sobre $\mathbb{Q}$. Sean $5x_1^3x_2^7x_3^6$ y $6x_1^3x_2^6x_3^8$ monomios de este anillo. Según el orden lexicográfico se tiene que $6x_1^3x_2^6x_3^8 \prec_{lex} 5x_1^3x_2^7x_3^6$ porque $(3,6,8) \prec (3,7,6)$ ya que al comparar componente por componente se obtiene que $3 = 3$ pero $6 < 7$.

Definición 3.1.18. *Sean* $\boldsymbol{x}^\alpha = x_1^{\alpha_1}x_2^{\alpha_2}\dots x_n^{\alpha_n}$ *y* $\boldsymbol{x}^\beta = x_1^{\beta_1}x_2^{\beta_2}\dots x_n^{\beta_n}$ *dos monomios del anillo* $\mathbb{F}[x_1, x_2, \dots, x_n]$ *sobre el cuerpo* $\mathbb{F}$. *Se dice que* $\boldsymbol{x}^\alpha \prec_{ilex} \boldsymbol{x}^\beta$ *si existe* $1 \leq i \leq n$ *tal que* $\alpha_j = \beta_j$ *para* $i+1 \leq j \leq n$ *y* $\alpha_i \prec \beta_i$ *. Este ordenamiento es llamado orden lexicográfico inverso o léxico inverso y denotado* $\prec_{ilex}$.

Ejemplo 3.1.19. Sea $\mathbb{F}[x_1, x_2]$ el anillo de polinomios en dos variables sobre el cuerpo $\mathbb{F}$. Bajo el orden $\prec_{ilex}$ se tiene que $x_1 \prec_{ilex} x_2$ porque el vector potencia de x_1 es (1,0) y el de x_2 es (0,1) y se tiene que $(1,0) \prec (0,1)$ porque comparando de derecha a izquierda $0 < 1$.

Ejemplo 3.1.20. Dando una generalización del ejemplo anterior si $\mathbb{F}[x_1, x_2, \dots, x_n]$ es el anillo de polinomios en n variables sobre el cuerpo $\mathbb{F}$, bajo el orden lexicográfico inverso se tiene que $x_1 \prec_{ilex} \cdots \prec_{ilex} x_{n-1} \prec_{ilex} x_n$.

Ejemplo 3.1.21. Sean $x_1^3x_2^3x_3^4$ y $x_1^2x_2^4x_3^4$ dos monomios del anillo $\mathbb{Q}[x_1, x_2, x_3]$. Se tiene que $x_1^3x_2^3x_3^4 \prec_{ilex} x_1^2x_2^4x_3^4$ porque comparando los vectores potencia de derecha a izquierda se tiene que $(3,3,4) < (2,4,4)$ pues aunque $4 = 4$, $3 < 4$.

Definición 3.1.22. *Sean* $\boldsymbol{x}^\alpha$ *y* $\boldsymbol{x}^\beta$ *dos monomios del anillo* $\mathbb{F}[x_1, x_2, \dots, x_n]$ *sobre el cuerpo* $\mathbb{F}$. *Se dice que* $\boldsymbol{x}^\alpha \prec_{grlex} \boldsymbol{x}^\beta$ *si* $|\alpha| < |\beta|$ *o si* $|\alpha| = |\beta|$ *y* $\boldsymbol{x}^\alpha \prec_{lex} \boldsymbol{x}^\beta$. *Este ordenamiento se denomina orden lexicográfico total por grados.*

Ejemplo 3.1.23. Se tiene que $x_1^3x_2^4 \prec_{grlex} x_1^6x_2^2$ porque $|(3,4)| = 7 < 8 = |(6,2)|$.

Ejemplo 3.1.24. $x_1^3x_2^4x_3^5 \prec_{grlex} x_1^4x_2^4x_3^4$ porque $|(3,4,5)| = 12 = |(4,4,4)|$ y $x_1^3x_2^4x_3^5 \prec_{lex} x_1^4x_2^4x_3^4$.

Definición 3.1.25. *Sean* $\boldsymbol{x}^\alpha$ *y* $\boldsymbol{x}^\beta$ *dos monomios del anillo* $\mathbb{F}[x_1, x_2, \ldots, x_n]$ *sobre el cuerpo* $\mathbb{F}$*. Se dice que* $\boldsymbol{x}^\alpha \prec_{igrlex} \boldsymbol{x}^\beta$ *si* $|\alpha| < |\beta|$ *o si* $|\alpha| = |\beta|$ *y* $\boldsymbol{x}^\alpha \prec_{ilex} \boldsymbol{x}^\beta$*. Este ordenamiento se denomina orden lexicográfico inverso total por grados.*

Ejemplo 3.1.26. $x_1^2 \prec_{igrlex} x_2^2$ porque $|(2,0)| = |(0,2)|$ y $x_1^2 \prec_{ilex} x_2^2$.

Ejemplo 3.1.27. $x_1^2x_2^5x_3^4 \prec_{igrlex} x_1^2x_2^4x_3^4$ porque $|(2,5,4)| = 11 > 10 = |(2,4,4)|$.

Definición 3.1.28. *Sean* $\boldsymbol{x}^\alpha$ *y* $\boldsymbol{x}^\beta$ *dos monomios del anillo* $\mathbb{F}[x_1, x_2, \ldots, x_n]$ *sobre el cuerpo* $\mathbb{F}$*. Sea* $\prec$ *un ordenamiento arbitrario de monomios y* $w \in \mathbb{N}^n$*. Se define un nuevo ordenamiento* $\prec_w$ *así:* $\boldsymbol{x}^\alpha \prec_w \boldsymbol{x}^\beta$ *si* $w \cdot \alpha < w \cdot \beta$ *, o, si* $w \cdot \alpha = w \cdot \beta$ *pero* $\boldsymbol{x}^\alpha \prec \boldsymbol{x}^\beta$*.*

Nota: $w \cdot \alpha$ denota el producto punto entre w y α.

Ejemplo 3.1.29. Sea $w = (2,1,3)$ y $\prec$ el orden lexicográfico. $x_1^2x_2^4x_3^4 \prec_w x_1^2x_2^5x_3^4$ porque $w \cdot (2,5,4) = 20 < 21 = w \cdot (2,5,4)$.

Ejemplo 3.1.30. Sea $w = (4,3,2)$ y $\prec$ el orden lexicográfico. Entonces $x_1^2x_2^5x_3^4 \prec_w x_1^4x_2^5$ porque $w \cdot (2,5,4) = 31 = w \cdot (4,5,0)$ y $x_1^2x_2^5x_3^4 \prec_{lex} x_1^4x_2^5$.

Un ordenamiento de monomios, induce una ordenación de los términos de un polinomio f de $\mathbb{F}[x_1, x_2, \ldots, x_n]$, organizándolos de modo decreciente.

Definición 3.1.31. *Sea* $\mathbb{F}[x_1, x_2, \ldots, x_n]$ *el anillo de polinomios en múltiples variables sobre el cuerpo* $\mathbb{F}$ *y sea* f *un elemento de este anillo. Sea* $\prec$ *un orden fijo. Se define el término líder de f, denotado por* $lt(f)$*, como el término mayor de* f *bajo el orden* $\prec$*; el coeficiente líder de f, denotado por* $lc(f)$ *como el coefiente del término líder de* f*, y el producto potencia del término líder, denotado por* $lp(f)$*, como el producto potencia líder.*

Note que lp, lc y lt son claramente multiplicativos, es decir, $lp(fg) = lp(f)lp(g)$, $lc(fg) = lc(f)lc(g)$ y $lt(fg) = lt(f)lt(g)$.

Ejemplo 3.1.32. Sea $f(x_1, x_2) = 3x_1^2x_2 + 5x_1x_2 + 4x_1x_2^2$ un polinomio de $\mathbb{Q}[x_1, x_2]$. Utilizando el orden lexicográfico se tiene que $5x_1x_2 \prec_{lex} 4x_1x_2^2 \prec_{lex} 3x_1^2x_2$. Organizando a f en forma decreciente queda $f(x_1, x_2) = 3x_1^2x_2 + 4x_1x_2^2 + 5x_1x_2$. Entonces $lt(f) = 3x_1^2x_2$, $lc(f) = 3$ y $lp(f) = x_1^2x_2$.

Ejemplo 3.1.33. Sea $f(x_1, x_2, x_3) = 5x_1^2x_2^2x_3^2 + 6x_1^2x_2x_3 + 8x_1x_2^2x_3 - 3x_1x_3 + 4x_2x_3 - x_2 + x_3 + 7x_1$ un polinomio de $\mathbb{Q}[x_1, x_2, x_3]$. Entonces:

- Bajo el orden lexicográfico, f es ordenado en orden decreciente así: $f(x_1, x_2, x_3) = 5x_1^2x_2^2x_3^2 + 6x_1^2x_2x_3 + 8x_1x_2^2x_3 - 3x_1x_3 + 7x_1 + 4x_2x_3 - x_2 + x_3$ y entonces $lt(f) = 5x_1^2x_2^2x_3^2$, $lp(f) = x_1^2x_2^2x_3^2$ y $lc(f) = 5$.

- Bajo el orden lexicográfico inverso, f queda ordenado en forma decreciente así: $f(x_1, x_2, x_3) = 5x_1^2x_2^2x_3^2 + 8x_1x_2^2x_3 + 6x_1^2x_2x_3 + 4x_2x_3 - 3x_1x_3 + x_3 - x_2 + 7x_1$ luego $lt(f) = 5x_1^2x_2^2x_3^2$, $lp(f) = 5x_1^2x_2^2x_3^2$ y $lc(f) = 5$.

- Si utilizamos el orden lexicográfico total por grados f quedaría ordenado así: $f(x_1, x_2, x_3) = 5x_1^2x_2^2x_3^2 + 6x_1^2x_2x_3 + 8x_1x_2^2x_3 - 3x_1x_3 + 4x_2x_3 + 7x_1 - x_2 + x_3$ y entonces $lt(f) = 5x_1^2x_2^2x_3^2$, $lp(f) = x_1^2x_2^2x_3^2$ y $lc(f) = 5$.

- Si utilizamos el orden lexicográfico inverso total por grados podríamos reordenar a f como sigue: $f(x_1, x_2, x_3) = 5x_1^2x_2^2x_3^2 + 8x_1x_2^2x_3 + 6x_1^2x_2x_3 + 4x_2x_3 - 3x_1x_3 + x_3 - x_2 + 7x_1$ en donde $lt(f) = 5x_1^2x_2^2x_3^2$, $lp(f) = x_1^2x_2^2x_3^2$ y $lc(f) = 5$.

Ejemplo 3.1.34. Sea $f = 3x^4y^2 + 3x^3y^4 - 4xy^4 + 2y^3 - 5y^5$ elemento del anillo $\mathbb{Q}[x, y]$

- Bajo el orden lexicográfico con $x > y$, f quedaría ordenado así $f = 3x^4y^2 + 3x^3y^4 - 4xy^4 - 5y^5 + 2y^3$ de donde $lt(f) = 3x^2y^2$.

- Bajo el orden lexicográfico inverso con $x > y$, f se ordena $f = -5y^5 - 4xy^4 + 3x^3y^4 + 2y^3 + 3x^4y^2$ y así $lt(f) = -5y^5$.
- Bajo el orden lexicográfico total por grados con $x > y$, $f = 3x^3y^4 + 3x^4y^2 - 4xy^4 - 5y^5 + 2y^3$ de donde $lt(f) = 3x^3y^4$.
- Bajo el orden lexicográfico inverso total por grados con $x < y$, $lt(f) = 3x^3y^4$ porque $f = 3x^3y^4 + 3x^4y^2 - 5y^5 - 4xy^4 + 2y^3$ y el término líder es $3x^3y^4$.

3.1.3. Reducción de polinomios

Definición 3.1.35. *Un polinomio h se dice que es una reducción de f módulo g en un paso con $f, g, h \in \mathbb{F}[x_1, x_2, \ldots, x_n]$, o también, f se reduce hasta h módulo g en un paso denotado por $f \xrightarrow{g} h$ si $lt(g)|lt(f)$ y*

$$h = f - \frac{lt(f)}{lt(g)}g$$

donde h es llamado el residuo que se obtiene al dividir f por g.

Repitiendo el proceso anterior un número finito de veces, es decir, si se reduce f módulo g en varios pasos, esto es, $f \xrightarrow{g} h_1 \xrightarrow{g} \cdots h_k \xrightarrow{g} h$ se denota por

$$f \xrightarrow{g_{(k)}} h$$

Ejemplo 3.1.36. Sean $f(x) = 5x^4 - 3x^2 + 1$ y $g(x) = x - 1$, elementos de $\mathbb{Q}[x]$. Si reducimos f módulo g en un paso queda:

$$h_1 = (5x^4 - 3x^2 + 1) - \frac{5x^4}{x}(x - 1) = 5x^3 - 3x^2 + 1$$

Y si se reduce f módulo g en 3 pasos quedaría así: primero se reduce f módulo g en un paso

$$h_1 = 5x^3 - 3x^2 + 1$$

ahora haciendo h_1 módulo g

$$h_2 = (5x^3 - 3x^2 + 1) - \frac{5x^3}{x}(x-1) = 2x^2 + 1$$

y se continua reduciendo h_2 módulo g

$$h_3 = (2x^2 + 1) - \frac{2x^2}{x}(x-1) = 2x + 1$$

y nuevamente, h_3 módulo g

$$h_3 = (2x + 1) - \frac{2x}{x}(x-1) = 3$$

luego $f \overset{g_{(3)}}{\to} 3$.

Ejemplo 3.1.37. Sea $f(x,y) = 8x^2y - 4xy + 4x - 5y + 1$ y $g(x,y) = 4xy - x + y - 2$ polinomios de $\mathbb{Q}[x,y]$ bajo el orden lexicográfico con $y \prec x$. Al reducir f hasta h módulo g, se tiene que

$$\begin{aligned} h(x,y) &= (8x^2y - 4xy + 4x - 5y + 1) - \frac{8x^2y}{4xy}(4xy - x + y - 2) \\ &= 2x^2 - 6xy + 8x - 5y + 1 \end{aligned}$$

porque $lt(f) = 8x^2y$ y $lt(g) = 4xy$ bajo este orden.

Definición 3.1.38. *Sea f, h y $f_1, f_2, \ldots, f_s$ con $f_i \neq 0$ para todo i, polinomios de $\mathbb{F}[x_1, x_2, \ldots, x_s]$ y $G = \{f_1, f_2, \ldots, f_s\}$. Se dice que f se reduce hasta h módulo G, denotado por $f \overset{G_+}{\to} h$ si existen $i_1, i_2, \ldots, i_t \in \{1, \ldots, s\}$ y $h_1, h_2, \ldots, h_{t-1}$ tales que*

$$f \overset{f_{i_1}}{\to} h_1 \overset{f_{i_2}}{\to} h_2 \overset{f_{i_3}}{\to} \cdots \overset{f_{i_{t-1}}}{\to} h_{t-1} \overset{f_{i_t}}{\to} h$$

Si no existe algún i_k en $\{1, \ldots, s\}$ se dice que f es irreducible módulo G.

Ejemplo 3.1.39. Considere el polinomio $f(x,y) = 4x^2y^2 - 3x^2y + xy^2 - 5x + y - 10$ y el conjunto G formado por los polinomios $f_1(x,y) = 2x^2 + 3x - 1$ y $f_2(x,y) = 5y^2 - x + 10$ elementos del anillo $\mathbb{Q}[x,y]$ bajo el orden lexicográfico. Vamos a mostrar como f se reduce hasta h módulo G.

Tenemos que $lt(f) = 4x^2y^2$, $lt(f_1) = 2x^2$ y $lt(f_2) = 5y^2$. Como $lt(f_1) \mid lt(f)$ entonces podemos reducir f módulo f_1 esto es $f \xrightarrow{f_1} h_1$

$$\begin{aligned} h_1 &= (4x^2y^2 - 3x^2y + xy^2 - 5x + y - 10) - \frac{4x^2y^2}{2x^2}(2x^2 + 3x - 1) \\ &= -3x^2y - 5xy^2 + 2y^2 - 5x + y - 10 \end{aligned}$$

Ahora $lt(h_1) = -3x^2y, lt(f_1) = 2x^2$ y $lt(f_2) = 5y^2$. Como $lt(f_1) \mid lt(h_1)$ entonces hacemos la reducción de h_1 módulo f_1, esto es

$$\begin{aligned} h_2 &= (-3x^2y - 5xy^2 + 2y^2 - 5x + y - 10) - \frac{-3x^2y}{2x^2}(2x^2 + 3x - 1) \\ &= -5xy^2 + 2y^2 + \tfrac{9}{2}xy - 5x - \tfrac{1}{2}y - 10 \end{aligned}$$

Ahora, el $lt(h_2) = -5xy^2$, $lt(f_1) = 2x^2$ y $lt(f_2) = 5y^2$, y así como $lt(f_2) \mid lt(h_2)$ entonces ahora hacemos la reducción de h_2 módulo f_2, entonces

$$\begin{aligned} h_3 &= (-5xy^2 + 2y^2 + \tfrac{9}{2}xy - 5x - \tfrac{1}{2}y - 10) - \frac{-5xy^2}{5y^2}(5y^2 - x + 10) \\ &= -x^2 + 2y^2 - \tfrac{9}{2}xy + 5x - \tfrac{1}{2}y - 10 \end{aligned}$$

y siguiendo $lt(h_3) = -x^2$, $lt(f_1) = 2x^2$ y $lt(f_2) = 5y^2$ y como $lt(f_1) \mid lt(h_3)$ entonces hacemos la reducción de h_3 hasta h_4 módulo f_1

$$\begin{aligned} h_4 &= -x^2 + 2y^2 - \tfrac{9}{2}xy + 5x - \tfrac{1}{2}y - 10 - \frac{-x^2}{2x^2}(2x^2 + 3x - 1) \\ &= -\tfrac{9}{2}xy + \tfrac{13}{2}x + 2y^2 - \tfrac{1}{2}y - \tfrac{21}{2} \end{aligned}$$

y de nuevo, $lt(h_4) = \frac{9}{2}xy$, $lt(f_1) = 2x^2$ y $lt(f_2) = 5y^2$. Como $lt(f_1) \nmid lt(h_4)$ y $lt(f_2) \nmid lt(h_4)$, decimos que h_4 es irreducible módulo G.

Definición 3.1.40. *Un polinomio f se llama completamente reducido hasta h respecto a g si no hay algún término en f que sea divisible por $lt(g)$. Es decir,*

$$h = f - \frac{X}{lt(g)}g$$

donde X es un término de f y $lt(g) \nmid X$.

Ejemplo 3.1.41. Sea $f = x^3 - x^2y^2 + y^4$ y $g = x^2y + xy^3 + y^2$ polinomios del anillo $\mathbb{Q}[x, y]$ dotado del orden lexicográfico. La reducción completa de f por g es:

Como $lt(f_1) = x^2y \mid -x^2y^2$ entonces

$$\begin{aligned} h &= x^3 - x^2y^2 + y^4 - \frac{-x^2y^2}{x^2y}(x^2y + xy^3 + y^2) \\ &= x^3 - x^2y^2 + y^4 + y(x^2y + xy^3 + y^2) \\ &= x^3 + xy^4 + y^4 + y^3 \end{aligned}$$

Definición 3.1.42. *Un polinomio f se llama completamente reducido con respecto a F si no hay algún término en f que sea divisible por algún $lt(f_i)$ para todo $f_i \in F$.*

Ejemplo 3.1.43. Sean $f = x^3y^2+xy^3+xy^2-4xy-3$, $f_1 = xy^2+3xy-2$ y $f_2 = x^2+xy-x$ elementos del anillo $\mathbb{Q}[x, y]$ con el orden lexicográfico, y F el conjunto $F = \{f_1, f_2\}$. La reducción completa de f por F sería:

Como $f = x^3y^2 + xy^3 + xy^2 - 4xy - 3$ y $lt(f_1) = xy^2 \mid x^3y^2$ entonces

$$\begin{aligned} h_1 &= (x^3y^2 + xy^3 + xy^2 - 4xy - 3) - \frac{x^3y^2}{xy^2}(xy^2 + 3xy - 2) \\ &= (x^3y^2 + xy^3 + xy^2 - 4xy - 3) - x^2(xy^2 + 3xy - 2) \\ &= -3x^3y + 2x^2 + xy^3 + xy^2 - 4xy - 3 \end{aligned}$$

Y seguimos ahora reduciendo h_1 por F. Como $lt(f_1) = xy^2 \mid xy^3$ entonces

$$\begin{aligned} h_2 &= (-3x^3y + 2x^2 + xy^3 + xy^2 - 4xy - 3) - \frac{xy^3}{xy^2}(xy^2 + 3xy - 2) \\ &= (-3x^3y + 2x^2 + xy^3 + xy^2 - 4xy - 3) - y(xy^2 + 3xy - 2) \\ &= -3x^3y + 2x^2 - 2xy^2 - 4xy + 2y - 3 \end{aligned}$$

y podemos seguir reduciendo h_2 por f_1 porque $lt(f_1) = xy^2 \mid -2xy^2$ luego

$$\begin{aligned} h_3 &= (-3x^3y + 2x^2 - 2xy^2 - 4xy + 2y - 3) - \frac{-2xy^2}{xy^2}(xy^2 + 3xy - 2) \\ &= (-3x^3y + 2x^2 - 2xy^2 - 4xy + 2y - 3) + 2(xy^2 + 3xy - 2) \\ &= -3x^3y + 2x^2 + 2xy + 2y - 7 \end{aligned}$$

Ahora no podemos reducir h_3 por f_1 porque no hay ningún término de h_3 que sea divisible por f_1. Pero como $lt(f_2) = x^2 \mid -3x^3y$ entonces

$$\begin{aligned} h_4 &= (-3x^3y + 2x^2 + 2xy + 2y - 7) - \frac{-3x^3y}{x^2}(x^2 + xy - x) \\ &= (-3x^3y + 2x^2 + 2xy + 2y - 7) + 3xy(x^2 + xy - x) \\ &= 3x^2y^2 - 3x^2y + 2x^2 + 2xy + 2y - 1 \end{aligned}$$

Podemos reducir h_4 por f_1 porque $lt(f_1) = xy^2 \mid 3x^2y^2$ y entonces

$$\begin{aligned} h_5 &= (3x^2y^2 - 3x^2y + 2x^2 + 2xy + 2y - 1) - \frac{3x^2y^2}{xy^2}(xy^2 + 3xy - 2) \\ &= (3x^2y^2 - 3x^2y + 2x^2 + 2xy + 2y - 1) - 3x(xy^2 + 3xy - 2) \\ &= -12x^2y + 2x^2 + 2xy + 6x + 2y - 7 \end{aligned}$$

y si seguimos reduciendo h_5 por f_2 ya que $lt(f_2) = x^2 \mid -12x^2y$ entonces

$$\begin{aligned} h_6 &= (-12x^2y + 2x^2 + 2xy + 6x + 2y - 7) - \frac{-12x^2y}{x^2}(x^2 + xy - x) \\ &= (-12x^2y + 2x^2 + 2xy + 6x + 2y - 7) + 12y(x^2 + xy - x) \\ &= 2x^2 + 12xy^2 - 10xy + 6x + 2y - 7 \end{aligned}$$

y ahora h_6 por f_1 debido que $lt(f_1) = xy^2 \mid 12xy^2$ luego

$$\begin{aligned} h_7 &= (2x^2 + 12xy^2 - 10xy + 6x + 2y - 7) - \frac{12xy^2}{xy^2}(xy^2 + 3xy - 2) \\ &= (2x^2 + 12xy^2 - 10xy + 6x + 2y - 7) - 12(xy^2 + 3xy - 2) \\ &= 2x^2 - 46xy + 6x + 2y + 17 \end{aligned}$$

y siguiendo

$$\begin{aligned} h_8 &= (2x^2 - 46xy + 6x + 2y + 17) - \frac{2x^2}{x^2}(x^2 + xy - x) \\ &= (2x^2 - 46xy + 6x + 2y + 17) - 2(x^2 + xy - x) \\ &= -48xy + 8x + 2y + 17 \end{aligned}$$

porque $lt(f_2) = x^2 \mid 2x^2$

Y así, al reducir completamente f por F da como resultado $-48xy + 8x + 2y + 17$.

Definición 3.1.44. *Un polinomio h se dice que está en la forma normal de f si y sólo si $f \overset{G+}{\rightarrow} h$ y h es completamente irreducible módulo G.*

Ejemplo 3.1.45. En el ejemplo anterior, se puede decir que $h_8 = -48xy+8x+2y+17$ está en la forma normal de $f = x^3y^2 + xy^3 + xy^2 - 4xy - 3$ respecto a $F = \{f_1 = xy^2 + 3xy - 2, f_2 = x^2 + xy - x\}$.

3.1.4. Algoritmo de división en múltiples variables

Definición 3.1.46. *Sean $f_1, f_2, \ldots, f_s$ polinomios de $\mathbb{F}[x_1, \ldots, x_n]$ con $f_i \neq 0$ para $1 \leq i \leq s$ y sea $\prec$ un orden fijo. Entonces, para todo $f \in \mathbb{F}[x_1, \ldots, x_n]$ si existen $u_1, u_2, \ldots, u_s$ y $r \in F[x_1, \ldots, x_n]$ tales que*

$$f = u_1 f_1 + u_2 f_2 + \cdots + u_s f_s + r$$

con r reducido respecto a $f_1, \ldots, f_s$ y $lp(f) = max(max_{1 \leq i \leq s}(lp(u_i)lp(f_i), lp(r)))$. Este procedimento es llamado algoritmo de división en $F[x_1, \ldots, x_n]$.

El algoritmo de división sobre un anillo $\mathbb{F}[x_1, \ldots, x_n]$ se describe en el **Algoritmo 1** cuyas entradas son $f, f_1, \ldots, f_s$ y la salida $u_1, \ldots, u_s, r$ todos ellos elementos de $\mathbb{F}[x_1, \ldots, x_n]$.

INICIO

Paso 1: Entrada $f, f_1, \ldots, f_s$

Paso 2: $u_1 = u_2 = \cdots = u_s = r = 0$, $m = f$

Paso 3: Mientras $m \neq 0$ haga

Encontrar el $i = min\{1, \ldots, s\}$ tal que $lp(f_i) \mid lp(m)$

Si i existe entonces haga:

$u_i = u_i + \frac{lt(m)}{lt(f_i)}$

$m = m - \frac{lt(m)}{lt(f_i)} f_i$

si no existe haga:

$r = r + lt(m)$

$m = m - lt(m)$

Paso 4: Retorne $u_1, \ldots, u_s, r$

FIN

Algoritmo 1: Algoritmo de división en múltiples variables

Ejemplo 3.1.47. Se quiere expresar al polinomio $f(x_1, x_2) = x_1^2 x_2 + x_1 x_2^2 + x_2^2 \in \mathbb{Q}[x_1, x_2]$ bajo el orden lexicográfico, en términos de $f_1(x_1, x_2) = x_1 x_2 - 1$ y $f_2(x_1, x_2) = x_2^2 - 1$, entonces aplicando el algoritmo de división se obtiene:

INICIO

Paso 2: $u_1 = 0, u_2 = 0, m = x_1^2 x_2 + x_1 x_2^2 + x_2^2$

Paso 3:

3.1. $m = x_1^2 x_2 + x_1 x_2^2 + x_2^2 \neq 0$

Como $lp(f_1) = x_1 x_2 \mid lp(m) = x_1^2 x_2$, entonces $i = 1$

$u_1 = 0 + \frac{x_1^2 x_2}{x_1 x_2} = x_1$

$m = (x_1^2 x_2 + x_1 x_2^2 + x_2^2) - \frac{x_1^2 x_2}{x_1 x_2}(x_1 x_2 - 1) = x_1 x_2^2 + x_2^2 + x_1$

3.2. $m = x_1 x_2^2 + x_2^2 + x_1 \neq 0$

Como $lp(f_1) = x_1 x_2 \mid lp(m) = x_1 x_2^2$, entonces $i = 1$

$u_1 = x_1 + \frac{x_1 x_2^2}{x_1 x_2} = x_1 + x_2$

$m = (x_1 x_2^2 + x_2^2 + x_1) - \frac{x_1 x_2^2}{x_1 x_2}(x_1 x_2 - 1) = x_1 + x_2^2 + x_2$

3.3. $m = x_1 + x_2^2 + x_2 \neq 0$

Como $lp(f_1) = x_1 x_2 \nmid lp(m) = x_1$ y $lp(f_2) = x_2^2 \nmid lp(m) = x_1$ entonces

i no existe, luego

$r = 0 + x_1 = x_1$

$m = (x_1 + x_2^2 + x_2) - x_1 = x_2^2 + x_2$

3.4. $m = x_2^2 + x_2 \neq 0$

Como $lp(f_1) = x_1 x_2 \nmid lp(m) = x_1$ pero $lp(f_2) = x_2^2 \mid lp(m) = x_2^2$

entonces $i = 2$, luego

$u_2 = 0 + \frac{x_2^2}{x_2^2} = 1$

$m = (x_2^2 + x_2) - \frac{x_2^2}{x_2^2}(x_2^2 - 1) = x_2 + 1$

3.5. $m = x_2 + 1 \neq 0$

Como $lp(f_1) = x_1 x_2 \nmid lp(m) = x_2$ y $lp(f_2) = x_2^2 \nmid lp(m) = x_2$ entonces

i no existe, luego

$r = x_1 + x_2$

$m = x_2 + 1 - x_2 = 1$

3.6. $m = 1 \neq 0$

Como $lp(f_1) = x_1x_2 \nmid lp(m) = 1$ y $lp(f_2) = x_2^2 \nmid lp(m) = 1$ entonces

i no existe , luego

$r = x_1 + x_2 + 1$

$m = 1 - 1 = 0$

3.7. $m = 0$

Paso 4: Entonces $u_1 = x_1 + x_2$, $u_2 = 1$ y $r = x_1 + x_2 + 1$

FIN

Nota: Es importante saber que de acuerdo al orden de los f_i en el algoritmo anterior, los cocientes y el residuo varían. Se ilustra esto con un ejemplo.

Desarrollemos el mismo ejercicio anterior pero con los f_i en diferente orden:

Ejemplo 3.1.48. Se quiere expresar al polinomio $f(x_1, x_2) = x_1^2x_2 + x_1x_2^2 + x_2^2 \in \mathbb{Q}[x_1, x_2]$ bajo el orden lexicográfico, en términos de $f_1(x_1, x_2) = x_2^2 - 1$ y $f_2(x_1, x_2) = x_1x_2 - 1$, entonces aplicando el algoritmo de división se obtiene:

INICIO

Paso 2: $u_1 = 0, u_2 = 0, m = x_1^2x_2 + x_1x_2^2 + x_2^2$

Paso 3:

3.1. $m = x_1^2x_2 + x_1x_2^2 + x_2^2 \neq 0$

Como $lp(f_1) = x_2^2 \nmid lp(m) = x_1^2x_2$, pero $lp(f_2) = x_1x_2 \mid lp(m) = x_1^2x_2$

entonces $i = 2$

$u_2 = 0 + \frac{x_1^2x_2}{x_1x_2} = x_1$

$m = (x_1^2x_2 + x_1x_2^2 + x_2^2) - \frac{x_1^2x_2}{x_1x_2}(x_1x_2 - 1) = x_1x_2^2 + x_2^2 + x_1$

3.2. $m = x_1x_2^2 + x_2^2 + x_1 \neq 0$

Como $lp(f_1) = x_2^2 \mid lp(m) = x_1x_2^2$, entonces $i = 1$

$u_1 = 0 + \frac{x_1x_2^2}{x_2^2} = x_1$

$m = (x_1x_2^2 + x_2^2 + x_1) - \frac{x_1x_2^2}{x_2^2}(x_2^2 - 1) = 2x_1 + x_2$

3.3. $m = 2x_1 + x_2^2 \neq 0$

Como $lp(f_1) = x_2^2 \nmid lp(m) = 2x_1$ y $lp(f_2) = x_1x_2 \nmid lp(m) = 2x_1$

entonces i no existe, luego

$r = 0 + 2x_1 = 2x_1$

$m = (2x_1 + x_2^2) - 2x_1 = x_2^2$

3.4. $m = x_2^2 \neq 0$

Como $lp(f_1) = x_2^2 \mid lp(m) = x_2^2$ entonces $i = 1$, luego

$u_1 = x_1 + \frac{x_2^2}{x_2^2} = x_1 + 1$

$m = x_2^2 - \frac{x_2^2}{x_2^2}(x_2^2 - 1) = 1$

3.5. $m = 1 \neq$

Como $lp(f_1) = x_2^2 \nmid lp(m) = 1$ y $lp(f_2) = x_1x_2 \nmid lp(m) = 1$ entonces

i no existe, luego

$r = 2x_1 + 1$

$m = 1 - 1 = 0$

3.6. $m = 0$

Paso 4: Entonces $u_1 = x_1 + 1$, $u_2 = x_1$ y $r = 2x_1 + 1$

FIN

Podemos encontrar una manera para que los cocientes en la división sesan únicos,

haciendo que los divisores de ésta formen una Base de Göbner.

3.1.5. Bases de Gröbner

Definición 3.1.49. *Sea $\mathbb{F}[x_1, \ldots, x_n]$ el anillo de polinomios sobre el cuerpo $\mathbb{F}$. Sea I un ideal de $\mathbb{F}[x_1, \ldots, x_n]$ y $\prec$ un ordenamiento de monomios para $F[x_1, \ldots, x_n]$. Un conjunto finito $G = \{g_1, \ldots, g_s\} \subseteq I$ se llama Base de Gröbner para I respecto a $\prec$ si $\forall f \in I$, $lt(g_i)|lt(f)$ para algún $1 \leq i \leq s$.*

Ejemplo 3.1.50. Sea el conjunto $G = \{g_1 = 3xy - 4x - y, g_2 = xy - y\} \subseteq \mathbb{Q}[x, y]$ dotado del orden lexicográfico con $y < x$, e $I = \langle g_1, g_2 \rangle$, ideal. Entonces G no es una base de Gröbner porque $f = -4x + 2y = 1(3xy - 4x - y) - 3(xy - y) \in I$ pero $lt(g_1) = 3xy \nmid lt(f) = -4x$ y $lt(g_2) = xy \nmid lt(f) = -4x$.

Definición 3.1.51. *Sea $F[x_1, \ldots, x_n]$ un anillo de polinomios y sea $\prec$ un ordenamiento de monomios. Sea $A \subseteq F[x_1, \ldots, x_n]$. Se define el ideal de los términos líderes de A o el ideal de los términos iniciales de A al ideal*

$$Lt_{\prec}(A) = \langle lt(f) | f \in A \rangle$$

En particular, si $A = I$ donde I es un ideal de $F[x_1, \ldots, x_n]$, al conjunto $Lt(I)$ se le llama ideal inicial. A los términos que no viven en este ideal se les llama términos estándar.

Nota: Tener presente que $Lt(I)$ consiste en los términos líderes de TODOS los polinomios que hacen parte del ideal, mas no solamente de los generadores del ideal.

Ejemplo 3.1.52. Se tiene que $Lt(I) = \langle x_1^2, x_2^3 \rangle$ para $I \subseteq \mathbb{Q}[x_1, x_2]$ ideal. Se dice que para $Lt(I)$ existen 6 monomios estándar, los cuales serían: $1, x_1, x_2, x_2^2, x_1x_2$ y $x_1x_2^2$.

Ejemplo 3.1.53. Para un ideal $J \subseteq \mathbb{Q}[x_1, x_2, x_3]$ se tiene que $Lt(J) = \langle x_1, x_2^3, x_3^4 \rangle$, de donde se puede decir que existen 12 monomios estándar: $1, x_2, x_2^2, x_3, x_3^2, x_3^3, x_2x_3$, $x_2x_3^2, x_2x_3^3, x_2^2x_3, x_2^2x_3^2$ y $x_2^2x_3^3$.

Ejemplo 3.1.54. Si $Lt(I) = \langle x_1^4, x_2x_3^2 \rangle$ es el ideal inicial de un ideal $I \subseteq \mathbb{Q}[x_1, x_2, x_3]$, el número de monomios estándar es infinito porque no aparece ninguna potencia de x_2, es decir, $1, x_2, x_2^2, x_2^3, x_2^4, \ldots$.

Teorema 3.1.55. *Sea I un ideal del anillo $F[x_1, \ldots, x_n]$. Los siguientes enunciados son equivalentes para un conjunto de polinomios $G = \{g_1, \ldots, g_s\} \subseteq I$:*

1. *G es una Base de Gröbner para I*
2. *$f \in I$ si y solo si $f \overset{G_+}{\to} 0$*
3. *$f \in I$ si y sólo si $f = \sum_{i=1}^{s} h_i g_i$ con $lp(g) = \underset{1 \leq i \leq s}{\text{máx}}(lp(h_i)lp(g_i))$*
4. *$Lt(I) = Lt(G)$*

Para su demostración ver [10].

Corolario 3.1.56. *Si $G = \{g_1, \ldots, g_s\}$ es una Base de Gröbner para I, entonces $I = \langle g_1, \ldots, g_s \rangle$.*

Dado un conjunto de generadores de un ideal $I \subseteq F[x_1, \ldots, x_n]$, existe un algoritmo, desarrollado por Bruno Buchberger, que produce una base de Gröbner para I. Este algoritmo se describe a continuación, pero primero se expondrá una definición que juega un papel importante en esta teoría:

Definición 3.1.57. *Sean $f \neq 0, g \neq 0 \in F[x_1, \ldots, x_n]$. Sea $\prec$ un orden de monomios fijo y sean $lt(f) = a_\alpha x^\alpha$ y $lt(g) = b_\beta x^\beta$ con $a_\alpha, b_\beta \in F$. Sea x^γ el mínimo común múltiplo de x^α y x^β. Se define el S-polinomio de f y g, denotado $S(f, g)$, como el polinomio*

$$S(f, g) = \frac{x^\gamma}{lt(f)} \cdot f - \frac{x^\gamma}{lt(g)} \cdot g$$

Ejemplo 3.1.58. Sean $f(x_1,\ldots,x_2,x_3) = x_1^2 + x_2^3 + x_2x_3 + x_3^2$ y $g(x_1,x_2,x_3) = x_1^2x_3 + x_1x_2x_3 - x_2x_3$ polinomios del anillo $\mathbb{Q}[x_1,x_2,x_3]$ bajo el orden lexicográfico. $lt(f) = x_1^2$, $lt(g) = x_1^2x_3$ y $mcm(x_1^2, x_1^2x_3) = x_1^2x_3$ Entonces

$$\begin{aligned} S(f,g) &= \frac{x_1^2x_3}{x_1^2}(x_1^2 + x_2^3 + x_2x_3 + x_3^2) - \frac{x_1^2x_3}{x_1^2x_3}(x_1^2x_3 + x_1x_2x_3 - x_2x_3) \\ &= -x_1x_2x_3 + x_2^3x_3 + x_2x_3^2 + x_2x_3 + x_3^3. \end{aligned}$$

Ejemplo 3.1.59. Sean $f = 3x^3 + y^2 + 1$ y $g = 5xy^2 - y^2 - 2$ elementos del anillo de polinomios $\mathbb{Q}[x,y]$ bajo el orden lexicográfico con $y \prec x$. Entonces $lt(f) = 3x^3$, $lt(g) = 5xy^2$ y $mcm(x^3, xy^2) = x^3y^2$. Entonces

$$\begin{aligned} S(f,g) &= \frac{x^3y^2}{3x^3}(3x^3 + y^2 + 1) - \frac{x^3y^2}{5xy^2}(5xy^2 - y^2 - 2) \\ &= \frac{x^2y^2}{5} + \frac{2x^2}{5} + \frac{y^4}{3} + \frac{y^2}{3} \end{aligned}$$

La importancia del $S-polinomio$ es la gran utilidad que produce al cancelar los términos líderes con respecto a un ordenamiento de términos.

Definición 3.1.60. *El algoritmo de Buchberger para encontrar una base de Gröbner de un ideal $I \subseteq \mathbb{F}[x_1,\ldots,x_n]$ con $\mathbb{F}[x_1,\ldots,x_n]$ el anillo de polinomios sobre un cuerpo $\mathbb{F}$, y $G = \{f_1,\ldots,f_s\}$ un conjunto generador de I, es el que se muestra en el* ***Algoritmo 2****.*

INICIO

Paso 1: Entrada $G := \{f_1, \ldots, f_s\}$

Paso 2: $M = \{\{f_i, f_j\} \mid f_i, f_j \in G, i \neq j\}$

Paso 3: Mientras $M \neq \varnothing$ haga

Seleccione cualquier $\{f, g\} \in M$

$M = M - \{\{f, g\}\}$

$S(f, g) \xrightarrow{g+} h$

Si $h \neq 0$ entonces

$M = M \cup \{\{h, u\}\ \forall u \in G\}$

$G = G \cup \{h\}$

Paso 4: Retorne G

FIN

Algoritmo 2: Algoritmo de Buchberger para Bases de Gröbner

Ejemplo 3.1.61. Sea $I = \langle G \rangle$, donde $G = \{f_1 = x^2 + y, f_2 = 3x^2 - 5xy + y^2, f_3 = xy + y^2\}$ donde $f_1, f_2, f_3 \in \mathbb{Q}[x, y]$ bajo el orden lexicográfico con $y < x$. Encontremos una base de Gröbner para este ideal.

INICIO

Paso 2: $M = \{\{f_1, f_2\}, \{f_1, f_3\}, \{f_2, f_3\}\}$

Paso 3:

3.1. Como $M = \{\{f_1, f_2\}, \{f_1, f_3\}, \{f_2, f_3\}\} \neq \varnothing$ entonces

seleccionamos $\{f_1, f_2\}$

$M = M - \{\{f_1, f_2\}\} = \{\{f_1, f_3\}, \{f_2, f_3\}\}$

$S(f_1, f_2) = \frac{5xy}{3} - \frac{y^2}{3} + y$

$h = -2y^2 + y$

Como $h \neq 0$ entonces

$f_4 = h$

$M = \{\{f_1, f_3\}, \{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}\}$

$G = \{f_1, f_2, f_3, f_4\}$

3.2. Como $M = \{\{f_1, f_3\}, \{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}\} \neq \varnothing$ entonces:

seleccionamos $\{f_1, f_3\}$

$M = M - \{\{f_1, f_3\}\} = \{\{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}\}$

$S(f_1, f_3) = -xy^2 + y^2$

$h = \frac{3y}{4}$

Como $h \neq 0$ entonces

$f_5 = h$

$M = \{\{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\}$

$G = \{f_1, f_2, f_3, f_4, f_5\}$

3.3. Como $M = \{\{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$ entonces seleccionamos $\{f_2, f_3\}$

$M = M - \{\{f_2, f_3\}\} = \{\{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\}$

$S(f_2, f_3) = -\frac{8xy^2}{3} + \frac{y^3}{3}$

$h = 0$

3.4. Como $M = \{\{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$ entonces seleccionamos $\{f_1, f_4\}$,

$M = M - \{\{f_1, f_4\}\} = \{\{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\}$

$S(f_1, f_4) = \frac{x^2y}{2} + y^3$

$h = 0$

3.5. Como $M = \{\{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$ entonces seleccionamos $\{f_2, f_4\}$

$M = M - \{\{f_2, f_4\}\} = \{\{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\}$

$S(f_2, f_4) = \frac{x^2y}{2} - \frac{5xy^3}{3} + \frac{y^4}{3}$

$h = 0$

3.6. Como $M = \{\{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$

entonces seleccionamos $\{f_3, f_4\}$

$M = M - \{\{f_3, f_4\}\} = \{\{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\}$

$S(f_3, f_4) = \frac{xy}{2} + y^3$

$h = 0$

3.7. Como $M = \{\{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$ entonces

seleccionamos $\{f_1, f_5\}$

$M = M - \{\{f_1, f_5\}\} = \{\{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\}$

$S(f_1, f_5) = y^2$

$h = 0$

3.8. Como $M = \{\{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$ entonces

seleccionamos $\{f_2, f_5\}$

$M = M - \{\{f_2, f_5\}\} = \{\{f_3, f_5\}, \{f_4, f_5\}\}$

$S(f_2, f_5) = -\frac{5xy^2}{3} + \frac{y^3}{3}$

$h = 0$

3.9. Como $M = \{\{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$ entonces

seleccionamos $\{f_3, f_5\}$

$M = M - \{\{f_3, f_5\}\} = \{\{f_4, f_5\}\}$

$S(f_3, f_5) = y^2$

$h = 0$

3.10. Como $M = \{\{f_4, f_5\}\} \neq \varnothing$ entonces

seleccionamos $\{f_4, f_5\}$

$M = M - \{\{f_4, f_5\}\} = \varnothing$

$S(f_4, f_5) = -\frac{y}{2}$

$h = 0$

3.11. $M = \varnothing$

Paso 4: Entonces $G = \{f_1, f_2, f_3, f_4, f_5\}$

FIN

Luego una base de Gröbner para este ideal es el conjunto $\{x^2 + y, 3x^2 - 5xy + y^2, xy + y^2, -2y^2 + y, \frac{3y}{4}\}$

Cabe anotar aquí que las bases de Gröbner obtenidas por el algoritmo de Buchberger pueden variar porque en su procedimiento interfieren el orden en el cual fueron introducidos los polinomios que generan el ideal (y que afectan el residuo hallado por el algoritmo de división) y la elección de la pareja $\{f_i, f_j\}$ para hallar el S-polinomio de f_i, f_j. Se expondrá esto con un ejemplo.

Desarrollemos el ejercicio anterior pero con los generadores del ideal en diferente orden:

Ejemplo 3.1.62. Sea $I = \langle G \rangle$, con $G = \{f_1 = 3x^2 - 5xy + y^2, f_2 = xy + y^2, f_3 = x^2 + y\}$ donde $f_1, f_2, f_3 \in \mathbb{Q}[x, y]$ bajo el orden lexicográfico con $y < x$. Encontremos una base de Gröbner para este ideal.

INICIO

Paso 2: $M = \{\{f_1, f_2\}, \{f_1, f_3\}, \{f_2, f_3\}\}$

Paso 3:

3.1. Como $M = \{\{f_1, f_2\}, \{f_1, f_3\}, \{f_2, f_3\}\} \neq \varnothing$ entonces

Seleccionamos $\{f_1, f_2\}$

$M = M - \{\{f_1, f_2\}\} = \{\{f_1, f_3\}, \{f_2, f_3\}\}$

$S(f_1, f_2) = -\frac{8xy^2}{3} + \frac{y^3}{3}$

$h = 3y^3$

Como $h \neq 0$ entonces

$f_4 = h$

$M = \{\{f_1, f_3\}, \{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}\}$

$G = \{f_1, f_2, f_3, f_4\}$

3.2. Como $M = \{\{f_1, f_3\}, \{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}\} \neq \varnothing$

entonces seleccionamos $\{f_1, f_3\}$

$M = M - \{\{f_1, f_3\}\} = \{\{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}\}$

$S(f_1, f_3) = -\frac{5xy}{3} + \frac{y^2}{3} - y$

$h = 2y^2 - y$

Como $h \neq 0$ entonces

$f_5 = h$

$M = \{\{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\},$
$\{f_3, f_5\}, \{f_4, f_5\}\}$

$G = \{f_1, f_2, f_3, f_4, f_5\}$

3.3. Como $M = \{\{f_2, f_3\}, \{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\},$
$\{f_3, f_5\}, \{f_4, f_5\}\} \neq \varnothing$ entonces seleccionamos $\{f_2, f_3\}$

$M = M - \{\{f_2, f_3\}\} = \{\{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\},$
$\{f_3, f_5\}, \{f_4, f_5\}\}$

$S(f_2, f_3) = xy^2 - y^2$

$h = -1/2y$

Como $h \neq 0$ entonces

$f_6 = h$

$M = \{\{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\},$

$\{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$

$G = \{f_1, f_2, f_3, f_4, f_5, f_6\}$

3.4. Como $M = \{\{f_1, f_4\}, \{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\},$
$\{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$
entonces seleccionamos $\{f_1, f_4\}$
$M = M - \{f_1, f_4\} = \{\{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\},$
$\{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$
$S(f_1, f_4) = -\frac{5xy^4}{3} + \frac{y^5}{3}$
$h = 0$

3.5. Como $M = \{\{f_2, f_4\}, \{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\},$
$\{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$
entonces seleccionamos $\{f_2, f_4\}$
$M = M - \{\{f_2, f_4\}\} = \{\{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\},$
$\{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$
$S(f_2, f_4) = y^4$
$h = 0$

3.6. Como $M = \{\{f_3, f_4\}, \{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\},$
$\{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$ entonces seleccionamos $\{f_3, f_4\}$
$M = M - \{\{f_3, f_4\}\} = \{\{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}, \{f_1, f_6\},$
$\{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$
$S(f_3, f_4) = y^4$
$h = 0$

3.7. Como $M = \{\{f_1, f_5\}, \{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\},$

$\{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$ entonces seleccionamos $\{f_1, f_5\}$

$M = M - \{\{f_1, f_5\}\} = \{\{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\},$
$\{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$

$S(f_1, f_5) = \frac{x^2y}{2} - \frac{5xy^3}{3} + \frac{y^4}{3}$

3.8. Como $M = \{\{f_2, f_5\}, \{f_3, f_5\}, \{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\},$
$\{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$ entonces seleccionamos $\{f_2, f_5\}$

$M = M - \{\{f_2, f_5\}\} = \{\{f_3, f_5\}, \{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\},$
$\{f_4, f_6\}, \{f_5, f_6\}\}$

$S(f_2, f_5) = \frac{xy}{2} + y^3$

$h = 0$

3.9. Como $M = \{\{f_3, f_5\}, \{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\},$
$\{f_5, f_6\}\} \neq \varnothing$ entonces seleccionamos $\{f_3, f_5\}$

$M = M - \{\{f_3, f_5\}\} = \{\{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\},$
$\{f_5, f_6\}\}$

$S(f_3, f_5) = \frac{x^2y}{2} + y^3$

$h = 0$

3.10. Como $M = \{\{f_4, f_5\}, \{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$
entonces seleccionamos $\{f_4, f_5\}$

$M = M - \{\{f_4, f_5\}\} = \{\{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$

$S(f_4, f_5) = \frac{y^2}{2}$

$h = 0$

3.11. Como $M = \{\{f_1, f_6\}, \{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$
entonces seleccionamos $\{f_1, f_6\}$

$M = M - \{\{f_1, f_6\}\} = \{\{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$

$S(f_1, f_6) = -\frac{5xy^2}{3} + \frac{y^3}{3}$

$h = 0$

3.12. Como $M = \{\{f_2, f_6\}, \{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$ entonces

Seleccionamos $\{f_2, f_6\}$

$M = M - \{\{f_2, f_6\}\} = \{\{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\}$

$S(f_2, f_6) = y^2$

$h = 0$

3.13. Como $M = \{\{f_3, f_6\}, \{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$ entonces

Seleccionamos $\{f_3, f_6\}$

$M = M - \{\{f_3, f_6\}\} = \{\{f_4, f_6\}, \{f_5, f_6\}\}$

$S(f_3, f_6) = y^2$

$h = 0$

3.14. Como $M = \{\{f_4, f_6\}, \{f_5, f_6\}\} \neq \varnothing$ entonces

Seleccionamos $\{f_4, f_6\}$

$M = M - \{\{f_4, f_6\}\} = \{\{f_5, f_6\}\}$

$S(f_4, f_6) = 0$

$h = 0$

3.15. Como $M = \{\{f_5, f_6\}\} \neq \varnothing$ entonces

Seleccionamos $\{f_5, f_6\}$

$M = M - \{\{f_5, f_6\}\} = \varnothing$

$S(f_5, f_6) = -\frac{y^2}{2}$

$h = 0$

Paso 4: Entonces $G = \{f_1, f_2, f_3, f_4, f_5, f_6\}$
FIN

Luego una base de Gröbner para este ideal es el conjunto $\{3x^2 - 5xy + y^2, xy + y^2, x^2 + y, 3y^3, 2y^2 - y, -\frac{y}{2}\}$

Nota: Es importante saber que una base de Gröbner con respecto a un ordenamiento de términos no es necesariamente una base de Gröbner para un ordenamiento de términos distinto. Veamos un ejemplo:

Ejemplo 3.1.63. Sea $I = \langle G \rangle$, con $G = \{f_1 = x^2 + y^5, f_2 = y^3x^3 - xy\}$ donde $f_1, f_2 \in \mathbb{Q}[x, y]$. Una base de Gröbner para el ideal I:

- Bajo el orden lexicográfico con $y < x$ es $\{x^2 + y^5, x^3y^3 - xy, xy^8 + xy, y^{13} + y^6\}$.
- Bajo el orden lexicográfico total con $y < x$ es $\{y^5 + x^2, x^3y^3 - xy, x^5 + xy^3\}$.

Existen diferentes tipos de bases de Gröbner para un ideal, los cuales definiermos a continuación:

Definición 3.1.64. *Sea $G = \{g_1, \ldots, g_s\}$ una Base de Gröbner. G se dice que es una Base de Gröbner mónica si $lc(g_i) = 1$ para $i = 1, \ldots, s$.*

Definición 3.1.65. *Una Base de Gröbner $G = \{g_1, \ldots, g_s\}$ es llamada Base de Gröbner minimal si $lc(g_i) = 1$ $\forall i$ y $lp(g_i) \nmid lp(g_j)$ $\forall i \neq j$ con $i, j = 1, \ldots, s$.*

Para obtener una base de Gröbner minimal a partir de un conjunto que ya es base de Gröbner simplemente se elimina o se remueve del conjunto aquel polinomio g_j tal que $lp(g_i) \mid lp(g_j)$ para algún $j \neq i$, con $i, j = 1, \ldots, s$. El procedimiento se muestra en el **Algoritmo 3**:

INICIO

Paso 1: Entrada $G = \{g_1, \ldots, g_s\}$

Paso 2: $F = \{\}$

Paso 3: Para i desde 1 hasta s haga

Si $lp(g_j) \nmid lp(g_i)$ para todo $j = 1, \ldots, s$ excepto i

$F = F \cup \{g_i\}$

Paso 4: Retorne F

FIN

Algoritmo 3: Algoritmo para hallar bases de Gröbner Minimal

Ejemplo 3.1.66. Sea el ideal $I = \langle x^3y - x, x^3 - y^2 \rangle \subseteq \mathbb{Q}[x, y]$ con el orden lexicográfico. Una base de Gröbner para I es $\{x^3y - x + y, x^3 - y^2, -x + y^3 + y, y^{10} + 3y^8 + 3y^6 + y^4 - y^3, y^9 + 3y^7 + 3y^5 + y^3 - y^2\}$ y una base de Göbner mónica es $\{x^3y - x + y, x^3 - y^2, x - y^3 - y, y^{10} + 3y^8 + 3y^6 + y^4 - y^3, y^9 + 3y^7 + 3y^5 + y^3 - y^2\}$. A continuación se halla una base de Gröbner minimal para este ideal.

INICIO

Paso 1: $G = \{g_1 = x^3y - x + y, g_2 = x^3 - y^2, g_3 = x - y^3 - y,$
$g_4 = y^{10} + 3y^8 + 3y^6 + y^4 - y^3, g_5 = y^9 + 3y^7 + 3y^5 + y^3 - y^2\}$

Paso 2: $F = \{\}$

Paso 3:

3.1. Como $i = 1 \leq 5$ entonces

como $lp(g_2) = x^3 \mid lp(g_1) = x^3y$ entonces

$F = \{\}$

3.2. Como $i = 2 \leq 5$ entonces

como $lp(g_3) = x \mid lp(g_2) = x^3$ entonces

$F = \{\}$

3.3. Como $i = 3 \leq 5$ entonces

como $lp(g_1) = x^3y \nmid lp(g_3) = x$, $lp(g_2) = x^3 \nmid lp(g_3) = x$,

$lp(g_4) = y^{10} \nmid lp(g_3) = x$ y $lp(g_5) = y^9 \nmid lp(g_3) = x$ entonces

$F = F \cup \{g_3\} = \{x - y^3 - y\}$

3.4. Como $i = 4 \leq 5$ entonces

Como $lp(g_5) = y^9 \mid lp(g_4) = y^10$ entonces

$F = \{x - y^3 - y\}$

3.5. Como $i = 5 \leq 5$ entonces

como $lp(g_1) = x^3y \nmid lp(g_5) = y^9$, y $lp(g_2) = x^3 \nmid lp(g_5) = y^9$,

$lp(g_3) = x \nmid lp(g_5) = y^9$ y $lp(g_4) = y^10 \nmid lp(g_5) = y^9$ entonces

$F = F \cup \{g_5\} = \{x - y^3 - y, y^9 + 3y^7 + 3y^5 + y^3 - y^2\}$

Paso 4: Entonces $F = \{x - y^3 - y, y^9 + 3y^7 + 3y^5 + y^3 - y^2\}$

FIN

Luego una base de Gröbner minimal para ideal $I = \langle x^3y - x, x^3 - y^2\rangle$ es $\{x - y^3 - y, y^9 + 3y^7 + 3y^5 + y^3 - y^2\}$.

Nota: Tenga en cuenta que un ideal no necesariamente tiene una única base de Gröbner minimal, porque para la obtención de ésta se pueden escoger diferentes elementos para removerlos de la base.

Ejemplo 3.1.67. Sea $I = \langle x^2 + xy + y^2, x + y, x\rangle \subseteq \mathbb{Q}[x, y]$ bajo el orden lexicográfico con $y < x$. Una base de Gröbner para I es $\{x^2 + xy + y^2, x + y, x, y^2, y\}$. Por otro lado,$\{x, y\}$, $\{y, x + y\}$ son dos bases de Gröbner minimales diferentes para este ideal.

Definición 3.1.68. *Una Base de Gröbner $G = \{g_1, \ldots, g_s\}$ se llama Base de Gröbner reducida si $lc(g_i) = 1$ y g_i es reducida con respecto a $G - \{g_i\}$ $\forall i = 1, \ldots, s$. Esto es, para todo i no hay ningún término en g_i divisible por algún $lp(g_j)$ con $j \neq i$.*

Nota: Observar que toda base de Gröbner reducida es una base de Gröbner minimal.

Para obtener una base de Gröbner reducida, reducimos completamente todo polinomio g_i con respecto a $G - \{g_i\}$ y reemplazamos g_i por su residuo, en el caso que este no sea cero.

Ejemplo 3.1.69. Sea el ideal $I = \langle 3x^2 - 5xy + y^2, xy + y^2, x^2 + y \rangle \subseteq \mathbb{Q}[x, y]$ con el orden lexicográfico. Entonces:

- Una base de Gröbner para este conjunto es $\{3x^2-5xy+y^2, xy+y^2, x^2+y, 3y^3, 2y^2-y, \frac{y}{2}\}$.
- Una base de Gröbner mónica que corresponde es $\{x^2 - \frac{5}{3}xy + y^2, xy + y^2, x^2 + y, y^3, y^2 - \frac{y}{2}, y\}$.
- Una base de Gröbner minimal puede ser $\{x^2 + y, y\}$
- Al reducir completamente esta base (en este caso, al hallar la forma normal de $x^2 + y$ por y), la base de Gröbner reducida es $\{y, x^2\}$.

Una vez que se tiene los preliminares entendidos, se procede con el algoritmo de Laubenbacher y Stigler.

3.2. Algoritmo Laubenbacher-Stigler

Dado un sistema biológico, como por ejemplo una red reguladora de genes, se tiene la información de los estados de los n genes en m tiempos consecutivos. Esto es, en el tiempo $t_j, j = 0, 1, \ldots, m-1$ el gen i, representado por la variable $x_i, i = 1, 2, \ldots, n$

adquiere el estado x_{ji}. Queremos encontrar entonces un PSD $f : \mathbb{X}^n \longrightarrow \mathbb{X}^n$ tal que nos muestre la transición de estados de cada uno de los genes en los m tiempos.

Por lo tanto, dada una sucesión S de m n-tuplas $s_0, s_1, \ldots, s_{m-1}$, donde $s_k = (s_{k1}, s_{k2}, \ldots, s_{kn})$ representa el estado de n genes en los tiempos 0 hasta $m-1$, el problema de ingeniería reversa consiste en encontrar una función f para la cual

$$f(s_0) = s_1, f(s_1) = s_2, \ldots, f(s_{m-2}) = s_{m-1}$$

lo que es equivalente a encontrar n funciones de transición f_i tales que $f_i(s_j) = s_{j+1,i}$ para $i = 1, \ldots n, j = 0, 1, \ldots, m$.

Ante este problema, R. Laubenbacher y B. Stigler en [9] plantean un algoritmo para resolverlo. La estructura se basa en un sistema dinámico polinomial determinístico en tiempo discreto con un conjunto finito de estados. La decisión de hacerlo sobre un modelo polinomial se basa en que hay teoría bien desarrollada, que provee herramientas matemáticas para la solución de éste en tiempo polinomial, además que el ruido en las mediciones de las redes permiten distinguir sólo un número pequeño de estados ([9]).

Para encontrar dicho polinomio, existen varios métodos que interpolan los datos. En particular, los autores utilizan la fórmula basada en el teorema chino de los residuos, el cual se enuncia a continuación:

Teorema 3.2.1. *Suponga $n_1, n_2, \ldots, n_k$ enteros positivos los cuales son coprimos dos a dos y sean $a_1, a_2, \ldots, a_k$ enteros. Entonces el sistema de congruencias simultáneas*

$$\begin{aligned} x &\equiv a_1 (mod\ n_1) \\ x &\equiv a_2 (mod\ n_2) \\ &\vdots \\ x &\equiv a_k (mod\ n_k) \end{aligned}$$

tiene una única solución $x\ mod(n_1 n_2 \cdots n_k)$ *donde* $0 \leq x \leq n_1 n_2 \cdots n_k$.

Para su demostración, ver [4].

Además, este teorema también tiene su generalización a anillos arbitrarios:

Teorema 3.2.2. *Si* R *es un anillo conmutativo e* $I_1, I_2, \ldots, I_k$ *ideales de* R *los cuales son coprimos dos a dos (esto es* $I_i + I_j = R$ *para* $i \neq j$*), entonces el producto* I *de esos ideales es igual a su intersección y el anillo cociente* R/I *es isomórfico al producto de los anillos* $R/I_1 \times R/I_2 \times \cdots \times R/I_k$ *via el isomorfismo*

$$f : R/I \longrightarrow R/I_1 \times R/I_2 \times \cdots \times R/I_k$$

tal que

$$f(x + I) = (x + I_1, x + I_2, \ldots, x + I_k)$$

para todo $x \in R$.

Como se enunciaba anteriormente, el Teorema Chino de los Residuos se aplica en la teoría de interpolación estableciendo una equivalencia entre la forma de interpolación de Lagrange de la siguiente manera:

Sean $x_1, x_2, \ldots, x_k$ elementos distintos de un cuerpo $\mathbb{F}$ y sean $y_1, y_2, \ldots, y_k \in \mathbb{F}$. El sistema de congruencias polinomiales

$$\begin{aligned} P(x) &\equiv y_1\ mod(x - x_1) \\ P(x) &\equiv y_2\ mod(x - x_2) \\ &\vdots \\ P(x) &\equiv y_k\ mod(x - x_k) \end{aligned}$$

tiene una única solución $mod(x-x_1)(x-x_2)\cdots(x-x_k)$. En particular, este tiene una única solución de grado $k-1$.

Note que la condición $P(x) \equiv y_i \; mod(x-x_i)$ es exactamente equivalente a la condición que $P(x_i) = y_i$ por el teorema del residuo.

En [9] enuncian que a partir del teorema chino de los residuos obtienen que para $i = 1, \ldots, n$ se va a encontrar el polinomio de interpolación f_i de la siguiente manera:

$$f_i(x) = \sum_{j=0}^{m-1} a_j r_j(\mathbf{x})$$

donde $a_j = s_{j+1,i}$ y los polinomios r_j están definidos a continuación:
Sea $0 \leq k \neq j \leq m-1$.

- Si $s_k \neq s_j$, encuentre la primera coordenada l en el cual ellos difieren. Defina

  $$b_{kj}(\mathbf{x}) = (s_{j,l} - s_{k,l})^{q-2}(x_l - s_{k,l})$$

  para todo $k \neq j$. Entonces para $k \neq j$,

  $$r_j = \prod_{k=0}^{m-1} b_{kj}(\mathbf{x})$$

- Si $s_k = s_j$ entonces se ha encontrado un ciclo límite y se debe restringir las series de tiempo a los estados $s_0, s_1, \ldots, s_{min\{k,j\}}$.

Notese que $r_j(s_j) = 1$ y $r_j(s_k) = 0$ con $k \neq j$.

Ejemplo 3.2.3. Consideremos las series de tiempo $s_0 = (5,3)$, $s_1 = (5,4)$, $s_2 = (4,3), s_3 = (3,2)$. Entonces se quiere encontrar un polinomio

$$f = (f_1, f_2) : \mathbb{Z}_7^2 \longrightarrow \mathbb{Z}_7^2$$

tal que $f(5,3) = (5,4), f(5,4) = (4,3)$ y $f(4,3) = (3,2)$. Utilizando la fórmula anterior entonces:

Para $0 \leq k \neq j \leq 2$ se tiene que:

- $b_{01} = (s_{1,l} - s_{0,l})^5(x_l - s_{0,l}) = (s_{1,2} - s_{0,2})^5(x_2 - s_{0,2}) = (4-3)^5(x_2 - 3) = (x_2 - 3)$
- $b_{02} = (s_{2,l} - s_{0,l})^5(x_l - s_{0,l}) = (s_{2,1} - s_{0,1})^5(x_1 - s_{0,1}) = (4-5)^5(x_1 - 5) = -(x_1 - 5)$
- $b_{10} = (s_{0,l} - s_{1,l})^5(x_l - s_{1,l}) = (s_{0,2} - s_{1,2})^5(x_2 - s_{1,2}) = (3-4)^5(x_2 - 4) = -(x_2 - 4)$
- $b_{12} = (s_{2,l} - s_{1,l})^5(x_l - s_{1,l}) = (s_{2,1} - s_{1,1})^5(x_1 - s_{1,1}) = (4-5)^5(x_1 - 5) = -(x_1 - 5)$
- $b_{20} = (s_{0,l} - s_{2,l})^5(x_l - s_{2,l}) = (s_{0,1} - s_{2,1})^5(x_1 - s_{2,1}) = (5-4)^5(x_1 - 4) = (x_1 - 4)$
- $b_{21} = (s_{1,l} - s_{2,l})^5(x_l - s_{2,l}) = (s_{1,1} - s_{2,1})^5(x_1 - s_{2,1}) = (5-4)^5(x_2 - 3) = (x_1 - 4)$

Ahora para $0 \leq j \leq 2$ obtenemos que los r_j son:

- $r_0 = \prod_{k=1}^{2} b_{k0} = b_{10} \times b_{20} = -(x_2 - 4)(x_1 - 4)$
- $r_1 = \prod_{k=0,k\neq 1}^{2} b_{k1} = b_{01} \times b_{21} = (x_2 - 3)(x_1 - 4)$
- $r_2 = \prod_{k=0}^{1} b_{k2} = b_{02} \times b_{12} = -(x_1 - 5)^2$

Por lo tanto, $f_1 = \sum_{j=0}^{2} a_j r_j(x) = a_0 r_0 + a_1 r_1 + a_2 r_2$ donde $a_0 = 5, a_1 = 4$ y $a_2 = 3$. Así

$$f_1(x_1, x_2) = -5(x_2 - 4)(x_1 - 4) + 4(x_2 - 3)(x_1 - 4) + 3(x_1 - 5)^2$$

y $f_2 = \sum_{j=0}^{2} a_j r_j(x) = a_0 r_0 + a_1 r_1 + a_2 r_2$ donde $a_0 = 4, a_1 = 3$ y $a_2 = 2$, esto es

$$f_2(x_1, x_2) = 4(x_2 - 4)(x_1 - 4) + 3(x_2 - 3)(x_1 - 4) + 2(x_1 - 5)^2$$

Para mirar la complejidad de esta fórmula, se puede observar que para hallar f_i se requiere m operaciones al calcular los polinomios r_j, quienes a su vez requieren m productos de polinomios b_k. Por lo tanto, se requiere un orden de $O(m^2)$ operaciones computar cada f_i. Como i varía entre 1 y n, para computar todos los f_i se requiere un orden de $O(nm^2)$ operaciones.

Otra manera de encontrar la función que interpole los datos puede ser utilizando la versión multivariada de la fórmula de interpolación de Lagrange:

$$f_i(\mathbf{x}) = \sum_{k=0}^{m-1} f_i(s_k) Q_k(\mathbf{x})$$

donde

$$Q_k(\mathbf{x}) = \prod_{j=0, j\neq k}^{m-1} \frac{x_{l_{jk}} - a_{jl_{jk}}}{a_{kl_{jk}} - a_{jl_{jk}}}$$

y l_{jk} es la primera componente en la cual s_j y s_k difieren.

Ejemplo 3.2.4. Vamos a hallar el PDS que genera los datos del ejemplo anterior utilizando la versión multivariada de polinomios de interpolación de Lagrange.

Para $0 \leq k \leq 2$ se tiene que:

$$
\begin{aligned}
Q_0(\mathbf{x}) &= \prod_{j=1}^{3} \frac{x_{l_{j0}} - a_{jl_{j0}}}{a_{0l_{j0}} - a_{jl_{j0}}} \\
&= \frac{x_{l_{10}} - a_{1l_{10}}}{a_{0l_{10}} - a_{1l_{10}}} \times \frac{x_{l_{20}} - a_{2l_{20}}}{a_{0l_{20}} - a_{2l_{20}}} \times \frac{x_{l_{30}} - a_{3l_{30}}}{a_{0l_{30}} - a_{3l_{30}}} \\
&= \frac{x_2 - a_{12}}{a_{02} - a_{12}} \times \frac{x_1 - a_{21}}{a_{01} - a_{21}} \times \frac{x_1 - a_{31}}{a_{01} - a_{31}} \\
&= \frac{x_2 - 4}{3 - 4}\frac{x_1 - 4}{5 - 4}\frac{x_1 - 3}{5 - 3} = -\frac{(x_2 - 4)(x_1 - 4)(x_1 - 3)}{2} \\
Q_1(\mathbf{x}) &= \prod_{j=0, j\neq 1}^{3} \frac{x_{l_{j1}} - a_{jl_{j1}}}{a_{kl_{j_1}} - a_{jl_{j1}}} \\
&= \frac{x_{l_{01}} - a_{0l_{01}}}{a_{1l_{01}} - a_{0l_{01}}} \times \frac{x_{l_{21}} - a_{2l_{21}}}{a_{1l_{21}} - a_{2l_{21}}} \times \frac{x_{l_{31}} - a_{3l_{31}}}{a_{1l_{31}} - a_{3l_{31}}} \\
&= \frac{x_2 - a_{02}}{a_{12} - a_{02}} \times \frac{x_1 - a_{21}}{a_{11} - a_{21}} \times \frac{x_1 - a_{31}}{a_{11} - a_{31}} \\
&= \frac{x_2 - 3}{4 - 3}\frac{x_1 - 4}{5 - 4}\frac{x_1 - 3}{5 - 3} = \frac{(x_2 - 3)(x_1 - 4)(x_1 - 3)}{2} \\
Q_2(\mathbf{x}) &= \prod_{j=0, j\neq 2}^{3} \frac{x_{l_{j2}} - a_{jl_{j2}}}{a_{kl_{j2}} - a_{jl_{j2}}} \\
&= \frac{x_{l_{02}} - a_{0l_{02}}}{a_{2l_{02}} - a_{0l_{02}}} \times \frac{x_{l_{12}} - a_{1l_{12}}}{a_{2l_{12}} - a_{1l_{12}}} \times \frac{x_{l_{32}} - a_{3l_{32}}}{a_{2l_{32}} - a_{3l_{32}}} \\
&= \frac{x_1 - a_{01}}{a_{21} - a_{01}} \times \frac{x_1 - a_{11}}{a_{21} - a_{11}} \times \frac{x_1 - a_{31}}{a_{21} - a_{31}} \\
&= \frac{x_1 - 5}{4 - 5}\frac{x_1 - 5}{4 - 5}\frac{x_1 - 3}{4 - 3} = (x_1 - 5)^2(x_1 - 3)
\end{aligned}
$$

Por lo tanto para $i = 1, 2$ se tiene que $f_i(\mathbf{x}) = \sum_{k=0}^{m-1} f_i(s_k)Q_k$, esto es

$$f_1(\mathbf{x}) = -\frac{5}{2}(x_2 - 4)(x_1 - 4)(x_1 - 3) + \frac{4}{2}(x_2 - 3)(x_1 - 1)(x_1 - 3) + 3(x_1 - 5)^2(x_1 - 3)$$

y

$$f_2(\mathbf{x}) = -\frac{4}{2}(x_2 - 4)(x_1 - 4)(x_1 - 3) + \frac{3}{2}(x_2 - 3)(x_1 - 1)(x_1 - 3) + 2(x_1 - 5)^2(x_1 - 3)$$

La complejidad de este algoritmo es también de orden $O(nm^2)$ como se puede ver en [11].

Observación: Para ambas maneras de hallar el polinomio que interpole los datos vistas anteriormente, el índice l_{jk} puede ser escogido como el índice de *cualquier* componente en el cual s_k y s_j difieren.

Una vez encontrado un polinomio que interpole las series de tiempo, que vamos a denominar f_0, o lo que los autores llaman "una solución particular de la fórmula" (porque el ejemplo anterior nos muestra que ese polinomio no es único), queremos encontrar todas las funciones polinomiales que se ajusten a los datos.

Para llevar a cabo este procedimiento, primero vamos a considerar dos polinomios $f, g \in \mathbb{F}[x_1, \ldots, x_n]$ tales que

$$f(s_j) = s_{j+1} = g(s_j)$$

. Entonces $(f - g)(s_j) = 0$ para todo j. Esto es, cualesquiera dos funciones difieren por una nueva función que tiene ceros en los puntos dados en la serie S. Por lo tanto, para encontrar todas los polinomios que se ajustan a los datos, necesitamos encontrar todas las funciones que se anulan en las series dadas. Tal conjunto de funciones es cerrado bajo la suma y multiplicación por cualquier polinomio en $\mathbb{F}[x_1, \ldots, x_n]$ y así forma un ideal.

Tal ideal está compuesto por todas las funciones que son idénticamente 0 en los puntos $s_i, i = 0, \ldots, m - 1$. Para encontrar dicho ideal definimos I_i como el ideal de todos los polinomios que toman el valor 0 en el punto s_i. Es fácil ver que I_i contiene al ideal

$$\langle x_1 - s_{i1}, x_2 - s_{i2}, \ldots, x_n - s_{in} \rangle$$

Este ideal es maximal con respecto a la inclusión, así que es igual a I_i. Entonces el ideal

I que estamos buscando es igual a la intersección de todos los I_i, esto es

$$I = \bigcap_{i=0}^{m} I_i$$

Una vez hallado el ideal, se desea reducir el polinomio f_0 encontrado anteriormente con respecto al ideal I, esto es simplemente hallar el residuo de f_0 bajo la división de la base de Gröbner reducida de I. En otros términos lo que se quiere es expresar a $f_0 = f + g$ con $g \in I$ y donde f es mínimo en el sentido que f no puede expresarse como $f' + g'$ con $g' \in I$. Y así, tal f es entonces el sistema polinomial que estábamos buscando.

El procedimiento de Laubenbacher y Stigler, entonces se puede resumir en el **Algoritmo 4**.

ENTRADA Una serie de tiempo $s_0, s_1, \ldots, s_{m-1} \in X^n$ que describen los estados de una red.

SALIDA Una función polinomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ tal que $f(s_j) = s_{j+1}$ y que f no contenga componentes polinomiales que se desvanecen en la serie de tiempo.

INICIO

PASO 1: Construya una solución particular f_0 tal que genere la data.

PASO 2: Compute el ideal I de todas las funciones que se desvanecen en la data.

PASO 3: Compute la reducción f de f_0 respecto a I.

FIN

Algoritmo 4: Algoritmo de Ingeniería Reversa por Laubenbacher y Stigler

Ejemplo 3.2.5. Se desea simular una red con 5 genes de la cual se posee la siguiente información: $s_0 = (0, 2, 1, 0, 0), s_1 = (2, 1, 0, 1, 0), s_2 = (0, 0, 0, 1, 1)$,
$s_3 = (2, 1, 1, 0, 2), s_4 = (2, 1, 0, 1, 2), s_5 = (1, 0, 0, 0, 1), s_6 = (2, 0, 2, 0, 2)$.

Entonces el modelo a encontrar va a ser una función $f = (f_1, f_2, f_3, f_4, f_5) : \mathbb{Z}_3^5 \longrightarrow \mathbb{Z}_3^5$

que simule la data anterior.

Siguiendo el algoritmo de Laubenbacher y Stigler, las soluciones particulares son: $f_1 = -2x_1^5 - 2x_1^3x_3^2 + 2x_1^3x_3x_5 + x_1^4 + 2x_1^2x_3^2 - 2x_1^3x_5 - 2x_1^2x_3x_5 - 2x_1^3 + 2x_1^2x_5 - 2x_1^2 + 2$

$f_1 = -x_1^3x_3^2 + 2x_1^4 + x_1^2x_3^2 + x_1^3 + x_1^2 + 1$

$f_1 = -2x_1^5 - x_1^4x_2 - x_1^4 + 2x_1^3x_2 + 2x_1^3 + 2x_1^2x_2 + 2x_1^2 + 2x_2 + 1$

$f_1 = x_1^4x_2 - x_1^3x_3^2 - 2x_1^3x_3x_5 - 2x_1^3x_2 - x_1^3x_3 + x_1^2x_3^2 + 2x_1^3x_5 + 2x_1^2x_3x_5 + x_1^3 - 2x_1^2x_2 + x_1^2x_3 - 2x_1^2x_5 - x_1^2 - 2x_2$

$f_1 = -2x_1^5 - 2x_1^4x_2 - 2x_1^3x_3^2 + x_1^4 - x_1^3x_2 - x_1^3x_3 + 2x_1^2x_3^2 - x_1^3 - x_1^2x_2 + x_1^2x_3 + 2x_1^2 - x_2 + 2$

Luego, los ideales que se desvanecen en la data son:

$I_1 = \langle x_5^2 + x_1 + 2x_2 + x_3 + x_4 - 2x_5, x_4x_5 - 2x_1 - x_2 - x_3 + 2x_4 - x_5 - 2, x_3x_5 + 2x_1 + x_2 + 2x_4 - 2, x_2x_5 + x_1 + 2x_2 + x_3 + x_4 - x_5, x_1x_5 + 2x_1 - x_2 - 2x_3 - 2x_4 - 2x_5 - 1, x_4^2 - x_4, x_3x_4, x_2x_4 + x_1 + 2x_2 + 2x_3 + x_4 - 1, x_1x_4 + 2x_1 - x_2 - x_3 + 2x_4 - 2, x_3^2 - x_3, x_2x_3 - x_1 + 2x_2 - 2x_3 - x_4 + 1, x_1x_3 + 2x_1 + x_2 + 2x_4 - 2, x_2^2 - 2x_1 - 2x_2 - 2x_3 - 2x_4 + 2, x_1x_2 - x_1 - x_3 - x_4 + 1, x_1^2 - 2x_1 - x_3 - x_4 + 1\rangle$

$I_2 = \langle\ x_5^2 + x_1 + 2x_2 + x_3 + x_4 - 2x_5, x_4x_5 - 2x_1 - x_2 - x_3 + 2x_4 - x_5 - 2, x_3x_5 + 2x_1 + x_2 + 2x_4 - 2, x_2x_5 + x_1 + 2x_2 + x_3 + x_4 - x_5, x_1x_5 + 2x_1 - x_2 - 2x_3 - 2x_4 - 2x_5 - 1, x_4^2 - x_4, x_3x_4, x_2x_4 + x_1 + 2x_2 + 2x_3 + x_4 - 1, x_1x_4 + 2x_1 - x_2 - x_3 + 2x_4 - 2, x_3^2 - x_3, x_2x_3 - x_1 + 2x_2 - 2x_3 - x_4 + 1, x_1x_3 + 2x_1 + x_2 + 2x_4 - 2, x_2^2 - 2x_1 - 2x_2 - 2x_3 - 2x_4 + 2, x_1x_2 - x_1 - x_3 - x_4 + 1, x_1^2 - 2x_1 - x_3 - x_4 + 1\rangle$

$I_3 = \langle x_5^2 + x_1 + 2x_2 + x_3 + x_4 - 2x_5, x_4x_5 - 2x_1 - x_2 - x_3 + 2x_4 - x_5 - 2, x_3x_5 + 2x_1 + x_2 + 2x_4 - 2, x_2x_5 + x_1 + 2x_2 + x_3 + x_4 - x_5, x_1x_5 + 2x_1 - x_2 - 2x_3 - 2x_4 - 2x_5 - 1, x_4^2 - x_4, x_3x_4, x_2x_4 + x_1 + 2x_2 + 2x_3 + x_4 - 1, x_1x_4 + 2x_1 - x_2 - x_3 + 2x_4 - 2, x_3^2 - x_3, x_2x_3 - x_1 + 2x_2 - 2x_3 - x_4 + 1, x_1x_3 + 2x_1 + x_2 + 2x_4 - 2, x_2^2 - 2x_1 - 2x_2 - 2x_3 - 2x_4 + 2, x_1x_2 - x_1 - x_3 - x_4 + 1, x_1^2 - 2x_1 - x_3 - x_4 + 1\rangle$

$I_4 = \langle x_5^2 + x_1 + 2x_2 + x_3 + x_4 - 2x_5, x_4x_5 - 2x_1 - x_2 - x_3 + 2x_4 - x_5 - 2, x_3x_5 + 2x_1 + x_2 + 2x_4 -$

$2, x_2x_5+x_1+2x_2+x_3+x_4-x_5, x_1x_5+2x_1-x_2-2x_3-2x_4-2x_5-1, x_4^2-x_4, x_3x_4, x_2x_4+x_1+$
$2x_2+2x_3+x_4-1, x_1x_4+2x_1-x_2-x_3+2x_4-2, x_3^2-x_3, x_2x_3-x_1+2x_2-2x_3-x_4+1, x_1x_3+$
$2x_1+x_2+2x_4-2, x_2^2-2x_1-2x_2-2x_3-2x_4+2, x_1x_2-x_1-x_3-x_4+1, x_1^2-2x_1-x_3-x_4+1\rangle$

$I_5 = \langle x_5^2+x_1+2x_2+x_3+x_4-2x_5, x_4x_5-2x_1-x_2-x_3+2x_4-x_5-2, x_3x_5+2x_1+x_2+2x_4-$
$2, x_2x_5+x_1+2x_2+x_3+x_4-x_5, x_1x_5+2x_1-x_2-2x_3-2x_4-2x_5-1, x_4^2-x_4, x_3x_4, x_2x_4+x_1+$
$2x_2+2x_3+x_4-1, x_1x_4+2x_1-x_2-x_3+2x_4-2, x_3^2-x_3, x_2x_3-x_1+2x_2-2x_3-x_4+1, x_1x_3+$
$2x_1+x_2+2x_4-2, x_2^2-2x_1-2x_2-2x_3-2x_4+2, x_1x_2-x_1-x_3-x_4+1, x_1^2-2x_1-x_3-x_4+1\rangle.$

Y luego de hacer la de la solución particular respecto al ideal, el PDS resultante es:

$$f_1 = -2x_1^5 - 2x_1^3x_3^2 + 2x_1^3x_3x_5 + x_1^4 + 2x_1^2x_3^2 - 2x_1^3x_5 - 2x_1^2x_3x_5 - 2x_1^3 + 2x_1^2x_5 - 2x_1^2 + 2,$$

$$f_2 = -x_1^3x_3^2 + 2x_1^4 + x_1^2x_3^2 + x_1^3 + x_1^2 + 1,$$

$$f_3 = -2x_1^5 - x_1^4x_2 - x_1^4 + 2x_1^3x_2 + 2x_1^3 + 2x_1^2x_2 + 2x_1^2 + 2x_2 + 1,$$

$$f_4 = x_1^4x_2 - x_1^3x_3^2 - 2x_1^3x_3x_5 - 2x_1^3x_2 - x_1^3x_3 + x_1^2x_3^2 + 2x_1^3x_5 + 2x_1^2x_3x_5 + x_1^3 - 2x_1^2x_2 + x_1^2x_3 - 2x_1^2x_5 - x_1^2 - 2x_2,$$

$$f_5 = -2x_1^5 - 2x_1^4x_2 - 2x_1^3x_3^2 + x_1^4 - x_1^3x_2 - x_1^3x_3 + 2x_1^2x_3^2 - x_1^3 - x_1^2x_2 + x_1^2x_3 + 2x_1^2 - x_2 + 2$$

Capítulo 4

Variables no redundates

En las redes reguladoras de genes, se puede determinar que en el estado de un gen i no interfieren los estados de otros genes. Esto es, en el modelo a encontrar, que el polinomio f_i contiene variables, que vamos a llamar redundantes, que no van a interferir en los valores que ésta vaya a adquirir.

El objetivo en este capítulo es tratar de evitar la aparición de aquellas variables redundantes que no son posibles eliminar con simplificaciones sencillas. Para esto, se encontrarán conjuntos de variables que no contengan dichas variables utilizando el método desarrollado por Sasao, el cual se explicará mas adelante.

4.1. Definiciones

Definición 4.1.1. *Una función* $f : F^n \longrightarrow Q$ *se dice que* ***depende de*** x_i *si existen* $a, b \in F^n, a = (a_1, \ldots, a_i, \ldots, a_n), b = (a_1, \ldots, b_i, \ldots, a_n)$ *con* $a_i \neq b_i$ *tal que* $f(a)$ *y* $f(b)$ *existen y* $f(a) \neq f(b)$.

Ejemplo 4.1.2. Considere la siguiente tabla que representa $f : \mathbb{Z}_3^4 \longrightarrow \mathbb{Z}_3$

x_1	x_2	x_3	x_4	f
2	2	1	0	0
0	1	1	1	0
1	2	1	1	0
2	2	1	1	1
0	0	0	1	1
0	2	1	0	1
0	2	1	1	2

Entonces f depende de x_1 porque $(2, 2, 1, 0)$ y $(0, 2, 1, 0)$ difieren en la primera componte y $f(2, 2, 1, 0) = 0 \neq 1 = f(0, 2, 1, 0)$. Similarmente, f depende de x_2 y x_4.

Ejemplo 4.1.3. Sea $g : \mathbb{Z}_3^6 \longrightarrow \mathbb{Z}_5$ dado por:

x_1	x_2	x_3	x_4	x_5	x_6	g
0	1	1	2	2	1	0
0	2	1	2	2	1	1
2	2	1	2	2	1	2
0	1	2	2	2	1	3

Entonces g depende de x_1, x_2, x_3 y x_4.

Nota: Si f depende de x_i entonces x_i es **esencial** en f y x_i debe aparecer en cualquier expresión para f.

Definición 4.1.4. *Una variable x_i se dice **redundante** si f se puede escribir sin usar x_i.*

Ejemplo 4.1.5. Sea $f = \mathbb{Z}_3^7 :\longrightarrow \mathbb{Z}_3$, donde f está dada por:

x_1	x_2	x_3	x_4	x_5	x_6	x_7	f
1	1	2	1	1	0	1	0
0	2	1	1	1	2	1	1
1	1	0	2	1	0	2	2

Entonces f puede estar dada por $f = x_1^2 - x_1x_3 + 1$. Aquí x_3 es redundante porque f también puede escribirse como $f = -x_4 - x_6 + 1$.

Definición 4.1.6. *Sea R un conjunto y n un entero positivo. Sea $U = \{x_1, x_2, \ldots, x_n\}$ un conjunto de n variables. Para cualquier conjunto $D = \{x_{i_1}, x_{i_2}, \ldots, x_{i_m}\} \subset U$ y cualquier $r = (r_1, \ldots, r_n) \in R^n$ se define la proyección de r sobre D como $proy_D r = (r_{i_1}, \ldots, r_{i_m})$.*

Ejemplo 4.1.7. Al trabajar sobre $\mathbb{Z}_7[x_1, \ldots, x_5]$, y considerando $D = \{x_2, x_3, x_4\}$ y $r = (2, 5, 4, 6, 0)$, entonces se tiene que $proy_D r = (5, 4, 6)$.

Definición 4.1.8. *Un conjunto $\mathcal{X} = \{x_{i_1}, \ldots, x_{i_m}\}$ se denomina el conjunto de variables no redundantes para f si:*

1. *Para todo $c, d \in D$, $proy_{\mathcal{X}} c = proy_{\mathcal{X}} d$ entonces $f(c) = f(d)$.*

2. *No hay otro conjunto de variables que contenga a $\mathcal{X}$ con esta propiedad.*

Ejemplo 4.1.9. Sea f la función representada por la siguiente tabla:

x_1	x_2	x_3	f
2	1	0	0
2	0	0	0
0	1	1	0
1	1	2	1
1	2	2	1
0	1	0	1

Entonces $\mathcal{X} = \{x_1, x_3\}$ es un conjunto de variables no redundates porque $proy_{\mathcal{X}}(2, 1, 0) = proy_{\mathcal{X}}(2, 0, 0)$ y $f(2, 1, 0) = 0 = f(2, 0, 0)$, similar con $(1, 1, 2)$ y $(1, 2, 2)$. Además

- $\mathcal{X} = \{x_1, x_2\}$ no es un conjunto de variables no redundates porque aunque $proy_{\mathcal{X}}(0, 1, 1) = proy_{\mathcal{X}}(0, 1, 0)$, $f(0, 1, 1) = 0 \neq 1 = f(0, 1, 0)$.

- $\mathcal{X} = \{x_2, x_3\}$ no es un conjunto de variables no redundates porque aunque $proy_{\mathcal{X}}(2,1,0) = proy_{\mathcal{X}}(0,1,0)$, $f(2,1,0) = 0 \neq 1 = f(0,1,0)$.
- $\mathcal{X} = \{x_1, x_2, x_3\}$ no es un conjunto de variables no redundates porque no cumple con la propiedad (1).

Definición 4.1.10. *Una variable $x_i \notin \mathcal{X}$ es una variable redundante (con respecto a $\mathcal{X}$), y una variable x_j es irredundante si x_j no es redundante.*

A continuación se muestra un resultado muy importante obtenido en [11].

Lema 4.1.11. *Si x_i es una variable esencial y $\mathcal{X}$ es un conjunto de variables no redundates, entonces $x_i \in \mathcal{X}$.*

Note que toda variable esencial pertenece al conjunto de variables no redundates, pero no todo elemento de este conjunto es esencial. En otras palabras, toda variable esencial es irredundante, pero no lo contrario. Se demostrará mas adelante con un ejemplo.

Ahora, si $f : D \longrightarrow B$, para todo i en $f(D)$ se denotará por $r(i) = \{a \in D | f(a) = i\}$.

Ejemplo 4.1.12. Considere la siguiente tabla:

x_1	x_2	x_3	x_4	x_5	f
0	0	0	0	1	0
1	1	0	2	1	0
0	2	2	1	1	0
1	0	2	2	1	0
1	1	0	2	1	1
1	0	1	0	0	1
2	2	0	0	2	1
1	0	1	2	2	1
0	0	0	1	1	2
2	0	0	0	1	2
1	0	1	0	1	2
2	2	2	2	2	2

entonces

$$r(0) = \{(0,0,0,0,1),(1,1,0,2,1),(0,2,2,1,1),(1,0,2,2,1)\}$$
$$r(1) = \{(1,1,0,2,1),(1,0,1,0,0),(2,2,0,0,2),(1,0,1,2,2)\}$$
$$r(2) = \{(0,0,0,1,1),(2,0,0,0,1),(1,0,1,0,1),(2,2,2,2,2)\}$$

Definición 4.1.13. *Para cada $r(i)$ y $r(j)$ definidas anteriormente donde $i \neq j$, se define $s(i,j)$ como la disyunción de las variables en el que cada pareja ordenada (u,v) de $r(i) \times r(j)$ difieren. Esto es para $i \neq j$*

$$s(i,j) = \bigwedge_{u \in r(i)} \bigvee_{v \in r(j)} \{x_m | u_m \neq v_m\}$$

Ejemplo 4.1.14. Sea $r(0) = (0,1,2,0)$ y $r(1) = (3,2,0,0)$ entonces $s(0,1) = x_1 \vee x_2 \vee x_3$ porque esas son las variables en el que difieren.

Ejemplo 4.1.15. Considere la siguiente tabla que describe a la función g:

x_1	x_2	x_3	g
0	2	1	0
3	0	0	0
1	2	2	0
2	0	1	1
0	0	2	1
0	1	1	2

entonces

$$r(0) = \{(0,2,1),(3,0,0),(1,2,2)\}$$
$$r(1) = \{(2,0,1),(0,0,2)\}$$
$$r(2) = \{(0,1,1)\}$$

por lo tanto

$$s(0,1) = (x_1 \vee x_2) \wedge (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_3) \wedge (x_1 \vee x_3) \wedge (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2)$$
$$s(0,2) = (x_2) \wedge (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee x_3)$$
$$s(1,2) = (x_1 \vee x_2) \wedge (x_1 \vee x_2 \vee x_3)$$

Definición 4.1.16. *Un sentencia se dice que está en* ***forma normal disyuntiva*** *si es una disyunción (sucesión de O's -$\bigvee$-); es decir, que consiste de una o mas disyunciones cada una de las cuales es una conjunción de uno o mas sentencias.*

Ejemplo 4.1.17. Algunos ejemplos de forma normal disyuntiva son:

- A

- $A \vee B$
- $A \wedge B$
- $A \vee (B \wedge C)$
- $(A \wedge B) \vee (B \wedge \neg C)$

Ejemplo 4.1.18. Si se tiene $(x_2 \vee x_3) \wedge (x_2 \vee x_3 \vee x_4) \wedge (x_1)$ entonces al expresarlo en forma normal disyuntiva, queda:

$$\begin{aligned}(x_2 \vee x_3) \wedge (x_2 \vee x_3 \vee x_4) \wedge (x_1) &= (x_2 \vee x_3 \vee x_4) \wedge (x_1) \\ &= (x_2 \wedge x_1) \vee (x_2 \wedge x_1) \vee (x_4 \wedge x_1)\end{aligned}$$

Nota: Por facilidad expresaremos $a \wedge b$ como ab.

Ejemplo 4.1.19. Expresar en forma normal disyuntiva $x_3(x_2 \vee x_5)(x_1 \vee x_5)$

$$\begin{aligned}x_3(x_2 \vee x_5)(x_1 \vee x_5) &= x_3\{x_2(x_1 \vee x_5) \vee x_5(x_1 \vee x_5)\} \\ &= x_3\{x_1x_2 \vee x_2x_5 \vee x_1x_5 \vee x_5\} \\ &= x_1x_2x_3 \vee x_2x_3x_5 \vee x_1x_3x_5 \vee x_3x_5 \\ &= x_1x_2x_5 \vee x_3x_5\end{aligned}$$

4.2. Algoritmo para encontrar conjuntos de variables no redundates

Considerando a f como una función en múltiples variables, Sasao en [12] propone un algoritmo para minimizar las variables dependientes de dicha función, esto es, expresar a f con el mínimo número de variables. Para esto, el autor utiliza una extensión del método de Halatsis-Gaitanis desarrollado en [7].

El procedimiento que propone se puede resumir en el **Algoritmo 5**.

ENTRADA Una función multivariable $f : D \subset P^n \longrightarrow Q$

SALIDA Los conjuntos $\mathcal{X}$ de variables tal que $\mathcal{X}$ contiene variables no redundantes para f

INICIO

PASO 1: Encuentre los $r(i)$ para $i \in f(D)$.

PASO 2: Halle $R = \bigwedge s(i,j)$ para todo $i \neq j$.

PASO 3: Exprese a R en forma normal disyuntiva.

FIN

Algoritmo 5: Algoritmo para encontrar los conjuntos de varibales no redundantes para f

Ejemplo 4.2.1. Sea f la función dada por la siguiente tabla:

x_1	x_2	x_3	x_4	f
0	0	1	1	0
1	0	1	1	0
2	1	1	1	1
0	1	1	1	1
0	1	2	2	2
0	2	0	1	2

Se quiere encontrar los conjuntos que no contienen variables redundantes para esta función. Aplicando el algoritmo de Sasao obtenemos:

INICIO

Paso 1: Los $r(i)$ para f son

$$r(0) = \{(0,0,1,1),(1,0,1,1)\}$$
$$r(1) = \{(2,1,1,1),(0,1,1,1)\}$$
$$r(2) = \{(0,1,2,2),(0,2,0,1)\}$$

Paso 2:

$s(0,1) = (x_1 \vee x_2)(x_2)(x_1 \vee x_2)(x_1 \vee x_2)$

$s(0,2) = (x_2 \vee x_3 \vee x_4)(x_2 \vee x_4)(x_2 \vee x_3)(x_1 \vee x_2 \vee x_3 \vee x_4)(x_1 \vee x_2 \vee x_3)$

$s(1,2) = (x_1 \vee x_3 \vee x_4)(x_1 \vee x_2 \vee x_3)(x_3 \vee x_4)e(x_2 \vee x_3)$ Así R es:

$R = (x_1 \vee x_2)(x_2)(x_3 \vee x_4)(x_2 \vee x_3)(x_1 \vee x_2)(x_1 \vee x_2)(x_1 \vee x_2 \vee x_3 \vee x_4)$

$(x_1 \vee x_2 \vee x_3)(x_1 \vee x_3 \vee x_4)(x_1 \vee x_2 \vee x_3)(x_3 \vee x_4)(x_2 \vee x_3)$

que se puede expresar más simplificado como

$R = (x_1 \vee x_2)(x_2)(x_3 \vee x_4)(x_2 \vee x_3)(x_1 \vee x_2 \vee x_3 \vee x_4)(x_1 \vee x_2 \vee x_3)(x_1 \vee x_3 \vee x_4)$.

Paso 3: Expresando a R en su forma normal disyuntiva queda que $R = x_2x_3 \vee x_2x_4$.

FIN

Por lo tanto hay dos conjuntos de variables no redundantes: $\mathcal{X}_1 = \{x_2, x_3\}$,
$\mathcal{X}_2 = \{x_2, x_4\}$.

Capítulo 5

Resultados

Una vez se ha recorrido por los capítulos anteriores, podemos dar una solución al siguiente problema:

Dada una red reguladora de genes, o un sistema biológico en general, a la cual se le conoce información de los estados de los genes a través del tiempo y además se posee información de cómo un gen en específico interfiere en los siguientes estados de otro, se quiere encontrar el modelo que cumpla dichas condiciones y que mas se ajuste a la red.

Para esto se realizará ingeniería reversa, encontrando un sistema dinámico polinomial determinístico en tiempo discreto sobre un conjunto finito de estados. Tal modelo se encontrará haciendo una modificación al algoritmo de Laubenbacher y Stigler y utilizando el resultado de [11]. Dicho procedimiento se discutirá en la primera sección de este capítulo y su implementación computacional, en la segunda.

5.1. Solución

En la red reguladora de genes a la cual se le quiere encontrar un modelo, vamos a considerar:

- n el número de genes de la red.
- $f : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$, un PDS, el modelo a encontrar que simule la red.
- El comportamiento de cada gen i será modelado por $f_i(x_1, \ldots, x_n)$.
- f_i asumirá valores en $\mathbb{F}_q^n$, donde $\mathbb{F}_q^n$ es el conjunto (finito) de estados posibles de los genes.
- En el tiempo j, el estado de los n genes es $s_j = (s_{j1}, \ldots, s_{jn})$ con $s_i \neq s_j$.
- m el número de mediciones que se tienen de la red.
- Las m medidas consecutivas de los estados de los genes serán simbolizadas por $s_0, s_1, \ldots, s_{m-1}$.
- I el gen que va a influir sobre los estados del gen J.

Es bien importante que el PDS f cumpla con que

$$f(s_0) = s_1, f(s_1) = s_2, \ldots, f(s_{m-2}) = s_{m-1}$$

o equivalente a que

$$f_i(s_j) = s_{j+1,i} \text{ para } i = 1, \ldots n, j = 0, 1, \ldots, m-1$$

además que f_J dependa de x_I, esto es, que f_J se escriba, entre otros, en términos de x_I.

Para cumplir con el objetivo, se realizó una modificación al algoritmo de Laubenbacher y Stigler, utilizando también los conceptos de bases mínimas y teoría de eliminación.

Este procedimiento se describe en el **Algoritmo 6**.

ENTRADA Una serie de tiempo $s_0, s_1, \ldots, s_{m-1} \in X^n$ que describen los estados de una red y las variables I, J donde I es el gen que influye en el estado del gen J.

SALIDA Una función polinomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ tal que $f(s_j) = s_{j+1}$ y que f no contenga componentes polinomiales que se desvanecen en la serie de tiempo y que f_J contenga x_I.

INICIO

PASO 1: Usando la versión del Algoritmo de Sasao, desarrollada por D. Bollman y O. Orozco podemos producir un conjunto $\mathcal{X}$ que no tenga variables redundantes.

Paso 2: Usando $\mathcal{X}$ y el Algoritmo del Teorema Chino de los residuos podemos encontrar una solución particular f_0 que interpole los datos en términos de x_J y las variables $\mathcal{X}$

PASO 3: Computar el ideal I de todos las soluciones que se desvanescan en los datos.

PASO 4: Usando teoría de eliminación podemos obtener la reducción f de f_0 con respecto a $I \cap F_q[\mathcal{X}, x_j]$.

FIN

Algoritmo 6: Solución al problema de ingeniería reversa con conocimiento previo

Ejemplo 5.1.1. Sea una red reguladora con 4 genes, donde los estados de ellos pueden asumir 5 valores donde se tienen las siguientes 8 mediciones

$$s_0 = (0, 0, 1, 1),$$
$$s_1 = (1, 2, 3, 2),$$
$$s_2 = (3, 2, 1, 3),$$
$$s_3 = (4, 1, 1, 4),$$
$$s_4 = (1, 4, 2, 0),$$
$$s_5 = (0, 4, 0, 1),$$
$$s_6 = (1, 4, 1, 0),$$
$$s_7 = (0, 1, 1, 1)$$

y se conoce además que los estados del gen 2 interfieren en los estados del gen 3. Entonces se quiere encontrar un modelo que la simule.

El PDS a encontrar es uno de la forma $f = (f_1, f_2, f_3, f_4) : \mathbb{Z}_5^4 \longrightarrow \mathbb{Z}_5^4$ tal que para f_1, f_2, f_3 y f_4 se tiene que

x_1	x_2	x_3	x_4	f_1	f_2	f_3	f_4
0	0	1	1	1	2	3	2
1	2	3	2	3	2	1	3
3	2	1	3	4	1	1	4
4	1	1	4	1	4	2	0
1	4	2	0	0	4	0	1
0	4	0	1	1	4	1	2
1	4	1	0	0	1	1	1

Si aplicamos el algoritmo de Stigler para hallar el sistema dinámico que mas se ajusta a las mediciones es

- $f_1 = 2x_4^2 - x_1 - 2x_2 + 2x_3 + 2x_4$,
- $f_2 = -2x_4^2 - x_1 - 2x_2 + x_4 - 2$,
- $f_3 = -2x_4^2 - x_1 + 2x_3 + x_4 + 2$,
- $f_4 = x_4^2 + x_1 + 2x_3 - 2x_4 + 1$

pero no cumple con la condición que el gen 2 interfiere en los estados del gen 3, esto es, f_3 no aparece en términos de x_2, por lo tanto, aplicamos nuestro algoritmo para f_3:

Aplicando el algoritmo de Sasao para hallar las bases mínimas de f_3 se obtiene que:

- Para f_3 las posibles bases son $\mathcal{X}_1 = \{x_2, x_3\}, \mathcal{X}_2 = \{x_1, x_3\}, \mathcal{X}_3 = \{x_1, x_2\}$.
  $\mathcal{X}_4 = \{x_3, x_4\}$

Usando la base $\mathcal{X}_1 = \{x_2, x_3\}$ que es una que contiene la variable x_2 que nos interesa, se halla un polinomio particular:

$$f_3 = 2x_2^6 - x_2^4x_3^2 - x_2^5 + x_2^4x_3 - 2x_2^4 + 2x_2^2x_3^2 + 2x_2^3 - 2x_2^2x_3 - x_2x_3^2 + x_2^2 + x_2x_3 + 2x_2 - 2$$

Además, se calcula el ideal I que se desvanezca en las mediciones dadas

$$\begin{aligned} I = \langle & x_2x_4 - 2x_3x_4 - x_4^2 + x_2 + 2x_4 + 1, x_1\,x_4 - x_3\,x_4 + x_4^2 + 2x_1 + x_2 + x_4 - 1, \\ & x_3^2 + 2x_3\,x_4 + 2x_3 - 2x_4 + 2, x_2\,x_3 - x_3\,x_4 + x_1 + x_3 + x_4 - 1, \\ & x_1\,x_3 + 2x_3\,x_4 - 2x_1 - x_2 - x_3 - 2x_4 + 1, x_2^2 + x_3\,x_4 - x_4^2 - x_1\,, \\ & x_1\,x_2 + x_4^2 + x_1 - x_4\,, x_1^2 + 2x_3\,x_4 - x_4^2 + 2x_1 - 2x_2 - x_4\,, \\ & x_4^3 - x_3\,x_4 + x_4^2 + x_2 - 2x_4 + 1, x_3\,x_4^2 - x_4^2 - x_1 - x_2\rangle \end{aligned}$$

Haciendo $I \cap \mathbb{Z}_5[x_2, x_3]$, el ideal resultante que obtenemos es
$\langle -x_2\,x_3^2 + 2x_3^3 + 2x_2\,x_3 - 2x_3^2 - x_2 + x_3 - 1, -x_2^2\,x_3 - x_3^3 + x_2^2 + 2x_2\,x_3 - 2x_3^2 - 2x_2 + x_3 + 2, x_3^3 - x_2\,x_3 + 2x_3^2 + x_2 + x_3 + 1, -x_2^4 + 2x_2^3 + x_2^2 - 2x_2\rangle$

y por último, si hacemos la reducción normal de f_3 respecto a este nuevo ideal, se obtiene que el nuevo f_3 es
$f_3 = -2x_2^3 + x_2^2 - 2x_2x_3 + 2x_3^2 + 2x_2 + x_3$.

Por lo tanto, el PDS que modela la data y cumple con la condición que x_2 aparece en f_3 es:

- $f_1 = 2x_4^2 - x_1 - 2x_2 + 2x_3 + 2x_4$,
- $f_2 = -2x_4^2 - x_1 - 2x_2 + x_4 - 2$,
- $f_3 = -2x_2^3 + x_2^2 - 2x_2x_3 + 2x_3^2 + 2x_2 + x_3$
- $f_4 = x_4^2 + x_1 + 2x_3 - 2x_4 + 1$

Ejemplo 5.1.2. Se obtienen las siguientes medidas de una red que posee 7 genes:

Tiempo	Gen 1	Gen 2	Gen 3	Gen 4	Gen 5	Gen 6	Gen 7
0	0	0	1	2	3	0	0
1	1	2	3	0	0	1	1
2	1	1	1	1	1	1	1
3	2	1	3	4	0	0	1
4	3	4	1	2	4	0	4
5	2	2	2	2	1	1	1
6	0	0	0	0	1	1	1

y se sabe que los genes 1,2 y 3 interfiere en los estados de los genes 4 y 5, por lo tanto se desea hallar un PDS que mejor modele esta red y que cumpla con tales condiciones.

Se encontrará un PDS $f = (f_1, \ldots, f_1) : \mathbb{Z}_5^7 \longrightarrow \mathbb{Z}_5^7$ tal que f interpole los datos.

- Se utiliza el Algoritmo de Laubenbacher y Stigler para encontrar el siguiente sistema dinámico:

- $f_1 = 2x_3 + x_4 - x_5 + 2x_6 - 2x_7$
- $f_2 = -x_3 - 2x_5 + 2x_6 - 2x_7 - 1$
- $f_3 = 2x_3 + x_5 + 2x_7 - 2$
- $f_4 = -x_3 + 2x_4 - x_5 + 2x_6 + 2x_7$
- $f_5 = x_3 + x_5 + 2x_6 + 1$
- $f_6 = 2x_3 - x_4 - x_5 + 2x_6 - x_7 - 1$
- $f_7 = 2x_3 - 2x_4 + x_5 - x_6 + x_7$

pero no cumple con la condición que los genes 1, 2 y 3 no interfieren en los estados de los genes 4 y 5, por tal motivo aplicamos nuestro algoritmo para f_4 y f_5

- Para f_4:
  - Utilizando el algoritmo de Sasao para hallar los conjuntos de variables que no son redundantes para f_4 se obtienen que tales conjuntos son $\mathcal{X}_1 = \{x_3, x_6, x_7\}$, $\mathcal{X}_2 = \{x_3, x_5, x_6\}$,$\mathcal{X}_3 = \{x_4, x_7\}$, $\mathcal{X}_4 = \{x_4, x_5\}$, $\mathcal{X}_5 = \{x_2, x_5\}$, $\mathcal{X}_6 = \{x_2, x_4\}$, $\mathcal{X}_7 = \{x_2, x_3\}$, $\mathcal{X}_8 = \{x_1, x_5\}$, $\mathcal{X}_9 = \{x_1, x_4\}$, $\mathcal{X}_{10} = \{x_1, x_3\}$, $\mathcal{X}_{11} = \{x_1, x_2\}$.
  - Si utilizamos el conjunto $\mathcal{X}_7$ junto con x_1 hallamos una solución particular que interpole la data en esas variables $f_4 = x_1^5 + x_1^4 + 2x_1^3x_2 + x_1^2x_2 - 2x_1^2 - x_1x_2 + 2x_1$.

- Se halla el ideal que se desvance para las mediciones obtenidas de la red:

$$I = \langle x_2 - x_3 - x_5 - x_6 - 2x_7 - 1, x_1 + x_3 - 2x_4 - x_5 - 2x_6 + 2x_7 + 1,$$
$$x_7^2 + 2x_3 - 2x_4 + x_5 + 2x_6 + 2x_7 - 1, x_6x_7 - x_6, x_5x_7 - 2x_3 + 2x_4 - 2x_5 - 2x_6 - x_7 - 1,$$
$$x_4x_7 - x_4 - 2x_7 + 2, x_3x_7 - x_3 - x_7 + 1, x_6^2 - x_6, x_5x_6 + 2x_3 - 2x_4 + 2x_6 + 2,$$
$$x_4x_6 - 2x_3 + x_4 - x_5 - x_6 - x_7 - 2, x_3x_6 + 2x_3 + 2x_4 - x_5 - x_7 + 2, x_5^2 - x_5 + x_7 - 1,$$
$$x_4x_5 - x_3 + 2x_5 - x_7 - 1, x_3x_5 + x_3 - 2x_4 - 2x_5 + 2x_6 - x_7 + 1, x_4^2 + 2x_3 - 2x_5 + x_6 - 2x_7,$$
$$x_3x_4 + 2x_4 + 2x_5 + 2x_7 - 2, x_3^2 + 2x_4 + x_5 - 2x_6 + x_7 + 2\rangle$$

- y al hacer $I \cap \mathbb{Z}_7^5[x_1, x_2, x_3]$, el ideal resultante es

$$\langle -x_3^3 - x_1x_3 - x_2x_3 + x_1 + x_2 + x_3, 2x_1x_3 + 2x_2x_3 + 2x_3^2 - 2x_1 - 2x_2 + x_3 + 2,$$
$$2x_2x_3^2 - x_1x_2 - x_2^2 - 2x_3^2 - 2x_3 - 1, x_1^2 + 2x_1x_2 + 2x_2^2 + x_2x_3 + x_3^2 - x_2 - 1,$$
$$- x_1x_2 - 2x_2^2 + 2x_2x_3 - 2x_1 - 2x_2 + 2x_3 - 2, -x_2^2 - x_2x_3 - 2x_3^2 - 2x_1 - x_2 + 2$$

- Por último, haciendo la reducción normal de f_4 con el ideal anterior se obtiene que: $f_4 = -x_2x_3 - x_3^2 - x_1 + x_2 + 2x_3 - 1$

- Para f_5:

  - Utilizando el algoritmo de Sasao para hallar los conjuntos de variables que no son redundantes para f_4 se obtienen que tales conjuntos son $\mathcal{X}_1 = \{x_3, x_6, x_7\}$, $\mathcal{X}_2 = \{x_3, x_5, x_6\}$,$\mathcal{X}_3 = \{x_4, x_7\}$, $\mathcal{X}_4 = \{x_4, x_5\}$, $\mathcal{X}_5 = \{x_2, x_5\}$, $\mathcal{X}_6 = \{x_2, x_6\}$, $\mathcal{X}_7 = \{x_2, x_4\}$, $\mathcal{X}_8 = \{x_2, x_3\}$, $\mathcal{X}_9 = \{x_1, x_5\}$, $\mathcal{X}_{10} = \{x_1, x_4\}$, $\mathcal{X}_{11} = \{x_1, x_3\}$, $\mathcal{X}_{12} = \{x_1, x_2\}$.

  - Si utilizamos el conjunto $\mathcal{X}_8$ junto con x_1 hallamos una solución particular que interpole la data en esas variables $f_5 = -2x_1^5 + x_1^4x_2 - x_1^4 + x_1^3x_2 - 2x_1^3 - x_1x_2 - x_1$.

- Se halla el ideal que se desvance para las mediciones obtenidas de la red:

  $I = \langle x_2 - x_3 - x_5 - x_6 - 2x_7 - 1, x_1 + x_3 - 2x_4 - x_5 - 2x_6 + 2x_7 + 1,$
  $x_7^2 + 2x_3 - 2x_4 + x_5 + 2x_6 + 2x_7 - 1, x_6x_7 - x_6, x_5x_7 - 2x_3 + 2x_4 - 2x_5 - 2x_6 - x_7 - 1,$
  $x_4x_7 - x_4 - 2x_7 + 2, x_3x_7 - x_3 - x_7 + 1, x_6^2 - x_6, x_5x_6 + 2x_3 - 2x_4 + 2x_6 + 2,$
  $x_4x_6 - 2x_3 + x_4 - x_5 - x_6 - x_7 - 2, x_3x_6 + 2x_3 + 2x_4 - x_5 - x_7 + 2, x_5^2 - x_5 + x_7 - 1,$
  $x_4x_5 - x_3 + 2x_5 - x_7 - 1, x_3x_5 + x_3 - 2x_4 - 2x_5 + 2x_6 - x_7 + 1, x_4^2 + 2x_3 - 2x_5 + x_6 - 2x_7,$
  $x_3x_4 + 2x_4 + 2x_5 + 2x_7 - 2, x_3^2 + 2x_4 + x_5 - 2x_6 + x_7 + 2\rangle$

- y al hacer $I \cap \mathbb{Z}_7^5[x_1, x_2, x_3]$, el ideal resultante es

  $\langle -x_3^3 - x_1x_3 - x_2x_3 + x_1 + x_2 + x_3, 2x_1x_3 + 2x_2x_3 + 2x_3^2 - 2x_1 - 2x_2 + x_3 + 2,$
  $2x_2x_3^2 - x_1x_2 - x_2^2 - 2x_3^2 - 2x_3 - 1, x_1^2 + 2x_1x_2 + 2x_2^2 + x_2x_3 + x_3^2 - x_2 - 1,$
  $- x_1x_2 - 2x_2^2 + 2x_2x_3 - 2x_1 - 2x_2 + 2x_3 - 2, -x_2^2 - x_2x_3 - 2x_3^2 - 2x_1 - x_2 + 2$

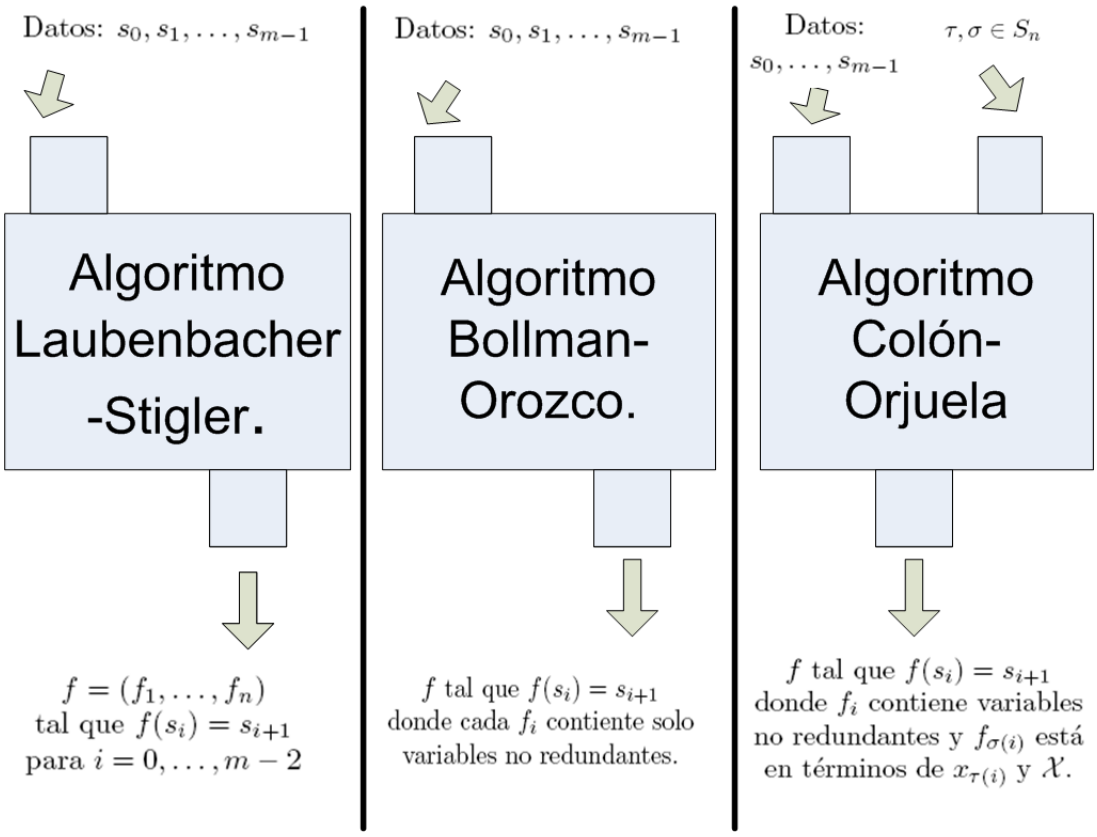
- Por último, haciendo la reducción normal de f_5 con el ideal anterior se obtiene que: $f_5 = -x_3^2 - x_1 + x_2 - x_3 + 2$

Por tal motivo, el PDS que se ajusta a los datos y cumpla con las condiciones requeridas es:

- $f_1 = 2x_3 + x_4 - x_5 + 2x_6 - 2x_7$
- $f_2 = -x_3 - 2x_5 + 2x_6 - 2x_7 - 1$
- $f_3 = 2x_3 + x_5 + 2x_7 - 2$
- $f_4 = -x_2x_3 - x_3^2 - x_1 + x_2 + 2x_3 - 1$
- $f_5 = -x_3^2 - x_1 + x_2 - x_3 + 2$
- $f_6 = 2x_3 - x_4 - x_5 + 2x_6 - x_7 - 1$
- $f_7 = 2x_3 - 2x_4 + x_5 - x_6 + x_7$

5.2. Algoritmo Colón-Orjuela vs. otros algoritmos

En el siguiente diagrama se muestra la diferencia del algoritmo propuesto con respecto al algoritmo de Laubenbaches-Stigler y el algoritmo Bollman-Orozco, éste último propuesto en [11].



Se aclarará lo anterior con un ejemplo:

Ejemplo 5.2.1. Sea una red que posee 4 nodos, cada uno de estos con 3 posibles estados, de la cual se pudieron obtener las siguientes medidas:

x_1	x_2	x_3	x_4
0	1	2	0
0	2	1	1
1	2	1	2
1	1	0	1
1	1	1	1
0	1	1	0
2	2	2	2
1	0	0	1

lo cual es equivalente a:

x_1	x_2	x_3	x_4	f_1	f_2	f_3	f_4
0	1	2	0	0	2	1	1
0	2	1	1	1	2	1	2
1	2	1	2	1	1	0	1
1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	0
0	1	1	0	0	2	2	2
2	2	2	2	1	0	0	1

además de conocer que el gen 1 y 2 interfieren en los estados del gen 2, y el gen 1 interfiere en sí mismo.

Aplicando los 3 algoritmos se obtiene que:

- Algoritmo Laubenbacher-Stigler:

Realizando el procedimiento descrito en el la Sección 3.2 se obtiene que el PDS $f : \mathbb{Z}_3^4 \longrightarrow \mathbb{Z}_3^4$ queda:

$$f_1 = x_3x_4 + x_4^2 + x_2 + x_3 - x_4$$
$$f_2 = -x_1 - 1$$
$$f_3 = x_3x_4 - x_1 - x_2 - x_3 - x_4 + 1$$
$$f_4 = -x_4^2 + x_1 - x_3 + x_4$$

Note que el conocimiento que se tenía de la influencia de unos genes sobre otros no se ve reflejado completamente.

- Algoritmo Bollman-Orozco:
  Para hallar cada f_i en términos de variables no redundantes, primero se halla lo que ellos llaman bases:

  - Para f_1 las bases son $\mathcal{X}_1 = \{x_2, x_3, x_4\}, \mathcal{X}_2 = \{x_1, x_3, x_4\}, \mathcal{X}_3 = \{x_1, x_2, x_3\}$
    Considerando cualquier conjunto de variables no redundantes, por ejemplo $\mathcal{X}_3$, se tiene que f_1 es
    $f_1 = x_1x_2x_3 + x_1x_2 + x_1x_3 - x_2x_3 + x_1 - x_2 - x_3 - 1$
  - Para f_2 las bases son $\mathcal{X}_1 = \{x_2, x_3, x_4\}, \mathcal{X}_2 = \{x_1\}$
    Considerando cualquier conjunto de variables no redundantes, por ejemplo $\mathcal{X}_2$ se obtiene que f_2 es
    $f_2 = -x_1 + 2$
  - Para f_3 las bases son $\mathcal{X}_1 = \{x_3, x_4\}, \mathcal{X}_2 = \{x_1, x_2, x_3\}$
    Considerando cualquier conjunto de variables no redundantes, por ejemplo $\mathcal{X}_3$ se tiene que f_3 es
    $f_3 = -x_3^2 - x_3x_4 - x_3 + 1$

- Para f_4 las bases son $\mathcal{X}_1 = \{x_2, x_3, x_4\}, \mathcal{X}_2 = \{x_1, x_3, x_4\}, \mathcal{X}_3 = \{x_1, x_2, x_3\}$
  Considerando cualquier conjunto de variables no redundantes, por ejemplo $\mathcal{X}_3$ se tiene que f_4 es
  $f_4 = x_1^4x_2^2x_3 - x_1^4x_2^2 + x_1^3x_2^2x_3 - x_1^3x_2^2 - x_1^4x_3 - x_1^2x_2^2x_3 + x_1^4 + x_1^2x_2^2 - x_1^3x_3 +$
  $x_1^2x_2x_3 - x_1x_2^2x_3 + x_1^3 - x_1^2x_2 + x_1x_2^2 + x_1^2x_3 - x_1^2 + x_1x_3 - x_2x_3 + x_2 - 1$

Por lo tanto, un PDS que se obtiene mediante este algoritmo puede ser:

$f_1 = x_1x_2x_3 + x_1x_2 + x_1x_3 - x_2x_3 + x_1 - x_2 - x_3 - 1$

$f_2 = -x_1 + 2$

$f_3 = -x_3^2 - x_3x_4 - x_3 + 1$

$f_4 = x_1^4x_2^2x_3 - x_1^4x_2^2 + x_1^3x_2^2x_3 - x_1^3x_2^2 - x_1^4x_3 - x_1^2x_2^2x_3 + x_1^4 + x_1^2x_2^2 - x_1^3x_3 + x_1^2x_2x_3 - x_1x_2^2x_3 + x_1^3$

pero note que por mas que se escojan otras bases, la condición de, por ejemplo, que el gen 2 influya en el gen 2 no se logra cumplir con este resultado.

- Algoritmo Colón-Orjuela:
  Realizando todos los pasos descritos en la sección 5.1, se obtiene que el PDS que mejor se ajusta a la data y que tiene en cuenta el conocimiento previo es:

$$f_1 = -x_1x_2 - x_1 + x_2 - 1$$
$$f_2 = -x_1x_2^4 - x_1x_2^3 - x_2^4 - x_2^3 + x_1x_2 + x_2$$
$$f_3 = x_3x_4 - x_1 - x_2 - x_3 - x_4 + 1$$
$$f_4 = -x_4^2 + x_1 - x_3 + x_4$$

5.3. Implementación

Los cálculos de los algoritmos nombrados anteriormente requieren mucho trabajo, razón por la cual se han implementando computacionalmente. La herramienta aquí utilizada es el programa de álgebra computacional libre denominado CoCoA cuya descarga se puede realizar del sitio web http://cocoa.dima.unige.it.

5.3.1. El Sistema Computacional CoCoA

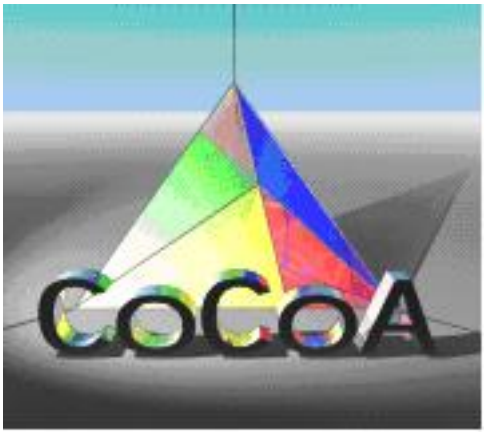


Figura 5.1: Logo de CoCoA

CoCoA, cuyas siglas significan Computational Conmutative Algebra, fue implementado desde 1983 originalmente por A. Giovini y Gianfranco Niesi, estudiantes de la Universidad de Génova, Italia. En su desarrollo, a través de los años, han contribuido gran cantidad de personas entre los que se destacan Lorenzo Robbiano, Ana Bigatti, Jhon Abbott y Massimo Caboara. Inicialmente fue elaborado en Pascal pero desde 1993 fue implementado en C para que pudiera ejecutarse sobre diversos sistemas operativos entre los que se encuentran Linux, Unix, Windowx, MacOS y Solaris.

CoCoA fue desarrollado especialmente para matemáticos con poco o nada de conocimientos en programación, con el fin de ser una herramienta útil y de poca complejidad para resolver problemas específicos en matemáticas, especialmente en álgebra, cuyos cálculos sean dispendiosos.

5.3.2. Algoritmos

A continuación se explican los algoritmos desarrollados durante este trabajo, junto con un pequeño ejemplo de entrada y la salida de cada uno de ellos. (El código fuente

se anexará al final del trabajo).

- Algoritmo Laubenbacher-Stigler

  OBJETIVO: A partir de una serie de puntos, encuentra el PDS que lo modela, aplicando el algoritmo de Ingeniería Reversa de Laubenbacher-Stigler.

  ENTRADA: Numvariables: Dimensión del espacio

  P: Número de elementos en el cuerpo finito

  Puntos: Serie de puntos

  SALIDA: La función $F = [f_1, f_2, \ldots, f_numvariables]$.

  Ejemplo 5.3.1. Se quiere simular una red de la que se tienen las siguientes mediciones:
  $s_0 = (0,0,1), s_1 = (2,0,1), s_2 = (1,1,1), s_3 = (2,1,0), s_4 = (2,0,0)$
  entonces si corremos el programa:

  Entrada: $AlgoritmoLS(3, 7, [[0,0,1],[2,0,1],[1,1,1],[2,1,0],[2,0,0]])$;
  Salida: $[-2x[1]^3-3x[1]^2x[2]+2x[1]^2+3x[1]x[2]+2, x[1]^3+3x[1]^2x[2]-3x[1]x[2], -2x[1]^3+3x[1]^2x[2]-3x[1]x[2]+x[1]+1]$

- PolinomiodePuntos

  OBJETIVO: A partir de una serie de puntos, encuentra la función que los genera.

  ENTRADA: Numvariables: Dimensión del espacio

  P: Número de elementos en el cuerpo finito

  Eses: Serie de puntos

  As: Valores funcionales de f, es decir, $f(Eses) = As$

  SALIDA: La función f_i minima.

  Ejemplo 5.3.2. Dados los datos de $f : \mathbb{Z}_7^3 \longrightarrow \mathbb{Z}_7^3$

x_1	x_2	x_3	x_4
0	0	1	2
2	0	1	1
1	1	1	2
2	1	0	2

Entrada: $PolinomiodePuntos(3, 7, [[0, 0, 1], [2, 0, 1], [1, 1, 1], [2, 1, 0]], [2, 1, 2, 2])$;

Salida: $3x[1] - 3x[2] + 3x[3] - 1$

- Funcion0

  OBJETIVO: Halla el polinomio de interpolación a partir de una serie de datos.

  ENTRADA: Numvariables: Dimensión del espacio

  Eses: Serie de puntos

  As: Valores funcionales de f, es decir, $f(Eses) = As$

  SALIDA: La función que interpola los puntos.

  Ejemplo 5.3.3. Dados los datos de $f : \mathbb{Z}_7^3 \longrightarrow \mathbb{Z}_7^3$

x_1	x_2	x_3	x_4
0	0	1	2
2	0	1	1
1	1	1	2
2	1	0	2

  Entrada: $Funcion0(3, [[0, 0, 1], [2, 0, 1], [1, 1, 1], [2, 1, 0]], [2, 1, 2, 2], 7)$

  Salida: $-2x[1]^3 - 3x[1]^2x[2] + 2x[1]^2 + 3x[1]x[2] + 2$

- QuitaSubconjuntos

  OBJETIVO: Dada una lista de conjuntos, encuentra que elementos son subconjuntos de otros en la lista y los elimina.

ENTRADA: Variables: Un Lista de Conjuntos

SALIDA: Una lista de conjuntos cuyos elementos no son subconjuntos de otros.

- BasesSasao

  OBJETIVO: Dada una función f encuentra las variables que son bases para esta.

  ENTRADA: (Puntos, N).

  Puntos: [A, B] donde $f(A) = B$, A: lista de dimensión N

  N: Dimensión del espacio

  SALIDA: Los conjuntos de variables que son bases de la función.

  Ejemplo 5.3.4. Hallar las bases de la función $g : \mathbb{Z}_3^4 \longrightarrow \mathbb{Z}_3$ descrita por

x_1	x_2	x_3	x_4	g
0	2	1	2	0
2	1	1	0	0
0	1	1	2	1
2	1	2	1	1
0	1	2	1	2
2	2	2	2	2
0	0	1	2	3
1	1	1	2	3

  Entrada: $BasesSasao([[[0,2,1,2],0],[[2,1,1,0],0],[[0,1,1,2],1],[[2,1,2,1],1],$
  ...cont...$[[0,1,2,1],2],[[2,2,2,2],2],[[0,0,1,2],3],[[1,1,1,2],3]],4)$;

  Salida: $[x[1]x[2]x[4], x[1]x[2]x[3]]$

- Interpolacionconvariables

  OBJETIVO: Halla el polinomio de interpolación a partir de una serie de datos solo en términos de las variables de la base y añadido.

ENTRADA: Variables: Todas las bases para f

Eses: Puntos a interpolar

As: Valores funcionales de f, es decir, $f(Eses) = As$

N: Dimension del espacio

Varinterfiere: Variable que se sabe que interfiere en un estado de otra

SALIDA: La función que interpola los puntos.

Ejemplo 5.3.5. Se quiere encontrar un PDS que interpole los siguientes datos, conociendo además que f depende de x_2, entonces

x_1	x_2	x_3	x_4	f
0	0	1	1	3
1	2	3	2	1
3	2	1	3	1
4	1	1	4	2
1	4	2	0	0
0	4	0	1	1

Entrada:*Interpolacionconvariables*([[2, 4], [3, 4], [2, 3]], [[0, 0, 1, 1], [1, 2, 3, 2], [3, 2, 1, 3], ...cont...[4, 1, 1, 4], [1, 4, 2, 0], [0, 4, 0, 1]], [3, 1, 1, 2, 0, 1], 4, 5, 2);

Salida: $x[2]^2 + x[4]^2 - 2x[2] + 2$

Apéndice A

Algoritmos

A continuación se encuentra el código fuente de los algoritmos mencionados en los Capítulos 4 y 5.

A.1. Algoritmo Quitasubconjuntos()

```
Define QuitaSubconjuntos(Variables)

I:=1;

while  I < Len(Variables) do
    J:=I+1;
    while J ≤ Len(Variables) do
        /* Evalua si es subcojunto para removerlo                    */
        if IsSubset(Variables[I], Variables[J]) then
            Remove(Variables,J);
        end
        else
            J:=J+1;
        end
    end
    I:=I+1;
end

Return Variables;

EndDefine;
```

Algoritmo 7: Código QuitaSubconjuntos

A.2. Algoritmo BasesSasao

```
Define BasesSasao(Puntos,N)
/* Definición de variables                                          */
FdePuntos:=[]; Rs:=[];
LenFdePuntos:=[];
Temporal:=[];
Variables:=[];
/* Saco los valores funcionales de f                                */
for I:=1 To Len(Puntos) do
|  Append(FdePuntos,Puntos[I,2]);
end
/* Quito los que se repiten                                         */
FdePuntos:=Set(FdePuntos);
/* Encuentro los puntos que tienen el mismo valor funcional         */
for I:=1 To Len(FdePuntos) do
|  Temporal:=[];
|  for J:=1 To Len(Puntos) do
|  |  if Puntos[J,2]=FdePuntos[I] then
|  |  |  Append(Temporal,Puntos[J,1]);
|  |  end
|  end
|  Append(Rs,Temporal);
|  Insert(LenFdePuntos,I,Len(Temporal));
end
```

Algoritmo 8: Código BasesSasao

```
/* CONTINUACIÓN BasesSasao                                          */
/* Para cada uno de los anteriores, encuentro la(s) componente(s)
   en la que ellos difieren es decir encuentro los s(i,j,k,l)       */
for I:=1 To Len(LenFdePuntos) do
    for J:=I+1 To Len(LenFdePuntos) do
        for (SubI:=1 To LenFdePuntos[I] do
            for SubJ:=1 To LenFdePuntos[J] do
                for K:=1 To N do
                    if Rs[I,SubI,K] <> Rs[J,SubJ,K] then
                        Append(Temporal,K);
                    end
                end
                Append(Variables,Temporal);
                Temporal:=[];
            end
        end
    end
end
/* Quito repeticiones                                               */
Variables:=Set(Variables);
/* Miro que conjuntos son subconjuntos de otros para eliminarlos    */
Variables:=QuitaSubconjuntos(Variables);
Reverse(Variables);
Variables:=QuitaSubconjuntos(Variables);
/* Defino R como la conjunción de los s(i,j,k,l) para i <> j y
   expreso a R en la forma normal disyuntiva                        */
```

```
/* CONTINUACIÓN BasesSasao                                          */
/* Producto Cartesiano                                              */
Producto:=Variables[1];
for I:=2 To Len(Variables) do
| Producto:=Producto¿¡Variables[I];
end
/* Elimino los paréntesis del producto Cartesiano                   */
Variables:=[];
for I:=1 To Len(Producto) do
| Append(Variables,Set(Flatten(Producto[I])));
end
/* elimino otra vez los subconjuntos despues del producto           */
Variables:=QuitaSubconjuntos(Variables);
Reverse(Variables);
Variables:=QuitaSubconjuntos(Variables);
//* Escogemos las variables que pertenecen a la base               */
VarVerd:=[];
Equis:=[];
for I:= 1 To N do
| Append(Equis,x[I]);
end
for I:=1 To Len(Variables) do
| Temp:=1;
| for J:=1 To Len(Variables[I]) do
| | Temp:=Temp* Equis[Variables[I,J]];
| end
| Append(VarVerd,Temp);
end
/* Retorna los conjuntos que son bases para f                       */
Return(VarVerd);
EndDefine;
```

A.3. Algoritmo PolinomiodePuntos()

```
Define PolinomiodePuntos(Numvariables,P,Eses,As)
/* Crea el polinomio f0:polinomio que interpola los datos        */
F0:=Funcion0(Numvariables,Eses, As,P);
/* Crea el ideal de polinomios que se desvanecen en los puntos   */
IdealdePuntos:=IdealOfPoints(Eses);
/* Hace la reducción del polinomio f0 con respecto al ideal      */
Fi:=NF(F0,IdealdePuntos);
Return(Fi);
EndDefine;
```

Algoritmo 9: Código PolinomiodePuntos

A.4. Algoritmo Funcion0()

```
Define Funcion0(Numvariables,Eses,As,P)
/* Declaración de variables                                    */
X:=[];
B:=[];
VectorB:=[];
Numerodatos:=Len(Eses);
/* Crea las variables                                          */
for I:=1 To Numvariables do
  | Append(X,x[I]);
end
/* Crea los bij, es decir, la primera coordenada donde difieren las
   eses                                                        */
for I:=1 To Numerodatos do
  | VectorBI:=[];
  | for J:=1 To Numerodatos do
  |   | if I ≠ J then
  |   |   | /* Halla la primera componente en que difieren los puntos
  |   |   |    */
  |   |   | Difieren:=Eses[I]-Eses[J];
  |   |   | K:=1;
  |   |   | while Difieren[K]=0 And K ≤ Numvariables do
  |   |   |   | K:=K+1;
  |   |   | end
  |   |   | if K ≤ Numvariables then
  |   |   |   | Append(VectorBI,(Eses[J,K] − Eses[I,K])^(P−2) * (X[K] −
  |   |   |   | Eses[I,K]));
  |   |   | end
  |   | end
  |   | else
  |   |   | Append(VectorBI,1);
  |   | end
  | end
  | Append(VectorB,VectorBI);
```

```
/* Crea las R                                                    */
R:=[];
for J:=1 To Numerodatos do
    Temporal:=1;
    for I:=1 To Numerodatos do
        Temporal:=Temporal*VectorB[I,J];
    end
    Append(R,Temporal);
end
/* Crea el polinomio f0                                          */
F0:=0;
for I:=1 To Numerodatos do
    F0:=F0+As[I]*R[I];
end
Return(F0);
EndDefine;
```

Algoritmo 10: Código Funcion0

A.5. Algoritmo AlgoritmoLS

```
Define AlgoritmoLS(Numvariables,P,Puntos)
/* Declaración de variables                                    */
F:=[];
Eses:=[];
As:=[];
Numerodatos:=Len(Puntos);
//Separa los puntos para cada función f_i
/* Selecciona los puntos (datos) menos el último               */
for K:=1 To Len(Puntos)-1 do
 | Append(Eses,Puntos[K]);
end
/* Separa las Ases (ed. los elementos de la coordenada i-esima de
   todos los puntos)                                           */
for I:=1 To Numvariables do
 | for J:=1 To Numerodatos-1 do
 |  | Append(As,Puntos[J+1,I]);
 | end
 | /* deduce el polinomio fi                                   */
 | Append(F,PolinomiodePuntos(Numvariables,P,Eses,As));
 | As:=[];
end
Return F;
EndDefine;
```

Algoritmo 11: Código AlgoritmoLS

Bibliografía

[1] D. BOLLMAN, O. COLÓN-REYES and E. OROZCO. Fixed Points in Discrete Models for Regulatory Genetic Networks, EURASIP Journal on Bioinformatics and Systems Biology, Vol. 2007, pp. 8, 2007.

[2] O. COLÓN-REYES, R. LAUBENBACHER and B. PAREIGIS. Boolean monomial dynamical systems. Annnals of combinatorics, 8 (2004), pp 425-439.

[3] O. COLÓN-REYES, A. S. JARRAH, R. LAUBENBACHER and B. STURMFELS. Monomial dynamical systems over finite fields. Journal of Complex Systems, vol. 16 No. 4, pp. 333-342, 2006.

[4] T.H. CORMEN, C.E. LEISERSON, R.L. RIVEST and C. STEIN. Introduction to Algorithms, Second Edition. MIT Press and McGraw-Hill, 2001.

[5] FILKOV, V., SKIENA, S., and ZHI, J., 2002. Analysis techniques for microarray time-series data. J. Comp. Biol. 9, 317-330.

[6] FRIEDMAN N., LINIA, l. N. M., NACHMAN, I., and Peér D., 2000. Using Bayensian networks to analyze expression data, J. Comp. Biol. 7, 601-620.

[7] C. HALATSIS and N. GAITANIS. Irredundant normal forms and minimal dependence set of a Boolean functions, IEEE Trans. on Computers, Vol C-27, No. 11, pp. 1064-1068, Nov 1978.

[8] HARTEMINK, A. J., GIFFORD, D. K., JAAKKOLA S., and YOUNG, R.A., 2001. Using graphical models ande genomic exprerssion data to statistically validate models of genetic regulatory networks. Pac. Symp. Biocomput., World Scientific, Singapore.

[9] R. LAUBENBACHER and B. STIGLER. A computational algebra approach to the reverse engineering of gene regulatory networks. J. Theor. Biol., 229:523-537, 2004.

[10] L. COX, L., J. LITTLE and D. O'SHEA. Ideals, Varieties and Algorithms. An Introduction to Computational Algebraic Geometry and Commutative Algebra. Third Edition. Springer. 2007

[11] E. OROZCO and D. BOLLMAN. Multivariate Polynomial Interpolation and Reverse Engineering Genetic Networks.

[12] T. SASAO. On the number of dependent variables for incompletely specifie multiple-valued functions. 30th International Symposium on multiple-valued logic, Portland, Oregon, May 23-25 (2000), 91-97.

[13] YEUNG. M. K. S., TEGNÉR, J., and COLLINS, J. J., 2002. Reverse engineering gene networks using singular value descompositioin and robust regression. Proc. Natl. Acad. Sci. 99, 6163-6168.