Verificación formal de desigualdades reales usando programación semidefinida

Eric Biagioli

2009

Verificación formal de desigualdades reales usando programación semidefinida

Eric Biagioli

Universidad Nacional de Rosario

Orientadores: Stéphane Gaubert[1] y Benjamin Werner[2]

Lugar: École Polytechnique, Palaiseau, Francia. Laboratorio de Ciencias de la Computación (LIX) y Laboratorio de Matemáticas Aplicadas (CMAP).

[1]Stephane.Gaubert@inria.fr; CMAP, Ecole Polytechnique e INRIA, equipo Maxplus
[2]Benjamin.Werner@inria.fr; LIX, Ecole Polytechnique e INRIA, equipo Typical

Índice general

Gracias!

Quiero en este trabajo darle las gracias a muchas personas. A mis abuelos Elvira y Adolfo (pinto) Ruggeri. A mis amigos de siempre: Asdrúbal, Nicolás, Sebastián. Por su apoyo incondicional y por permitirme sentir que cuento con ellos.

Muchas gracias a Pablo Lotito, por todo el apoyo y la ayuda que me brindó para ir a Francia y para el doctorado que empezaré dentro de poco. Gracias a Benjamín Werner y Stephane Gaubert por aceptarme en la Polytechnique para este trabajo, y por ofrecerme hacer doctorado con en sus equipos; gracias a Roland Zumkeller por su gran amabilidad y por todas las explicaciones y aclaraciones útiles que me brindó.

Gracias a Pablo Speciale y a Santiago Zanella por la ayuda que me dieron con Coq. Gracias a Guido Macchi, por los múltiples cafés y charlas que tuvimos en estos años.

Gracias a todos!

Capítulo 1

Introducción

Existen resultados cuyas demostraciones involucran una mezcla de razonamientos complejos con un gran número de cálculos. Un ejemplo de esto es el teorema de los cuatro colores ([4]). Este resultado dice que cualquier división del plano en regiones contiguas (como la división de un país en provincias) puede ser coloreada usando a lo sumo cuatro colores de manera que no haya dos regiones adyacentes con el mismo color. Dos regiones se consideran adyacentes si y sólo si comparten un segmento de borde, no sólo un punto; si dos regiones comparten sólo un punto, no se consideran adyacentes.

En 1852, Francis Guthrie, siendo un reciente graduado de la universidad de Londres, escribió a su hermano (aún estudiante allí) preguntándole si existía alguna demostración del hecho de que cuatro colores son suficientes para colorear adecuadamente cualquier mapa. Su hermano no supo darle la respuesta, pero preguntó a uno de sus profesores: De Morgan. De Morgan tampoco supo demostrarlo, pero preguntó a otros matemáticos y la pregunta llegó finalmente a Cayley. Casi 26 años después de la carta de Guthrie, en 1878, Cayley propuso el problema como interesante a la London Mathematical Society. Un año después un abogado de Londres, Arthur Kempe, publicó una demostración. La demostración de Kempe fue aceptada como válida durante mas de once años, hasta que Heawood encontró un fallo en el argumento que Kempe proponía.

Tres colores son suficientes para mapas simples, pero se requiere un cuarto color adicional si una región está rodeada por tres regiones adyacentes dos a dos. Para el teorema de los *cinco* colores existe una demostración corta y elemental; fue probado hacia finales del sigo XIX. La demostración del teorema de los cuatro colores mostró ser significativamente mas complicada. Un número considerable de intentos fallidos han aparecido desde la carta de Guthrie en 1852.

Kempe en su demostración define como mapa *penta* a un mapa que *exige* cinco colores. Define también el concepto de mapa *normal*. Un mapa *normal* será un mapa que verifique que (a) no hay ningún país aislado dentro de otro y (b) ningún punto de frontera es frontera de mas de tres países vecinos. Kempe demostró que si existe un mapa penta, entonces existe un mapa penta normal. Por otro lado, como cada mapa penta normal tiene un número finito de países, en caso de que exista un mapa penta normal existirá un mapa penta normal mínimo (*i.e.:* con el mínimo número posible de países). Kempe propuso un conjunto *inevitable* $\mathcal{C}$ de cuatro configuraciones para los mapas penta normales. Esto es, al menos un elemento de $\mathcal{C}$ tiene que estar incluído en cualquier mapa penta normal. Finalmente, demostró que cada uno de los cuatro elementos de $\mathcal{C}$ era *reducible*, lo que implicaba que ninguno de ellos podía estar presente en un mapa penta normal mínimo. Así, si existiera mapa penta normal mínimo se tendría una contradicción. No puede existir un mapa penta normal mínimo, y consecuentemente no puede existir un

mapa penta normal. El error de la demostración de Kempe estuvo en la prueba de la reductibilidad de una de las cuatro configuraciones del conjunto que propuso.

En 1976 Appel y Haken, con en esta misma idea y con ayuda de la computadora, dieron una demostración al teorema de los cuatro colores. Mostraron un conjunto inevitable de 1482 configuraciones, cada una de ellas reducible. Nuevamente, demostraron que

- al menos un elemento del conjunto tiene que estar incluido en cualquier mapa (el conjunto es *inevitable*) y
- ninguno puede ser parte del menor contraejemplo (*i.e.:* todas las configuraciones del conjunto son reducibles).

Appel y Haken usaron un programa *ad-hoc* y cientos de páginas de análisis *a mano* para mostrar que su conjunto satisfacía estas dos propiedades. Inicialmente, su demostración no fue aceptada por la comunidad de matemáticos porque la prueba asistida no podía ser verificada a mano por un humano. Desde entonces, la prueba ha ido ganando cada vez mayor aceptación. En 2005, Benjamin Werner y Georges Gonthier formalizaron en Coq la demostración del teorema ([4]).

Otro problema cuya demostración combina un gran número de cálculos con un razonamiento complejo es la conjetura de Kepler ([8]). Esta conjetura establece que ninguna disposición de esferas sólidas e idénticas en el espacio tiene una densidad media mayor que la disposición cúbica o la hexagonal. La densidad media de estas disposiciones es de $\frac{\pi}{\sqrt{18}} \approx 0{,}7404$. Se llama así debido a que fue enunciada primeramente por Johannes Kepler, en 1611. Kepler había empezado a estudiar disposiciones de esferas a raíz de un intercambio de correspondencia en 1606 con el matemático y astrónomo inglés Thomas Harriot. Harriot era amigo y asistente de Sir Walter Raleigh, quién le había encargado el problema de determinar la mejor manera de apilar balas de cañón en las cubiertas de sus barcos. Harriot publicó un estudio de varias disposiciones en 1591, y continuó con el desarrollo de una versión temprana de la teoría atómica.

Kepler no probó su conjetura; el primer paso en la demostración fue dado por Gauss, que demostró (en 1831) que si se consideran solamente disposiciones regulares entonces la conjetura es correcta. Esto implica que si alguna disposición no cumple la conjetura de Kepler, tiene que ser irregular. Eliminar todas las disposiciones irregulares es muy difícil, y es lo que hizo que esta conjetura sea tan difícil de demostrar. Luego de Gauss, no hubo progreso en la demostración durante el siglo XIX. En el año 1900 Hilbert incluyó al problema en su lista de los 23 problemas no resueltos. La conjetura de Kepler forma parte del problema número 18 de esa lista.

El siguiente paso en la demostración lo dió en 1953 el húngaro László Fejes Tóth, que mostró que el problema de determinar la máxima densidad de todos los cubrimientos (regulares e irregulares) puede reducirse al análisis de un número finito de casos. Esto implica que puede intentarse, al menos en teoría, una prueba exhaustiva.

Hubo numerosos intentos incorrectos de demostrar esta conjetura, hasta que en 1998 Tomas Hales anunció que su demostración estaba completa. Hales había anunciado previamente (en 1996) un esquema de la demostración exhaustiva que llevaría a cabo, y había explicado que llevaría un año o dos completarla. La demostración de Hales tenía 250 páginas y mas de 3 gigabytes de resultados.

Después de 4 años de evaluación, el jurado determinó que estaban *99 %* convencidos de que la prueba era correcta, pero que no podían asegurar la correctitud de los cálculos. En 2003 Hales anunció el comienzo del proyecto FlysPecK ([7]) para producir una prueba formal de la conjetura. El objetivo de este proyecto es eliminar

cualquier incerteza restante acerca de la validez de la demostración, creando una prueba que pueda ser verificada automáticamente con softwares como Coq.

Muchos de los resultados pendientes de la demostración de Hales son lemas que afirman que ciertos polinomios son no negativos en cajas dadas. En este trabajo presentamos la primera parte de una contribución al proyecto FlysPecK, que pretende automatizar las demostraciones de esos lemas. Los resultados *positivstellensatz* y *nullstellensatz* de geometría algebraica nos servirán como fundamentos teóricos para el trabajo. Si bien no aplican de manera directa al resultado que perseguimos, sin duda son similares a lo que buscamos y servirán como referencia.

En el capítulo 2 se dan los conceptos generales previos. Se introducen algunos conceptos de álgebra lineal y de geometría algebraica. Se introducen los conceptos relacionados con programación semidefinida que se utilizarán luego. El contenido original se encuentra fundamentalmente en los capítulos 3 y 4, que tratan sobre los certificados de no negatividad (la generación y el uso, respectivamente).

Capítulo 2

Preliminares

Este trabajo es la primera parte de un aporte al proyecto FlysPecK ([7]) que pretende automatizar la verificación de los lemas que afirman que ciertos polinomios son no negativos en cajas dadas. Se utilizarán técnicas de optimización y, mas precisamente, técnicas de *sumas de cuadrados* para demostrar desigualdades polinomiales en el asistente de pruebas Coq.

En este capítulo se introducirán los conceptos que serán utilizados en el resto del trabajo. En la sección 2.1 se describe el contexto en el cual surge este trabajo. En la sección 2.2 se introduce la definición de programa semidefinido. El la sección 2.3 se introducen conceptos relacionados a las representaciones como suma de cuadrados, y se introducen el nullstellensatz de Hilbert, el positivstellensatz de Stengle y la forma de Schmüdgen para el positivstellensatz. Estos son los conceptos de geometría algebraica que se usarán posteriormente para la generación de los certificados. A lo largo de todas las secciones se define la notación que se utilizará en el resto del trabajo.

2.1. Contexto

2.1.1. Coq

Coq [1] es un *asistente de pruebas* desarrollado en el INRIA. Esto significa que es un programa que verifica formalmente pruebas de enunciados matemáticos. Mas precisamente:

- por un lado, ayuda al usuario a construir una prueba formal,
- por otro lado, verifica que la prueba es correcta chequeando que la misma verifica las reglas formales de la lógica matemática.

El objetivo es alcanzar el mayor grado posible de veracidad en la correctitud de un enunciado matemático. Este objetivo es compartido con otros sistemas de prueba (Isabelle, PVS, HOL, Agda...) y tiene aplicaciones mas allá de la matemática pura (por ejemplo, la certificación de software crítico).

Una característica original de Coq es cómo maneja el cálculo:

- Los objetos del formalismo son realmente *programas funcionales*; por ejemplo la suma es definida a través de un algoritmo.
- Los objetos (siendo programas), son identificados lógicamente módulo el cálculo. Por ejemplo, no hay diferencia lógica entre $2 + 2$ y 4.

- Como consecuencia, las proposiciones $2 + 2 = 4$ y $4 = 4$ son también identificadas lógicamente; la primera puede ser demostrada en sólo un paso lógico: reflexivity.

El hecho de que el cálculo está profundamente conectado con la lógica de Coq permite obtener pruebas muy cortas de algunas prooposiciones. Por ejemplo la primalidad de números largos[5] puede ser demostrada combinando:

- una función de prueba `test : int → bool`,
- una prueba de corrección de que $\forall n, \texttt{test}(n) = \texttt{true} \Rightarrow \texttt{prime}(n)$.

En este trabajo se desarrolla un mecanismo similar, pero en el campo de la optimización.

2.1.2. Resultados complejos

Un promisorio campo de aplicación de las matemáticas es la obtención de nuevos resultados complejos. En este caso, el adelanto en certeza provisto por las verificaciones formales es particularmente bienvenido. Esto es aún mas veraz cuando los argumentos de las pruebas mezclan razonamientos matemáticos complejos *y* cálculos complejos.

Dado que incluye cálculos, Coq está bien provisto para esas tareas. En particular es, hasta el momento, el único sistema en el que se ha formalizado la prueba del teorema de los cuatro colores [4].

Un ejemplo es la reciente prueba (por Tomas Hales) de la *conjetura de Kepler* [8], de mas de 400 años de antigüedad. Esta prueba es muy difícil de admitir porque, entre otras cosas, se construye sobre resultados establecidos por extenso código *ad-hoc*. Por tal motivo, se está realizando esfuerzo para formalizar esta prueba [7]. El presente trabajo es la primera parte de una contribución a ese esfuerzo.

2.1.3. Desigualdades de Hales

La prueba de Thomas Hales incluye un gran número de desigualdades de la forma:

$$x_1 \in I_1 \wedge \cdots \wedge x_6 \in I_6 \Rightarrow P(x_1, \ldots, x_6) > 0$$

donde $I_1 \ldots I_6$ son intervalos y P es una expresión real.

En A.1, A.2 y A.3 se pueden ver tres ejemplos arbitrarios de esto. En estos capítulos se dará un mecanismo que permite la verificación en Coq de estas desigualdades.

2.2. Programación semidefinida

La formulación estándar de un programa semidefinido es

$$\begin{array}{ll} \min & \langle C, X \rangle \\ \text{s.t.} & \langle \mathcal{A}_i, X \rangle = b_i \qquad \forall i = 1, \ldots, m \\ & X \succeq 0. \end{array}$$

donde $X, C, \mathcal{A}_i \in \mathcal{S}_n$

La programación semidefinida es la programación lineal en el cono de las matrices semidefinidas. El vector $x \in \mathbb{R}^n_+$ de variables se reemplaza por una matriz $X \in \mathcal{S}^n_+$. En otras palabras, el cono del semieje $x \geq 0$ es reemplazado por el cono de matrices semidefinidas $X \succeq 0$.

La *función objetivo* $\langle C, X\rangle$ es la traza del producto CX. Esto es,

$$\sum_{i,j} c_{ij}x_{ij}.$$

2.3. Representaciones y certificados

En esta sección nos enfocaremos sobre las *representaciones* de los polinomios que son objetos de nuestro interés. Si un polinomio se puede expresar como suma de cuadrados de otros polinomios entonces se puede asegurar que el polinomio es no negativo en todo su dominio. Si queremos probar que un polinomio p es no negativo en un conjunto semialgebraico

$$\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^n \mid f_i(\mathbf{x}) \geq 0,\ i = 1 \dots m\},$$

podemos dar un certificado de que el conjunto de puntos en $\mathcal{K}$ en los que p es negativo es vacío. Introduciremos un teorema que nos servirá para generar ese certificado, el teorema de *positivstellensatz* de Stengle.

2.3.1. Sumas de cuadrados

Consideremos el siguiente problema.

Problema 1. *Sea p un polinomio en n variables. Proveer condiciones chequeables o un procedimiento para verificar la validez de la proposición*

$$p(\mathbf{x}) \geq 0, \qquad \forall \mathbf{x} \in \mathbb{R}^n.$$

Este es un problema NP-completo. Aparece en muchas áreas de aplicación y eso hace existan diferentes enfoques. Véase [12] para mas información sobre la historia del problema.

Una condición muy evidente que tiene que cumplir p es la de tener grado par. Una condición suficiente obvia para que p sea no negativo es la existencia de una descomposición como suma de cuadrados de otros polinomios:

$$p(\mathbf{x}) = \sum_i p_i^2(\mathbf{x}) \quad \forall \mathbf{x} \in \mathbb{R}^n, \qquad p_i \in \mathbb{R}[\mathbf{x}].$$

Es claro que si p puede escribirse de esta manera para algunos p_i, entonces p es no negativo para todos los valores de $\mathbf{x} \in \mathbb{R}^n$.

El método conocido como *Matriz de Gram* es una construcción fundamental para decidir si p es representable como suma de cuadrados. Si p tiene grado $2d$ podría ser una suma de cuadrados de subpolinomios, cada uno de los cuales tiene grado a lo sumo d. De hecho, cada monomio del polinomio original tiene grado a lo sumo $2d$ y puede ser expresado, posiblemente de mas de una manera, como un producto de dos monomios de grados a lo sumo d. En lo que sigue, llamaremos *supermonomios* a los monomios de grado a lo sumo $2d$ y *submonomios* a los monomios de grado a lo sumo d. Viendo a cada supermonomio como un producto de submonomios, el polinomio entero es una suma de números reales por productos de dos submonomios. Es decir, es una forma cuadrática en los submonomios.

Un ejemplo hará esto claro. Consideremos el polinomio $p(x, y) = 2x^3y + 3x^2y^2 - xy^3$. Aquí los supermonomios, de grado 4, son x^3y, x^2y^2 y xy^3. Cada uno puede ser escrito como producto de dos submonomios de grado 2. Por ejemplo $x^3y = (x^2)(xy)$, $x^2y^2 = (xy)^2$, y $xy^3 = (xy)(y^2)$. Esto conduce a la siguiente expresion para p:

$$\begin{pmatrix} x^2 & xy & y^2 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 3 & -\frac{1}{2} \\ 0 & -\frac{1}{2} & 0 \end{pmatrix} \begin{pmatrix} x^2 \\ xy \\ y^2 \end{pmatrix}$$

Por supuesto, esta expresión no es única. De hecho, para que una forma cuadrática particular represente un polinomio, todo lo que se requiere es que la suma de los coeficientes (en la forma cuadrática) de todos los productos de dos submonomios que resulten en el mismo supermonomio sea igual al coeficiente (en el polinomio) de ese supermonomio. De manera que el conjunto de formas cuadráticas que representan a un polinomio dado es un subespacio afín del conjunto de matrices simétricas.

Aprovechamos este ejemplo para introducir la notación que utilizaremos para las formas cuadráticas en los monomios. Por ejemplo, para notar la forma

$$\begin{pmatrix} x^2 & xy & y^2 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 3 & -\frac{1}{2} \\ 0 & -\frac{1}{2} & 0 \end{pmatrix} \begin{pmatrix} x^2 \\ xy \\ y^2 \end{pmatrix}$$

escribiremos

$$\left(\begin{array}{c|ccc} & x^2 & xy & y^2 \\ \hline x^2 & 0 & 1 & 0 \\ xy & 1 & 3 & -\frac{1}{2} \\ y^2 & 0 & -\frac{1}{2} & 0 \end{array}\right)$$

Para hacer gráfico lo que acabamos de decir, consideremos la forma siguiente:

$$\left(\begin{array}{c|ccc} & x^2 & xy & y^2 \\ \hline x^2 & t_1 & t_4 & t_7 \\ xy & t_2 & t_5 & t_8 \\ y^2 & t_3 & t_6 & t_9 \end{array}\right)$$

Aquí, t_1 será coeficiente de $(x^2)(x^2) = x^4$, t_2, será coeficiente de $(xy)(x^2) = x^3y$, t_4 será coeficiente de $(x^2)(xy) = x^3y$, etc. En definitiva, esta forma expresa el polinomio $x^4(t_1) + x^3y(t_4 + t_2) + x^2y^2(t_3 + t_5 + t_7) + xy^3(t_6 + t_8) + y^4(t_9)$.

Teorema 1 (Caracterización de polinomios representables como suma de cuadrados)**.** *Un polinomio p es representable como suma de cuadrados si y sólo si puede ser representado como una forma cuadrática semidefinida positiva de submonomios.*

Demostración. ($\Rightarrow$) (p representable como suma de cuadrados $\Rightarrow$ p representable como forma cuadratica semidefinida positiva de monomios)

Supongamos que tenemos $p = \sum_{i=1}^{k} p_i^2$. Sea $\mathbf{m}$ un vector que tiene por componentes a los monomios que aparecen en los p_i, de manera que todos los monomios que aparecen al menos una vez en al menos un p_i están presentes en $\mathbf{m}$, y de manera también que ningún monomio aparezca dos o mas veces en $\mathbf{m}$.

Sea $\mathbf{c}_i$ el vector con los coeficientes del polinomio p_i encolumnados, expresado de manera tal que valga $p_i = \mathbf{c}_i^T \mathbf{m}$.

Así, tenemos que

$$p = \sum_{i=1}^{k} p_i^2 = \sum_{i=1}^{k} (\mathbf{c}_i^T \mathbf{m})^2 = \sum_{i=1}^{k} (\mathbf{m}^T \mathbf{c}_i)(\mathbf{c}_i^T \mathbf{m}) = \mathbf{m}^T \left(\sum_{i=1}^{k} \mathbf{c}_i \mathbf{c}_i^T \right) \mathbf{m}$$

con $\left(\sum_{i=1}^{k} \mathbf{c}_i \mathbf{c}_i^T \right) \succeq 0$.

($\Leftarrow$) (p representable como forma cuadratica semidefinida positiva de monomios $\Rightarrow$ p representable como suma de cuadrados)

Tenemos que existen una matriz $Q \succeq 0$ de $s \times s$ y un vector de monomios $\mathbf{m}$ tales que $p = \mathbf{m}^T Q \mathbf{m}$.

Como $Q \succeq 0$, tenemos que existe L de $k \times s$ tal que $Q = L^T L$. Luego,

$$p = \mathbf{m}^T Q \mathbf{m} = \mathbf{m}^T L^T L \mathbf{m} = (L\mathbf{m})^T (L\mathbf{m}) = \sum_{i=1}^{k} (L\mathbf{m})_i^2$$

□

Observación 1 (La matriz de una forma cuadrática puede considerarse simétrica)**.** Toda matriz $\mathcal{M}$ puede expresarse como suma de una matriz simétrica $\mathcal{S}$ y una antisimétrica $\mathcal{A}$. Cualquier forma cuadrática en monomios dada por una matriz antisimétrica será igual a 0. Esto nos permite agregar al teorema el hecho de que la matriz de la representación del polinomio como forma cuadrática de monomios tiene que ser simétrica. De hecho, si existe una representación de p como forma cuadrática de monomios, entonces existe una representación de p como forma cuadrática de monomios en la que la matriz es simétrica.

Nuestro problema pedía proveer condiciones chequeables para concluir la no negatividad de un polinomio p de n variables. Dijimos que ese problema es NP-completo y que una condición suficiente es la existencia de una descomposición de p como suma de cuadrados de otros polinomios. Acabamos de demostrar que esa descomposición existe si y sólo si el polinomio puede ser representado como una forma cuadrática semidefinida positiva de submonomios. Esto es: si existen un vector de monomios $\mathbf{m}$ y una matriz simétrica semidefinida positiva Q tales que $p = \mathbf{m}^T Q \mathbf{m}$.

Para decidir si existen tales Q y $\mathbf{m}$ podemos tomar un $\mathbf{m}$ de manera tal el conjunto de monomios presentes en p (que llamaremos, de ahora en mas, mon(p)) esté incluído en el conjunto de monomios presentes en $\mathbf{m}^T\mathbf{m}$, y decidir la existencia de Q resolviendo un problema de programación semidefinida. Veamos esto con un ejemplo.

Ejemplo 1. Decidir si $p(x, y) = 2x^4 + 2x^3y - x^2y^2 + 5y^4$ es representable como suma de cuadrados.

En primer lugar tenemos que encontrar un $\mathbf{m}$ tal que mon(p) $\subseteq$ mon($\mathbf{m}^T\mathbf{m}$). Para un polinomio p de grado $2d$, un vector $\mathbf{m}$ que *siempre* funcionará será el vector de *todos* los monomios de grado menor o igual a d en las variables de p. En nuestro caso, sería el vector de todos los monomios de grado menor o igual que 2 en las variables x e y. Esa elección de $\mathbf{m}$ siempre funcionará, pero no será óptima. Véase 2.3.1. Otra elección de $\mathbf{m}$ que en nuestro caso funciona es tomar $\mathbf{m} = (x^2, y^2, xy)^T$. En este caso, vale que mon(p) $= \{x^4, x^3y, x^2y^2, y^4\} \subseteq \{x^4, x^3y, x^2y^2, xy^3, y^4\} =$ mon($\mathbf{m}^T\mathbf{m}$)

$$\begin{aligned} p(x,y) &= 2x^4 + 2x^3y - x^2y^2 + 5y^4 \\ &= \left(\begin{array}{c|ccc} & x^2 & xy & y^2 \\ \hline x^2 & t_1 & t_2 & t_3 \\ xy & t_2 & t_4 & t_5 \\ y^2 & t_3 & t_5 & t_6 \end{array}\right) \\ &= x^4(t_1) + x^3y(t_2 + t_2) + x^2y^2(t_3 + t_4 + t_3) + xy^3(t_5 + t_5) + y^4(t_6). \end{aligned}$$

Es decir, queremos decidir la factibilidad de

$$\begin{cases} t_1 = 2 \\ 2t_2 = 2 \\ 2t_3 + t_4 = -1 \\ 2t_5 = 0 \\ t_6 = 5 \end{cases}$$

sujeto a que

$$\begin{pmatrix} t_1 & t_2 & t_3 \\ t_2 & t_4 & t_5 \\ t_3 & t_5 & t_6 \end{pmatrix} \succeq 0$$

Usando programación semidefinida, encontramos una solución particular:

$$0 \preceq Q = \begin{pmatrix} 2 & -3 & 1 \\ -3 & 5 & 0 \\ 1 & 0 & 5 \end{pmatrix}$$

con lo que podemos concluir que $p(x,y) = 2x^4+2x^3y-x^2y^2+5y^4$ es representable como suma de cuadrados. Si queremos dar la representación, podemos seguir la idea de la demostración constructiva del teorema 2.3.1. Sea L tal que

$$L^T L = \begin{pmatrix} 2 & -3 & 1 \\ -3 & 5 & 0 \\ 1 & 0 & 5 \end{pmatrix}$$

En nuestro caso,

$$L = \begin{pmatrix} \frac{2}{\sqrt{2}} & -\frac{3}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{3}{\sqrt{2}} \end{pmatrix}$$

De manera que

$$L\mathbf{m} = \begin{pmatrix} \frac{2}{\sqrt{2}} & -\frac{3}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{3}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} x^2 \\ xy \\ y^2 \end{pmatrix} = \begin{pmatrix} \frac{2}{\sqrt{2}}x^2 - \frac{3}{\sqrt{2}}xy + \frac{1}{\sqrt{2}}y^2 \\ \frac{1}{\sqrt{2}}xy + \frac{3}{\sqrt{2}}y^2 \end{pmatrix}$$

Así,

$$\begin{aligned} p(x,y) &= 2x^4 + 2x^3y - x^2y^2 + 5y^4 \\ &= \left(\tfrac{2}{\sqrt{2}}x^2 - \tfrac{3}{\sqrt{2}}xy + \tfrac{1}{\sqrt{2}}y^2\right)^2 + \left(\tfrac{1}{\sqrt{2}}xy + \tfrac{3}{\sqrt{2}}y^2\right)^2 \end{aligned}$$

Mas ejemplos en [12], [10] y [2].

De esta manera, vemos que chequear si un polinomio es representable como suma de cuadrados reduce a resolver un problema de programación semidefinida, que es un problema tratable para el que ya se conocen soluciones eficientes. Esto es lo que hace a la representabilidad en suma de cuadrados interesante desde un punto de vista computacional, en contraste con la propiedad de no negatividad que no puede ser chequeada eficientemente.

Según 2.3.1, si existen una matriz simétrica $Q \succeq 0$ y un vector de monomios $\mathbf{m}$ tales que $p = \mathbf{m}^T Q \mathbf{m}$, entonces $p \geq 0$ en todo su dominio. En tal caso, el conjunto $\mathcal{C} = \{\mathbf{x} \in \mathrm{dom}(p) \mid p(\mathbf{x}) < 0\}$ es *no factible* y la matriz Q junto con el vector $\mathbf{m}$ forman un *certificado de infactibilidad* de $\mathcal{C}$.

En una dimensión, el anillo $\mathbb{R}[x]$ de los polinomios reales de una sola variable tiene la propiedad fundamental de que todo polinomio no negativo $p \in \mathbb{R}[x]$ es una suma de cuadrados de polinomios. En varias dimensiones, sin embargo, es posible que un polinomio sea no negativo sin que sea representable como suma de cuadrados. Un ejemplo explícito de polinomio no negativo que no es suma de cuadrados es la forma de Motzkin siguiente: (aquí para $n = 3$)

$$M(x, y, z) = x^4y^2 + x^2y^4 + z^6 - 3x^2y^2z^2.$$

En este caso, la no negatividad puede mostrarse fácilmente usando la desigualdad aritmético-geométrica [1] y una demostración muy sencilla de la no existencia de una descomposición en suma de cuadrados puede encontrarse en [13].

De manera que tenemos que podemos usar programación semidefinida para producir certificados de representabilidad como suma de cuadrados. Sin embargo, como ya mostramos existen polinomios no negativos que no son representables como suma de cuadrados. En esos casos este método será incapaz de producir el certificado que buscamos. Por otro lado, los certificados de representabilidad como suma de cuadrados certifican no negatividad en *todo* el dominio del polinomio. En nuestros ejemplos de la demostración de Hales necesitaremos probar probar la no negatividad de polinomios dados en subconjuntos del dominio. Analizaremos en secciones posteriores algunos resultados de geometría algebraica que nos serán de utilidad para generar certificados de no negatividad en conjuntos semialgebraicos dados.

La elección de los monomios

Nos hemos encontrado ya con dos situaciones en las que tuvimos que elegir un vector de monomios. En el ejemplo 1 teníamos que elegir un vector de monomios $\mathbf{m}$ tal que $\text{mon}(2x^4 + 2x^3y - x^2y^2 + 5y^4) = \{x^4, x^3y, x^2y^2, y^4\} \subseteq \text{mon}(\mathbf{m}^T\mathbf{m})$, y previamente habíamos tenido que elegir un $\mathbf{m}$ tal que $\text{mon}(2x^3y + 3x^2y^2 - xy^3) \subseteq \text{mon}(\mathbf{m}^T\mathbf{m})$.

Se explicó en 1 que si tomamos a $\mathbf{m}$ como el vector de *todos* los monomios de grados menores o iguales a la mitad del grado del polinomio p, en las variables del polinomio p entonces podremos estar seguros que $\text{mon}(p) \subseteq \text{mon}(\mathbf{m}^T\mathbf{m})$. Sin embargo, del ejemplo anterior se puede ver que la cantidad de variables que tendrá el programa semidefinido que necesitaremos resolver para decidir si un polinomio es representable como suma de cuadrados será $1 + 2 + \cdots + n = \frac{n(n+1)}{2}$, donde n es el número de monomios (nótese que el número de variables no es n^2 debido a que podemos considerar que la matriz es simétrica). Esto hace que el costo de resolver el problema es cuadrático en la cantidad de monomios. Interesa, por lo tanto, elegir uno conjunto de los $\mathbf{m}$ que tengan el menor número posible de monomios.

Supongamos que queremos ver si el polinomio $p(x, y, z) = 9x^2y^4 + 9x^2z^4 + 36x^2y^3 + 36x^2y^2 - 48xyz^2 + 4y^4 + 4z^4 - 16y^3 + 16y^2$ es representable como suma de cuadrados. Primero enumeramos todos los submonomios de grado a lo sumo 3 en las tres variables. Tenemos 20 submonomios: 1, x, y, z, x^2, xy, xz, y^2, yz, z^2, x^3, x^2y, x^2z, xy^2, xyz, xz^2, y^3, y^2z, yz^2, z^3.

Es fácil ver que los submonomios como y^2z, que no dividen a ninguno de los supermonomios en p, pueden descartarse. Pero hay un resultado mucho mas fuerte, demostrado por Reznick en [14]. Consideremos a los monomios como vectores de enteros no negativos (los exponentes de cada variable). La cápsula convexa de los vectores correspondientes a los monomios que aparecen en p se llama *polítopo de Newton* del polinomio. Reznick mostró que los únicos submonomios que se necesitan

[1] Sigue trivialmente de la desigualdad $\frac{a+b+c}{3} \geq \sqrt[3]{abc}$ aplicada a $(a, b, c) = (x^4y^2, x^2y^4, z^6)$.

son los que corresponden a vectores que caen dentro del polítopo que es la mitad del polítopo de Newton del polinomio.

En nuestro caso, los vectores correspondientes a los monomios de p son $(2,4,0)$, $(2,0,4)$, $(2,3,0)$, $(2,2,0)$, $(1,1,2)$, $(0,4,0)$, $(0,0,4)$, $(0,3,0)$, $(0,2,0)$. El punto $(2,2,0)$ puede expresarse como $\frac{1}{4}((2,2,0)+(0,2,0)+(0,0,4)+(2,0,4))$. Luego, el polítopo de Newton de p será la cápsula convexa de $(0,2,0)$, $(0,4,0)$, $(2,2,0)$, $(2,4,0)$, $(0,0,4)$ y $(2,0,4)$. La mitad de este polítopo es la cápsula convexa de $(0,1,0)$, $(0,2,0)$, $(1,1,0)$, $(1,2,0)$, $(0,0,2)$ y $(1,0,2)$. Los únicos puntos que caen dentro de esta cápsula son $(0,1,1)$ y $(1,1,1)$. De manera que sólo necesitaremos los siguientes 8 monomios: y, y^2, xy, xy^2, z^2, xz^2, yz, xyz.

2.3.2. Nullstellensatz

Definición 1 (Ideal)**.** *Decimos que un conjunto $I \subseteq \mathbb{C}[x_1,\dots,x_n]$ es un* ideal *si satisface:*

- $0 \in I$.
- *Si $a, b \in I$, entonces $a+b \in I$.*
- *Si $a \in I$ y $b \in \mathbb{C}[x_1,\dots,x_n]$, entonces $a \cdot b \in I$.*

Definición 2 (Ideal generado por un conjunto de polinomios)**.** *Dado un conjunto finito de polinomios $(f_i)_{i=1,\dots,s}$, definimos el conjunto*

$$\langle f_1,\dots,f_s\rangle := \left\{ \sum_{i=1}^{s} f_i g_i, \qquad g_i \in \mathbb{C}[x_1,\dots x_n] \right\}.$$

Se puede mostrar fácilmente que $\langle f_1,\dots,f_s\rangle$ es un ideal, y lo llamamos ideal generado por (f_i).

Teorema 2 (Nullstellensatz de Hilbert)**.** *Sea $(f_j)_{j=1,\dots,s}$ una familia finita de polinomios en $\mathbb{C}[x_1,\dots,x_n]$. Sea I el ideal generado por $(f_j)_{j=1,\dots,s}$. Luego, las siguientes afirmaciones son equivalentes:*

1. *El conjunto $\{x \in \mathbb{C}^n | f_i(x) = 0, \qquad i = 0,\dots,s\}$ es vacío.*
2. *El polinomio 1 pertenece al ideal (*i.e.: *$1 \in I$).*
3. *El ideal es igual al anillo completo: $I = \mathbb{C}[x_1,\dots,x_n]$.*
4. *Existen polinomios $g_i \in \mathbb{C}[x_1,\dots,x_n]$ tales que*

$$f_1(x)g_1(x) + \cdots + f_s(x)g_s(x) = 1.$$

Ejemplo 2. Supongamos que queremos probar que el sistema de ecuaciones polinomiales siguiente no tiene soluciones en $\mathbb{C}$:

$$\begin{cases} f_1(\mathbf{x}) := x^2 + y^2 - 1 = 0 \\ f_2(\mathbf{x}) := x + y = 0 \\ f_3(\mathbf{x}) := 2x^3 + y^3 + 1 = 0 \end{cases}$$

Consideramos los polinomios siguientes.

$$\begin{aligned} g_1(\mathbf{x}) &:= \tfrac{1}{7}(1 - 16x - 12y - 8xy - 6y^2) \\ g_2(\mathbf{x}) &:= \tfrac{1}{7}(-7y - x + 4y^2 - 16 + 12xy + 2y^3 + 6y^2x) \\ g_3(\mathbf{x}) &:= \tfrac{1}{7}(8 + 4y). \end{aligned}$$

Se puede verificar que

$$f_1 g_1 + f_2 g_2 + f_3 g_3 = 1,$$

con lo que (por el teorema de Nullstellensatz de Hilbert) no hay soluciones en $\mathbb{C}$ para nuestro sistema de ecuaciones. El conjunto $\{g_1; g_2; g_3\}$ es, entonces, un certificado de que el sistema inicial no tiene soluciones en $\mathbb{C}$.

2.3.3. Positivstellensatz

Las condiciones del Nullstellensatz son necesarias y suficientes sólo en el caso en el que el campo es algebraicamente cerrado[2] (como es el caso de $\mathbb{C}$). Por ejemplo, en los reales, la ecuación

$$x^2 + 1 = 0$$

no tiene solución. Sin embargo, el ideal asociado no incluye al elemento 1. Cuando estamos interesados principalmente en soluciones reales, la ausencia de clausura algebraica en $\mathbb{R}$ fuerza a un enfoque diferente.

Antes de presentar el Nullstellensatz para el caso real, necesitaremos introducir algunos conceptos.

Definición 3 (Monoide multiplicativo generado por un conjunto de funciones)**.** *Dado un conjunto de polinomios $p_i \in \mathbb{R}[x_1, \ldots, x_n]$, sea $M(p_i)$ el conjunto de todos los productos finitos de p_i (incluyendo el producto vacío). Llamamos a este conjunto* monoide multiplicativo *generado por los p_i.*

Definición 4 (Cono)**.** *Un cono P de $\mathbb{R}[x_1, \ldots, x_n]$ es un subconjunto de $\mathbb{R}[x_1, \ldots, x_n]$ que satisface las siguientes propiedades:*

- $a, b \in P \Rightarrow a + b \in P$
- $a, b \in P \Rightarrow a \cdot b \in P$
- $a \in \mathbb{R}[x_1, \ldots, x_n] \Rightarrow a^2 \in P$

Dado un conjunto $S \subseteq \mathbb{R}[x_1, \ldots, x_n]$, sea $P(S)$ el menor cono de $\mathbb{R}[x_1, \ldots, x_n]$ que contiene a S. Es fácil ver que $P(\emptyset)$ corresponde a los polinomios que pueden ser expresados como suma de cuadrados, y es el menor cono en $\mathbb{R}[x_1, \ldots, x_n]$. Para un conjunto finito $S = \{a_1, \ldots, a_m\} \subseteq \mathbb{R}[x_1, \ldots, x_n]$, el cono asociado es:

$$P(S) = \left\{ p + \sum_{i=1}^{r} q_i b_i \quad | \quad p, q_1, \ldots, q_r \in P(\emptyset), \quad b_1, \ldots, b_r \in M(a_i) \right\}.$$

El positivstellensatz de Stengle dice que para un sistema de ecuaciones y desigualdades polinomiales, o bien existe una solución en $\mathbb{R}^n$ o bien existe una identidad polinomial que *certifica* el hecho de que no existe solución.

Teorema 3 (Positivstellensatz de Stengle)**.** *Sean $(f_j)_{j=1,\ldots,s}$, $(g_k)_{k=1,\ldots,t}$, $(h_l)_{l=1,\ldots,u}$ familias finitas de polinomios en $\mathbb{R}[x_1, \ldots, x_n]$. Sean:*

[2] Un campo F se dice *algebraicamente cerrado* si todo polinomio p en una variable con coeficientes en F (*i.e.:* $p \in F[x]$) de grado al menos 1 tiene una raíz en F. Dos propiedades equivalentes son:

- Los únicos polinomios irreductibles son los de grado uno.
- Todos los polinomios son producto de polinomios de grado uno.

- *P el cono generado por $(f_j)_{j=1,\dots,s}$,*
- *M el monoide multiplicativo generado por $(g_k)_{k=1,\dots,t}$*
- *I el ideal generado por $(h_l)_{l=1,\dots,u}$*

Luego, las siguientes propiedades son equivalentes:

1. *El conjunto*
$$\left\{ x \in \mathbb{R}^n \;\middle|\; \begin{array}{ll} f_j(x) \geq 0, & j = 1, \dots, s \\ g_k(x) \neq 0, & k = 1, \dots, t \\ h_l(x) = 0, & l = 1, \dots, u \end{array} \right\}$$
es vacío.
2. *Existen $f \in P, g \in M, h \in I$ tales que $f + g^2 + h = 0$.*

Positivstellensatz garantiza la existencia de *certificados de infactibilidad* o *refutaciones*, dados por los polinomios f, g y h en los casos en los que el sistema no tiene solución.

Ejemplo 3. Consideremos una ecuación cuadrática estándar:

$$x^2 + ax + b = 0$$

Esta ecuación no tiene solución en los reales en los casos en los que el discriminante es negativo. Esto es:

$$D := b - \frac{a^2}{4} > 0.$$

En esos casos, Positivstellensatz asegura que existe un certificado (f, g, h) tal que $f \in P, g \in M, h \in I$ tales que $f + g^2 + h = 0$. En este caso, de hecho, podemos tomar

$$\begin{aligned} f &:= \left[\tfrac{1}{\sqrt{D}} (x + \tfrac{a}{2}) \right]^2 \\ g &:= 1 \\ h &:= -\tfrac{1}{D}(x^2 + ax + b) \end{aligned}$$

y se cumplen las condiciones que el teorema asegura.

2.3.4. Forma de Schmüdgen para el positivstellensatz

Observación 2 (Notación)**.** Utilizaremos la notación $\Sigma[\mathbf{x}]$ para denotar el conjunto de polinomios en $\mathbb{R}[\mathbf{x}]$ que son sumas de cuadrados de elementos de $\mathbb{R}[\mathbf{x}]$

La forma de Schmüdgen para el positivstellensatz afirma que si p es estrictamente positivo en el conjunto semialgebraico **compacto** $\mathcal{K}$, entonces p pertenece al cono generado por las restricciones que determinan $\mathcal{K}$.

Teorema 4 (Forma de Schmüdgen para el positivstellensatz)**.** *Sean $\{f_i\}_{i=1}^m \subset \mathbb{R}[\mathbf{x}]$ tales que*

$$\mathcal{K} := \{\mathbf{x} \in \mathbb{R}^n \mid f_i(\mathbf{x}) \geq 0,\ i = 1, \dots, m\}$$

es compacto. Si $p \in \mathbb{R}[\mathbf{x}]$ es estrictamente positivo en $\mathcal{K}$, entonces $p \in P(f_1, \dots, f_m)$, esto es:

$$p = \sum_{J \subset \{1,\ldots,m\}} SOS_J f_j, \qquad para\ algunos SOS_J \in \Sigma[\mathbf{x}]$$

y $f_J = \prod_{j \in J} f_j$.

Capítulo 3

Generación de los certificados

Se estudiará en este capítulo una forma de certificar la no negatividad de $P \in \mathbb{R}[n]$ en el subconjunto *compacto* $\mathcal{K} \subseteq \mathbb{R}^n$ dado por la familia de restricciones polinomiales $\{f_i \geq 0\}_{i \in 1 \ldots k}$. Es decir, en el subconjunto

$$\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^n \mid f_i(\mathbf{x}) \geq 0 \qquad \forall i \in \mathbb{N}, 1 \leq i \leq k\}$$

Se ha mencionado que existen *representaciones* a partir de las cuales la no negatividad de un polinomio p resulta evidente. Si p es representable, por ejemplo, como una suma de cuadrados de otros polinomios, resulta evidente que es no negativo en todo su dominio y por lo tanto en $\mathcal{K}$. En este capítulo se pretende dar una representación que permita concluir (en Coq) la no negatividad de p en $\mathcal{K}$.

Schmüdgen (2.3.4) dice que si un polionomio q es estrictamente positivo en $\mathcal{K}$, entonces pertenece al cono generado por $\{f_i\}_{i \in 1 \ldots k}$. Es decir,

$$q = \sum_{J \subseteq \{1 \ldots k\}} f_J \; SOS_J \quad \text{para algunos } SOS_J \in \Sigma[\mathbf{x}]$$

donde $f_J = \prod_{j \in J} f_j$.

Nótese, además, que todo polinomio perteneciente al cono generado por las f_i es no negativo en $\mathcal{K}$.

Para probar la no negatividad de p en $\mathcal{K}$, podemos entonces calcular el mayor γ para el cual $p - \gamma$ pertenece al cono generado por las f_i. Es decir, podemos buscar el mayor γ para el cual vale que

$$p - \gamma = \sum_{J \subseteq \{1 \ldots k\}} f_J \; SOS_J. \tag{3.1}$$

Si vale que $\gamma >= 0$, tendremos que $p \geq 0$ en $\mathcal{K}$. Este problema puede resolverse usando programación semidefinida, de manera similar a la mostrada en 1.

Ejemplo 4. Encontrar el mínimo valor que toma $p(x_1, x_2) = x_1^2 + 2x_2^2$, cuando $x_1^2 - x_2 \geq 1$ y $x_2 \geq x_1$.

El programa semidefinido, en este caso, es:

$$\begin{array}{ll} \min & -\gamma \\ \text{s.t.} & p(x_1, x_2) - \gamma = x_1^2 + 2x_2^2 - \gamma \geq 0 \\ & (x_1, x_2) \in \mathcal{K} \end{array}$$

Las restricciones que tenemos, expresadas de la forma $f_i \geq 0$ son:

$$f_1(x_1, x_2) = x_1^2 - x_2 - 1 \geq 0$$
$$f_2(x_1, x_2) = x_2 - x_1 \geq 0$$

Queremos encontrar el mayor γ tal que

$$\begin{aligned} x_1^2 + 2x_2^2 - \gamma = \quad & SOS_{\emptyset} \\ & + SOS_{\{1\}} f_1(x_1, x_2) \\ & + SOS_{\{2\}} f_2(x_1, x_2) \\ & + SOS_{\{1;2\}} f_1(x_1, x_2) f_2(x_1, x_2) \end{aligned}$$

para algunos $SOS_{\emptyset}, SOS_{\{1\}}, SOS_{\{2\}}, SOS_{\{1;2\}} \in \Sigma[x_1, x_2]$.

En 2.3.1 se mostró que un polinomio $q \in \mathbb{R}[n]$ puede expresarse como suma de cuadrados si y sólo si existen un vector de monomios $z = (\text{monomio}_1; \ldots; \text{monomio}_n)^T$ y una matriz $Q \succeq 0$ tales que $q = z^T Q z$.

De manera que nuestro problema se reescribe como el problema de encontrar el mayor γ tal que existen vectores de monomios $z_{\emptyset}$, $z_{\{1\}}$, $z_{\{2\}}$, $z_{\{1;2\}}$ y matrices semidefinidas positivas $Q_{\emptyset}$, $Q_{\{1\}}$, $Q_{\{2\}}$, $Q_{\{1;2\}}$ que satisfagan:

$$\begin{aligned} x_1^2 + 2x_2^2 - \gamma = \quad & z_{\emptyset}^T Q_{\emptyset} z_{\emptyset} \\ & + z_{\{1\}}^T Q_{\{1\}} z_{\{1\}} f_1(x_1, x_2) \\ & + z_{\{2\}}^T Q_{\{2\}} z_{\{2\}} f_2(x_1, x_2) \\ & + z_{\{1;2\}}^T Q_{\{1;2\}} z_{\{1;2\}} f_1(x_1, x_2) f_2(x_1, x_2) \end{aligned} \tag{3.2}$$

Como se mencionó en 2.3.1, elegir de manera óptima los monomios que componen los z_i es un problema no trivial. En los ejemplos que siguen, especificaremos de manera explícita cuales son los monomios que elegimos en cada caso. Al momento de implementar el algoritmo se sugiere tener en cuenta las obervaciones hechas en 2.3.1 para tal elección.

Si en 3.2 elegimos los monomios de las z_i de manera que los términos del lado derecho de la igualdad tengan grados menores o iguales a 4, obtenemos el problema siguiente (siguiendo la notación propuesta en 2.3.1):

$$x_1^2 + 2x_2^2 - t_1 =$$

$$= \left(\begin{array}{c|ccccc} & 1 & x_1 & x_2 & x_1^2 & x_2^2 \\ \hline 1 & t_{21} & t_{26} & t_{31} & t_{36} & t_{41} \\ x_1 & t_{22} & t_{27} & t_{32} & t_{37} & t_{42} \\ x_2 & t_{23} & t_{28} & t_{33} & t_{38} & t_{43} \\ x_1^2 & t_{24} & t_{29} & t_{34} & t_{39} & t_{44} \\ x_2^2 & t_{25} & t_{30} & t_{35} & t_{40} & t_{45} \end{array}\right) + \left(\begin{array}{c|ccc} & 1 & x_1 & x_2 \\ \hline 1 & t_2 & t_5 & t_8 \\ x_1 & t_3 & t_6 & t_9 \\ x_2 & t_4 & t_7 & t_{10} \end{array}\right) (x_2 - x_1)$$

$$+ \left(\begin{array}{c|ccc} & 1 & x_1 & x_2 \\ \hline 1 & t_{11} & t_{14} & t_{17} \\ x_1 & t_{12} & t_{15} & t_{18} \\ x_2 & t_{13} & t_{16} & t_{19} \end{array}\right) (x_1^2 - x_2 - 1) + \left(\begin{array}{c|c} & 1 \\ \hline 1 & t_{20} \end{array}\right) (x_1^2 - x_2 - 1)(x_2 - x_1) \tag{3.3}$$

Este problema puede resolverse con programación semidefinida. La formulación como programa semidefinido es directa. Planteamos las ecuaciones teniendo en cuenta los coeficientes de los monomios a cada lado de la igualdad y obtenemos el sistema de ecuaciones. De manera que queremos minimizar $-t_1$ sujeto a que

$$\begin{cases} [1] & = -t_1 & = t_{21} + t_2 + t_{11} + t_{20} \\ [x_1] & = 0 & = t_{22} + t_{26} + t_3 + t_5 + t_{12} + t_{14} \\ [x_2] & = 0 & = t_{23} + t_{31} + t_4 + t_8 + t_{13} + t_{17} \\ [x_1^2] & = 1 & = t_{24} + t_{27} + t_{36} + t_6 + t_{15} \\ [x_2^2] & = 2 & = t_{25} + t_{33} + t_{41} + t_{10} + t_{19} \\ [x_1x_2] & = 0 & = t_{28} + t_{32} + t_7 + t_9 + t_{16} + t_{18} \\ [x_1^3] & = 0 & = t_{29} + t_{37} \\ [x_1^2x_2] & = 0 & = t_{34} + t_{38} \\ [x_1x_2^2] & = 0 & = t_{30} + t_{42} \\ [x_2^3] & = 0 & = t_{35} + t_{43} \\ [x_1^4] & = 0 & = t_{39} \\ [x_1^2x_2^2] & = 0 & = t_{40} + t_{44} \\ [x_2^4] & = 0 & = t_{45} \end{cases}$$

y a que

$$\begin{pmatrix} t_{21} & t_{26} & t_{31} & t_{36} & t_{41} \\ t_{22} & t_{27} & t_{32} & t_{37} & t_{42} \\ t_{23} & t_{28} & t_{33} & t_{38} & t_{43} \\ t_{24} & t_{29} & t_{34} & t_{39} & t_{44} \\ t_{25} & t_{30} & t_{35} & t_{40} & t_{45} \end{pmatrix} \succeq 0$$

$$\begin{pmatrix} t_2 & t_5 & t_8 \\ t_3 & t_6 & t_9 \\ t_4 & t_7 & t_{10} \end{pmatrix} \succeq 0 \qquad .$$

$$\begin{pmatrix} t_{11} & t_{14} & t_{17} \\ t_{12} & t_{15} & t_{18} \\ t_{13} & t_{16} & t_{19} \end{pmatrix} \succeq 0$$

$$\begin{pmatrix} t_{20} \end{pmatrix} \succeq 0$$

En 3.4 se muestra una solución particular a este programa. A partir de esta solución particular podemos obtener una representación que permitirá deducir en Coq que $x_1^2+2x_2^2 \geq 0{,}8750$ en los (x_1, x_2) que cumplan $x_1^2-x_2-1 \geq 0$ y que x_2-x_1. Lo hacemos de la misma manera que lo hicimos en 1.

$$x_1^2 + 2x_2^2 - 0{,}8750 =$$

$$= \left(\begin{array}{c|ccccc} & 1 & x_1 & x_2 & x_1^2 & x_2^2 \\ \hline 1 & 0{,}125 & 0 & 0{,}5 & 0 & 0 \\ x_1 & 0 & 0 & 0 & 0 & 0 \\ x_2 & 0{,}5 & 0 & 2 & 0 & 0 \\ x_1^2 & 0 & 0 & 0 & 0 & 0 \\ x_2^2 & 0 & 0 & 0 & 0 & 0 \end{array}\right) + \left(\begin{array}{c|ccc} & 1 & x_1 & x_2 \\ \hline 1 & 0 & 0 & 0 \\ x_1 & 0 & 0 & 0 \\ x_2 & 0 & 0 & 0 \end{array}\right) (x_2 - x_1)$$

$$+ \left(\begin{array}{c|ccc} & 1 & x_1 & x_2 \\ \hline 1 & 1 & 0 & 0 \\ x_1 & 0 & 0 & 0 \\ x_2 & 0 & 0 & 0 \end{array}\right) (x_1^2 - x_2 - 1) + \left(\begin{array}{c|c} & 1 \\ \hline 1 & 0 \end{array}\right) (x_1^2 - x_2 - 1)(x_2 - x_1) \tag{3.4}$$

Tenemos que:

$$\begin{pmatrix} t_{21} & t_{26} & t_{31} & t_{36} & t_{41} \\ t_{22} & t_{27} & t_{32} & t_{37} & t_{42} \\ t_{23} & t_{28} & t_{33} & t_{38} & t_{43} \\ t_{24} & t_{29} & t_{34} & t_{39} & t_{44} \\ t_{25} & t_{30} & t_{35} & t_{40} & t_{45} \end{pmatrix} = \begin{pmatrix} 0{,}125 & 0 & 0{,}5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0{,}5 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{2\sqrt{2}} \\ 0 \\ \sqrt{2} \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} \frac{1}{2\sqrt{2}} & 0 & \sqrt{2} & 0 & 0 \end{pmatrix} .$$

Con lo que

$$\left(\begin{array}{c|ccccc} & 1 & x_1 & x_2 & x_1^2 & x_2^2 \\ \hline 1 & 0{,}125 & 0 & 0{,}5 & 0 & 0 \\ x_1 & 0 & 0 & 0 & 0 & 0 \\ x_2 & 0{,}5 & 0 & 2 & 0 & 0 \\ x_1^2 & 0 & 0 & 0 & 0 & 0 \\ x_2^2 & 0 & 0 & 0 & 0 & 0 \end{array}\right) = (\tfrac{1}{2\sqrt{2}} \cdot 1 + \sqrt{2} \cdot x_2)^2$$

De la misma manera,

$$\begin{pmatrix} t_{11} & t_{14} & t_{17} \\ t_{12} & t_{15} & t_{18} \\ t_{13} & t_{16} & t_{19} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} .$$

Con lo que

$$\left(\begin{array}{c|ccc} & 1 & x_1 & x_2 \\ \hline 1 & t_{11} & t_{14} & t_{17} \\ x_1 & t_{12} & t_{15} & t_{18} \\ x_2 & t_{13} & t_{16} & t_{19} \end{array}\right) (x_1^2 - x_2 - 1) = (1 \cdot 1 + 0 \cdot x_1 + 0 \cdot x_2)^2 (x_1^2 - x_2 - 1)$$

Luego, tenemos que

$$x_1^2 + 2x_2^2 - 0{,}8750 = (\tfrac{1}{2\sqrt{2}} \cdot 1 + \sqrt{2} \cdot x_2)^2 + (1 \cdot 1 + 0 \cdot x_1 + 0 \cdot x_2)^2 (x_1^2 - x_2 - 1)$$

Nótese que en todas las representaciones que obtengamos de esta manera los términos de la suma de la parte derecha de la igualdad serán siempre un producto de dos factores; uno de ellos es una suma de cosas al cuadrado y el otro es un producto de restricciones (habrá un término que será el producto de 0 restricciones; que se considera igual a 1). El factor que está representado como suma de cuadrados es no negativo, con lo que tenemos que cuando todas las restricciones son no negativas, la parte izquierda de la igualdad es no negativa. Los puntos en los que las restricciones son no negativas son precisamente los puntos del conjunto $\mathcal{K}$. De manera que hemos dado una representación para la solución 3.4 que permitirá demostrar en Coq (como veremos mas adelante) que $x_1^2 + 2x_2^2 \geq 0$ en $\mathcal{K}$. Hemos dado un **certificado** de que $p \geq 0{,}8750$ (y consecuentemente $p \geq 0$ en $\mathcal{K}$. En este ejemplo particular la no negatividad se podía deducir de manera directa porque tanto x_1^2 como $2x_2^2$ son no negativos, pero se pretende graficar el mecanismo a través de un ejemplo sencillo.

3.1. Una mejora

El teorema 2.3.4 es un resultado poderoso, pero notemos que tiene 2^k términos, donde k es el número de restricciones. En nuestro ejemplo, podemos ver que la suma 3.3 tiene 4 términos y tenemos 2 restricciones. Puede hacerse una mejora importante bajo una suposición relativamente débil sobre el compacto $\mathcal{K}$ (véase [10]).

Suposición 1. *Sea* $\{g_i\}_{j=1}^m \subset \mathbf{R}[\mathbf{x}]$*, sea* $\mathcal{K}$ *igual que antes y sea*

$$M = \left\{ u = q_0 + \sum_{j=1}^{m} q_j g_j \mid q_j \in \Sigma[\mathbf{x}] \right\}.$$

Asumimos que existe $u \in M$ *tal que* $\{\mathbf{x} \in \mathbb{R}^n \mid u(\mathbf{x}) \geq 0\}$ *es compacto.*

Teorema 5. *Bajo la suposición anterior, si* p *es estrictamente positivo en* $\mathcal{K}$*, entonces* $p \in M$*. Esto es:*

$$p = f_0 + \sum_{j=1}^{m} f_j g_j, \quad f_j \in \Sigma[\mathbf{x}], \quad j = 0, 1, \ldots, m.$$

A diferencia del teorema 2.3.4, el número de términos en esta representación es lineal en el número de restricciones que definen $\mathcal{K}$, eso es una mejora *radical* desde el punto de vista computacional. La condición que hemos impuesto no es muy restrictiva. De hecho, es satisfecha al menos en los siguientes casos:

1. Todas las f_i's son lineales, con lo que $\mathcal{K}$ es un polítopo.

2. El conjunto $\{\mathbf{x} \in \mathbb{R}^n \mid f_i(\mathbf{x}) \geq 0\}$ es compacto para algún $j \in \{0, 1, \ldots, m\}$

En los polinomios de la demostración de Hales de la conjetura de Kepler, las restricciones serán siempre (o al menos en la gran mayoría de los casos) simplemente restricciones que definen cajas. Estaremos interesados en demostrar que un polinomio p es no negativo en una caja dada por algunas restricciones. De manera que le suposición efectuada no restringe el conjunto de problemas a demostrar de la demostración de Hales.

3.2. Cuestiones de implementación. Resultados.

Actualmente las rutinas que generan los certificados están implementadas en MatLab. Para la solución de los SDP's se usa la librería SeDuMi. Se implementó un conjunto sencillo de rutinas para automatizar la formulación del programa semidefinido a partir del problema inicial. SeDuMi busca una solución numérica aproximada al SDP que recibe como entrada. Eso hace que en algunos casos el polinomio para el cual se genera el certificado resulte ser levemente diferente al polinomio original.

La solución *temporal* que se implementó para resolver este problema es calcular *exactamente* el polinomio que resulta de restar el polinomio del certificado al polinomio original, y calcular una cota superior para ese error usando un software que implementa la conversión a bases de Bernstein que funciona muy bien para polinomios con coeficientes muy pequeños. Por ahora esa cota se agregará como axioma en Coq (véase el capítulo siguiente) y no será demostrada. A futuro, se pretende reescribir el algoritmo que SeDuMi implementa para que los cálculos sean hechos de manera simbólica.

Los ejemplos que se citan en el apéndice A fueron probados *con* y *sin* la suposición y eligiendo los monomios de las formas bilineales de manera que los términos tengan grados menores o iguales a DEGREE. Los tiempos de ejecución y las cotas que se encontraron fueron:

A.1		
	Con la suposición	Sin la suposición
DEGREE = 4	*La ejecución falló porque los errores numéricos en SeDuMi llevan a una división por 0.*	1781.47 170 segundos
DEGREE = 6	1781.47 150 segundos	1781.47 aprox. 7 hs. y 10 minutos
A.2		
DEGREE = 4	*La ejecución falló porque los errores numéricos en SeDuMi llevan a una división por 0.*	1340.1165 168 segundos
DEGREE = 6	1340.1167 149 segundos	1340.1167 aprox. 7 hs. y 25 minutos
A.3		
DEGREE = 4	0 1.14 segundos	0 12.7 segundos
DEGREE = 6	0 10.62 segundos	0 262.5 segundos
DEGREE = 8	0 102 segundos	*Se canceló la ejecución después de varias horas.*

La diferencia entre el polinomio entrada y el polinomio para el cual se genera el certificado se debe a errores numéricos en SeDuMi. Debido a que estas deferencias de redondeos son muy pequeñas, el polinomio diferencia tiene coeficientes muy pequeños y eso hace que la representación del polinomio sea muy larga. (vease el capítulo siguiente). Para calcular una cota para esos polinomios se utilizó Amber [15]; un software que implementa aritmética de intervalos y conversión a bases de Bernstein. En casos como este, en los que el polinomio tiene coeficientes muy chicos, ese software ha demostrado ser muy eficiente. (véase [16]).

3.3. Estado actual y trabajo futuro

Se pretende, a futuro, reescribir las rutinas en C y reescribir el algoritmo que está implementando SeDuMi para que realice los cálculos de manera simbólica. Resolver el problema de los errores numéricos permitiría demostrar en Coq la no negatividad de los polinomios sin introducir ningún hipótesis extra. Una de las alternativas a analizar es la reescritura del algoritmo que implementa SeDuMi utilizando una representación simbólica[1] para los números, en lugar de la representación de punto flotante que se está usando actualmente.

[1] Mi trabajo de doctorado estará relacionado con la continuación y profundización de este tema.

Capítulo 4

Uso de los certificados

En el capítulo 3 mostramos un mecanismo para generar certificados de no negatividad para un polinomio dado p en un conjunto semialgebraico dado $\mathcal{K}$. En este capítulo se pretende demostrar la no negatividad en Coq a partir de las representaciones generadas.

Debe notarse que no podremos probar la no negatividad de p debido a que el certificado que se genera no es de p sino de un polinomio $\tilde{p}$ *parecido*, debido a errores numéricos introducidos por SeDuMi. Se pretende en el futuro eliminar este problema de alguna manera (véase 3.3). Con el objetivo de graficar esto, continuamos con el ejemplo sencillo del capítulo anterior. El problema dice pide *encontrar el mínimo valor que toma* $p(x_1, x_2) = x_1^2 + 2x_2^2$ *cuando* $(x_1, x_2) \in \mathcal{K} = \{(x_1, x_2) \mid x_1^2 - x_2 - 1 \geq 0, x_2 - x_1 \geq 0\}$. Lo que intentaremos en este capítulo será un problema ligeramente diferente. Intentaremos **demostrar en Coq que p es no negativo en** $\mathcal{K}$. Sin embargo, la forma que usaremos será *calcular una cota inferior para* p *en* $\mathcal{K}$ *usando los certificados generados en el capítulo anterior, y si esa cota resulta ser no negativa, entonces podremos concluir la no negatividad de* p *en* $\mathcal{K}$. El certificado exacto y el certificado generado por SeDuMi (*i.e.:* con errores numéricos) para este problema se muestran a continuación.

$$
\begin{aligned}
x_1^2 + 2x_2^2 = \quad & 0{,}8750 \\
& +(0)^2(x_2 - x_1) \\
& +(1)^2(x_1^2 - x_2 - 1) \\
& +\Big(\Big(0 - 0x_1 - 0x_2\Big)^2 \\
& \quad +\Big(0 + 0x_1 - 0x_2\Big)^2 \\
& \quad +\Big(\tfrac{1}{2\sqrt{2}} + 0x_1 + \sqrt{2}x_2\Big)^2\Big).
\end{aligned}
$$

$$
\begin{aligned}
x_1^2 + 2x_2^2 \approx\ & \tfrac{87500000020642221}{100000000000000000} \\
&+\left(\tfrac{8156757153382901}{295147905179352825856}\right)^2 (x_2 - x_1) \\
&+\left(\tfrac{4503599628137997}{4503599627370496}\right)^2 (x_1^2 - x_2 - 1) \\
&+\left(\left(\tfrac{4549305117837025}{59029581035870565 1712} - \tfrac{3582049525977251}{590295810358705651712} x_1 - \tfrac{4549305115835355}{2361183241434822606848} x_2\right)^2\right. \\
&+\left(\tfrac{3742010189156515}{295147905179352825856} + \tfrac{1262372114354249}{73786976294838206464} x_1 - \tfrac{7484020376380523}{2361183241434822606848} x_2\right)^2 \\
&\left.+\left(\tfrac{3184525835170645}{9007199254740992} + \tfrac{2327083786035767}{773712524553362671811952 64} x_1 + \tfrac{3184525836358511}{2251799813685248} x_2\right)^2\right).
\end{aligned}
$$

Al simplificar el lado derecho de la suma del certificado generado por SeDuMi no se obtiene exactamente $p(x_1, x_2) = x_1^2 + 2x_2^2$, sino el polinomio $\tilde{p}$ que se muestra a continuación.

$$
\begin{aligned}
\tilde{p}(\mathbf{x}) = \quad & -\tfrac{66861483600734399863531186158974773 47783}{265845599156983174580761412056068915200000000000000000} \\
& -\tfrac{7381620893576059179845478 6549}{6968928745408197317299119602026129706 1888} x_2 \\
& -\tfrac{1401498650201019840347747882 48519}{3484491437270409865864955980101306485309 44} x_1 \\
& +\tfrac{5986310710520345669275474599498119518156571478271953}{5986310706507378352962293074805895248510699696029696} x_1^2 \\
& +\tfrac{1547425048598860606712747 85}{6968928745408197317299119602026129706 1888} x_1 x_2 \\
& +\tfrac{5575186300005831758065016711344174473001225}{2787593149816327892691964784081045188247552} x_2^2
\end{aligned}
$$

Nótese que los coeficientes de x_1^2 y x_2^2 son aproximadamente 1 y 2, respectivamente; y nótese también que todos los demás coeficientes son aproximadamente 0. $\tilde{p}$ es *aproximadamente* igual a p, pero no es igual.

Sea $\varepsilon(\mathbf{x}) = p(\mathbf{x}) - \tilde{p}(\mathbf{x})$. Podemos calcular ε simbólicamente, de manera que valga *exactamente* $p - \tilde{p}$. En nuestro ejemplo,

$$
\begin{aligned}
\varepsilon(x_1, x_2) \quad & = P(x_1, x_2) - \tilde{P}(x_1, x_2) \\
& = \tfrac{66861483600734399863531186158974773 47783}{265845599156983174580761412056068915200000000000000000} \\
& -\tfrac{3731759726810871431820840965061 21}{2787593149816327892691964784081045188247552} x_2^2 \\
& -\tfrac{4012967316313181524692224269645871782242257}{5986310706507378352962293074805895248510699696029696} x_1^2 \\
& +\tfrac{7381620893576059179845478 6549}{6968928745408197317299119602026129706 1888} x_2 \\
& +\tfrac{1401498650201019840347747882 48519}{3484491437270409865864955980101306485309 44} x_1 \\
& -\tfrac{1547425048598860606712747 85}{6968928745408197317299119602026129706 1888} x_1 x_2
\end{aligned}
$$

lo que permite notar que los errores numéricos introducidos por SeDuMi son pequeños pero hacen que la expresión pierda la sencillez y que no sea verdadera la igualdad $p = \tilde{p}$. Sin embargo, la igualdad que sí es verdadera es

$$\forall \mathbf{x} \bullet p(\mathbf{x}) = \tilde{p}(\mathbf{x}) + \varepsilon(\mathbf{x}).$$

Si la función ε es siempre positiva, entonces la cota inferior encontrada por SeDuMi es válida (en nuestro caso 0,87500000020642221). Sin embargo, si ε toma valores negativos tendremos que asegurarnos de que $\tilde{p} + \varepsilon$ es mayor o igual que la cota propuesta.

Si tomamos una cota inferior β tal que $\varepsilon(\mathbf{x}) \geq \beta \ \forall \mathbf{x} \in \mathcal{K}$, podremos asegurar que $p(\mathbf{x}) = \tilde{p}(\mathbf{x}) + \varepsilon(\mathbf{x}) \geq \tilde{p}(\mathbf{x}) + \beta$ en $\mathcal{K}$.

Nótese que si $\beta \leq 0$ (que es lo que se espera que ocurra en la mayoría de los casos), la cota inferior para p en $\mathcal{K}$ que podrá demostrarse será menor o igual a la encontrada por SeDuMi. En nuestros ejemplos, calcularemos β utilizando un software auxiliar, y luego agregaremos a Coq esa cota inferior para el error en forma de hipótesis. El software que se utiliza para la búsqueda de la cota es Amber ([15]). Este programa implementa conversión a bases de Bernstein y aritmética de intervalos; es muy adecuado para trabajar con polinomios de ceoficientes muy pequeños, como es nuestro caso. Sin embargo, como ya se dijo, se pretende a futuro reemplazar este paso haciendo que el error sea nulo; posiblemente a través de la reescritura del algoritmo implementado por SeDuMi pero utilizando una representación simbólica para los números.

El conjunto de problemas demostrables de esta manera es claramente menor que el conjunto de problemas demostrables en el caso en el que el error es nulo, pero por ahora se utilizará este método. El razonamiento recién expuesto aplicado a nuestro ejemplo permite ver claramente uno de los mayores problemas que introducen los errores numéricos: el polinomio diferencia tomará valores muy grandes en valor absoluto cuando sus variables también lo hagan. En conjuntos no acotados, esto puede hacer que no exista una cota inferior β, y en conjuntos acotados podría ocurrir que la cota exista, pero que el β elegido lleve a que no podamos demostrar la no negatividad del polinomio en cuestión.

En nuestro ejemplo, con el certificado generado por SeDuMi recién expuesto (que conduce a la función ε también recién expuesta), no podemos asegurar que $p \geq 0$ en $\mathcal{K}$, ya que el ε no está acotado inferiormente en $\mathcal{K}$. Afortunadamente este problema que ocurre en nuestro ejemplo no ocurrirá en los problemas de la demostración de Hales debido a que los conjuntos sobre los que intentaremos minimizar los polinomios serán conjuntos acotados. El conjunto $\mathcal{K}$ sobre el que estamos intentando minimizar p en nuestro ejemplo no es acotado, de manera que el polinomio ε toma valores tan bajos como querramos en $\mathcal{K}$. Si bien en los problemas de la demostración de Hales no ocurrirá esto, habrá algo que sí ocurrirá y que no es bueno que ocurra. Sea PC (abreviatura de *PROBLEMA CUALQUIERA*) un problema cualquiera de la demostración de Hales, que pida demostrar que $p_{PC} \geq 0$ en $\mathcal{K}_{PC}$, y sea t la cota inferior propuesta por SeDuMi para p_{PC} en $\mathcal{K}_{PC}$. Siguiendo el procedimiento que recién explicado, en el intento por demostrar que $p_{PC} \geq 0$ en $\mathcal{K}_{PC}$ buscaremos el mínimo β_{PC} del polinomio diferencia ε_{PC} y sólo podremos demostrar que que $p \geq \beta_{PC} + t$ en $\mathcal{K}_{PC}$. Podría ocurrir que $\beta_{PC} + t < 0$. En esos casos no podremos demostrar la no negatividad de p_{PC} en $\mathcal{K}_{PC}$. En algunas de esas situaciones realmente se debe a que p_{PC} no es no negativo en $\mathcal{K}_{PC}$, pero en otros casos estaremos frente a polinomios que efectivamente son no negativos y no podremos demostrarlo. Este segundo conjunto de casos se transformará en vacío cuando se eliminen los errores numéricos.

Con el objetivo de mantener la simplicidad en el ejemplo, agreguemos una restricción extra para hacer que el conjunto sobre el que queremos minimizar p sea acotado. Consideremos que

$$x_1^2 + x_2^2 \leq 10^2$$

Al agregar esta restricción, obtenemos que que una cota inferior para ε es $\beta = -0{,}00000000077103753031812334$, que se obtiene en

$$(x_1, x_2) = (-0{,}779551903579300710, -0{,}523850199151939640).$$

De manera que

$$p(x_1, x_2) = \tilde{p}(x_1, x_2) + \varepsilon(x_1, x_2) \geq \tilde{p}(x_1, x_2) - 0{,}00000000077103753031812334$$

La cota inferior que había encontrado SeDuMi para nuestro problema (sin la restricción que agregamos) era 0,87500000020642221 (con la nueva restricción que acabamos de agregar esa cota inferior continía siendo válida), de manera que estaremos en condiciones de probar (en Coq) que $p(x_1, x_2) \geq 0{,}87500000020642221 - 0{,}00000000077103753031812334 = 0{,}87499999435384650$

Nótese que podremos demostrar que $p \geq 0{,}87499999435384650$ en lugar de que $p \geq 0{,}8750$. En cualquiera de los casos, podremos demostrar que $p \geq 0$[1]. Esto mismo ocurrirá en la mayoría de los problemas de la demostración de Hales, pero, como ya dijimos, podrán existir situaciones en las que la presencia de este error numérico transforme un problema que sería demostrable mediante este procedimiento en uno que no lo es.

A continuación se muestra la demostración en Coq de que $p(x_1, x_2) = x_1^2 + 2x_2^2 \geq 0$ cuando $x_1^2 - x_2 - 1 \geq 0$, $x_2 - x_1 \geq 0$ y $100 - x_1^2 - x_2^2 \geq 0$. El mismo mecanismo y la misma demostración que proponemos en la figura sirven para cualquiera de los problemas propuestos en la demostración de Hales. El mecanismo fue:

1. Calcular una cota inferior t para el polinomio p dado, en el conjunto semialgebraico dado $\mathcal{K}$, dando una representación que permita concluir esa cota en Coq. En este momento, la representación y la cota no serán de p sino de $\tilde{p}$.
2. Calcular de manera exacta el polinomio $\varepsilon = p - \tilde{p}$.
3. Calcular una cota inferior β para ε en $\mathcal{K}$.
4. Demostrar (en Coq) que $p \geq t + \beta$ en $\mathcal{K}$.
5. (sólo será posible (en este trabajo) si $t + \beta \geq 0$) Concluir en Coq que $p \geq 0$ en $\mathcal{K}$.

[1] Sin embargo, la cota 0,8750 es una cota inferior *mejor*, ya que intentamos buscar la mayor de las cotas inferiores.

```
Require Import Reals.

Open Scope R_scope.

Ltac split_plus :=
 match goal with
 |- 0 <= _ + _ => apply Rplus_le_le_0_compat; [ try split_plus |]
 end.

Ltac split_mult :=
 match goal with
 |- 0 <= _ * _ => apply Rmult_le_pos; [ try split_mult |]
 end.

Lemma square_ge_0 : forall a, 0 <= a^2.
Proof.
 intro; simpl; rewrite Rmult_1_r; apply Rle_0_sqr.
Qed.

Ltac kill :=
 split_plus; try split_mult; try assumption; try split_plus;
 apply square_ge_0.

Ltac split_goal :=
 match goal with
 |- 0 <= ?x + _ + _ => rewrite (Rplus_comm x), Rplus_assoc;
                       apply Rplus_le_le_0_compat
 end.

Ltac t1 :=
 match goal with
 |- _ <= ?x => field_simplify (x)
 end.

Ltac t2 :=
 match goal with
 |- 0 <= ?a / ?b => apply(Rmult_le_reg_l b); prove_sup;
                   unfold Rdiv
 end.

Ltac t3 :=
 match goal with
 |- _ <= ?a * (?b * ?c) => rewrite <- (Rmult_assoc a b c)
 end.

Ltac t4 :=
 match goal with
 |- _ <= ?a * ?b * _ => rewrite (Rinv_r_simpl_m a b)
 end.


Section TEST.
 Variables x1 x2 : R.
```

```
Hypothesis h1 : 0 <= (x2 - x1).
Hypothesis h2 : 0 <= (x1^2-x2-1).

Definition p := x1^2 + 2*x2^2.
Definition pTilde_BOUND := 87500000020642221/100000000000000000.
Definition pTilde_BODY :=   (8156757153382901/295147905179352825856)^2 * (x2 - x1)
                          + (4503599628137997/4503599627370496)^2 * (x1^2 - x2 - 1)
                          + (
                             (
                              4549305117837025/590295810358705651712
                              - (3582049525977251/590295810358705651712) * x1
                              - (4549305115835355/2361183241434822606848) * x2
                             )^2
                            +
                             (
                              3742010189156515/295147905179352825856
                              + (1262372114354249/73786976294838206464) * x1
                              - (7484020376380523/2361183241434822606848) * x2
                             )^2
                            +
                             (
                              3184525835170645/9007199254740992
                              + (2327083786035767/773712524553362671811952 64) * x1
                              + (3184525836358511/2251799813685248) * x2
                             )^2
                            ).
Definition pTilde := pTilde_BOUND + pTilde_BODY.
Definition e := 66861483600734399863531186158974773477 83 /
                 265845599156983174580761412056068915200000000000000000 00
                - 37317597268108714318208409650612 1 /
                  27875931498163278926919647840810451881 88247552 * x2^2
                - 40129673163131815246922242696458717822422 57 /
                  59863107065073783529622930748058952485106996960296 96 * x1^2
                + 73816208935760591798454786549 /
                  69689828745408197317299119602026129706188 8 * x2
                + 14014986502010198403477478824851 9 /
                  34844914372704098658649559801013064853094 4 * x1
                - 15474250485988606067127478 5 /
                  69689828745408197317299119602026129706188 8 * x1 * x2.

Theorem igualdad : p = pTilde + e.
 unfold p.
 unfold pTilde.
 unfold pTilde_BOUND.
 unfold pTilde_BODY.
 unfold e.
 field.
Qed.

Definition cota_e := -77103753031812334/1000000000000000000000000000.
Hypothesis cota_inferior_e : cota_e <= e.

Goal 0 <= p.
  rewrite igualdad.
```

```
    unfold pTilde.
    apply Rle_trans with (pTilde_BOUND + pTilde_BODY + cota_e);
      [ | apply Rplus_le_compat_l; trivial ].
    split_goal.
    unfold pTilde_BODY.
    kill.
    unfold pTilde_BOUND.
    unfold cota_e.
    t1 ; t2 ; t3 ; t4.
    rewrite Rmult_0_r. left. prove_sup. discrR.
  Qed.
End TEST.
```

Apéndice A

Tres ejemplos

En el presente capítulo se muestran algunos de los ejemplos del proyecto Flyspeck en los que hay que demostrar desigualdades polinomiales bajo restricciones dadas.

En los primeros tres ejemplos usaremos las siguientes definiciones:

$$t_0 := \frac{251}{200}$$

$$\mathbf{X}_{751442360} := \left([(2t_0)^2\,; (\tfrac{337}{125})^2], [4; (\tfrac{271}{125})^2], [4; (\tfrac{271}{125})^2], [4; (2t_0)^2], [4; (2t_0)^2], [4; (2t_0)^2]\right)$$

$$\mathrm{perm}_2 x := (x_2, x_1, x_3, x_4, x_5, x_6)$$

$$\Delta x := \frac{1}{2}\begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & x_3 & x_2 & x_1 \\ 1 & x_3 & 0 & x_4 & x_5 \\ 1 & x_2 & x_4 & 0 & x_6 \\ 1 & x_1 & x_5 & x_6 & 0 \end{bmatrix}$$

$$\begin{aligned} = {} & x_1x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) \\ & + x_2x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) \\ & + x_3x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) \\ & - x_2x_3x_4 - x_1x_3x_5 - x_1x_2x_6 - x_4x_5x_6. \end{aligned}$$

A.1.

Probar que

$$\forall x \in \mathbf{X}_{751442360} \quad \bullet \quad 0 < 4x_2 \cdot \Delta\left(\mathrm{perm}_2 x\right)$$

Se puede reescribir el problema expandiendo $\Delta\left(\mathrm{perm}_2 x\right)$, y se obtiene

$$\forall (x_1, x_2, x_3, x_4, x_5, x_6) \in \mathbf{X}_{751442360} \quad \bullet$$

$$\begin{aligned} 0 < {} & -x_2^2x_5 + x_2x_5x_1 + x_2x_5x_3 - x_2x_5^2 + x_2x_5x_4 + x_2x_5x_6 \\ & + x_1x_4x_2 - x_1^2x_4 + x_1x_4x_3 + x_1x_4x_5 - x_1x_4^2 + x_1x_4x_6 \\ & + x_3x_6x_2 + x_3x_6x_1 - x_3^2x_6 + x_3x_6x_5 + x_3x_6x_4 - x_3x_6^2 \\ & - x_1x_3x_5 - x_2x_3x_4 - x_2x_1x_6 - x_5x_4x_6 \end{aligned}$$

A.2.

Probar que

$$\left(\partial_4 \Delta \left(\mathrm{perm}_2 \mathbf{x}\right)\right)^2 < \frac{61}{51} \cdot 4x_2 \cdot \Delta \left(\mathrm{perm}_2 \mathbf{x}\right).$$

cuando $\mathbf{x} \in \mathbf{X}_{751442360}$.

A.3.

Probar que

$$\begin{aligned} p(x_1, x_2, x_3, x_4) = {} & 100x_1^4 + 90x_3^4 - 200x_1^2 * x_2 - 180x_3^2 x_4 + x_1^2 + 110{,}1x_2^2 \\ & + x_3^2 + 100{,}1x_4^2 + 19{,}8x_2 x_4 - 2x_1 - 40x_2 - 2x_3 - 40x_4 + 42 \end{aligned}$$

es no negativo en $\mathcal{K} = [-10, 10] \times [-10, 10] \times [-10, 10] \times [-10, 10]$.

Bibliografía

[1] Yves Bertot and Pierre Casteran. *Interactive Theorem Proving and Program Development*. SpringerVerlag, 2004.

[2] Ruchira S. Datta. Using semidefinite programming to minimize polynomials. Available at math.berkeley.edu/~datta/ee227apaper.pdf, 2001.

[3] Miguel de Guzmán. *Cuentos con cuentas*. Red olímpica, 1996.

[4] G. Gonthier. A computer checked proof of the four-color theorem. Available on the web, 2005.

[5] Benjamin Grégoire, Laurent Théry, and Benjamin Werner. A computational approach to pocklington certificates in type theory. In Hagiya and Wadler [6], pages 97–113.

[6] Masami Hagiya and Philip Wadler, editors. *Functional and Logic Programming, 8th International Symposium, FLOPS 2006, Fuji-Susono, Japan, April 24-26, 2006, Proceedings*, volume 3945 of *Lecture Notes in Computer Science*. Springer, 2006.

[7] T. Hales. Flyspeck project. http://www.flyspeck-blog.blogspot.com/.

[8] T. Hales. The kepler conjecture. http://www.math.pitt.edu/~thales/kepler98/.

[9] Cristoph Helmberg. *Semidefinite Programming for Combinatorial Optimization*. SpringerVerlag, 2000.

[10] Lasserre. Notes de cours. Chapters 2-3, 2008.

[11] Elon Lages Lima. *Álgebra Linear*. Instituto de Matemática Pura e Aplicada, 2001.

[12] Pablo A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming*, 96(2):293–320, Mayo 2003.

[13] Bruce Reznick. Some concrete aspects of hilbert 17th problem. *Contemporary Mathematics*, 253:251–272, 2000.

[14] Bruce Reznick. Extremal psd forms with few terms. *Duke Math J.*, 2001.

[15] Roland Zumkeller. Amber. http://roland.zumkeller.googlepages.com/software.

[16] Roland Zumkeller. *Global optimization in Type Theory*. PhD thesis, Ecole Polytechnique, Paris, 2008.