

MINIMIZACIÓN GLOBAL DE UN POLINOMIO EN LA RECTA REAL

C. BELTRÁN ROYO

Universitat Politècnica de Catalunya*

En este artículo presentamos y probamos numéricamente un nuevo algoritmo para la minimización global de un polinomio de grado par. El algoritmo está basado en la simple idea de trasladar verticalmente el grafo del polinomio hasta que el eje OX sea tangente al grafo del polinomio trasladado. En esta privilegiada posición, cualquier raíz real del polinomio trasladado es un mínimo global del polinomio original.

Globally minimizing a polynomial on the real line.

Palabras clave: Optimización global, máximo común divisor de polinomios, algoritmo de Euclides, secuencia de Sturm, división sintética de dos polinomios.

Clasificación AMS (MSC 2000): 49J05

*Dept. d'Estadística i Investigació Operativa. Secció d'Informàtica. Universitat Politècnica de Catalunya. Pau Gargallo, 5. 08028 Barcelona (Espanya).

–Recibido en octubre de 1997.

–Aceptado en julio de 1998.

1. RESUMEN

Dado un polinomio $P(x)$ de grado par, queremos resolver el problema de minimizar *globalmente* $P(x)$ en el conjunto de los reales (problema P-1). Para resolver el problema hemos diseñado un algoritmo o método nuevo denominado método del máximo común divisor (MCD).

El algoritmo del máximo común divisor se basa en:

- a) Conocer el número de raíces reales diferentes del polinomio sin tener que calcularlas (método de Sturm).
- b) En el cálculo del máximo común divisor de dos polinomios que denotamos por MCD.

La idea del método es trasladar verticalmente el polinomio P hasta conseguir que el eje OX sea tangente al grafo de P trasladado (denotamos por P^* este P trasladado). Una vez ya tenemos P^* calculamos el máximo común divisor de P^* y su derivada $P^{*'} (Q = \text{MCD}(P^*, P^{*'}))$ que en la mayoría de los casos será un polinomio de grado uno. Finalmente, cualquier raíz del polinomio Q es un mínimo global del polinomio P .

Para poner en práctica el anterior esquema usaremos conocidos métodos y algoritmos:

- a) Vía método de Sturm podemos conocer el número de raíces reales distintas de P y un uso iterativo de esta información nos conducirá a P^* .
- b) Vía el algoritmo de Euclides calcularemos Q .
- c) La división sintética de dos polinomios será requerida para aplicar el algoritmo de Euclides.
- d) Finalmente usaremos el método de Birge-Vieta para obtener una raíz de Q .

2. PRECEDENTES

En el artículo Goldstein (1971) resuelven el problema P-1 de forma directa i.e. se obtiene un mínimo global extrayendo de P cada mínimo local encontrado vía división de polinomios. Esta estrategia puede llevar a tener que resolver un gran número de minimizaciones locales, además de que si no se hace minimización local exacta, los errores acumulados pueden degradar muy seriamente la solución.

Otros autores, resuelven el problema P-1 localmente i.e. minimizan *globalmente* $P(x)$ en un intervalo pero no en todo \mathbb{R} (denominamos este problema P-2). Repasemos tres trabajos:

- 1) En el artículo Wingo (1985) se propone un algoritmo sencillo para el problema P-2 que no requiere la evaluación de derivadas.
- 2) En el artículo Floudas (1992) se resuelve el problema P-2 vía programación lineal con restricciones no lineales. La idea del método es cambiar en el polinomio cada potencia de x en una nueva variable. Así pasan de la minimización unidimensional de $P(x)$ en un intervalo (a, b) , a la minimización de una función lineal $c'x$ en un dominio de \mathbb{R}^{n+1} .
- 3) En el artículo Bromberg (1992) se propone un algoritmo para la minimización global de una función $f(x)$ en un intervalo (a, b) . La idea de su método consiste en reducir el intervalo inicial (a, b) aplicando técnicas de minimización local.

Por lo tanto, a excepción del trabajo de Goldstein (1971), ninguno de los trabajos citados resuelve el problema de minimizar $P(x)$ en toda la recta real (problema P-1). El método que ahora se propone resuelve el problema P-1 completamente.

3. NOTACIÓN

- Consideramos polinomios $P(x) = a_1 \cdot x^n + a_2 \cdot x^{n-1} + \dots + a_n \cdot x + a_{n+1}$ de coeficientes reales y variable real.
- Con $P(x, c)$ denotemos el polinomio que tiene $a_{n+1} = c$ i.e.

$$P(x, c) = a_1 \cdot x^n + a_2 \cdot x^{n-1} + a_3 \cdot x^{n-2} + \dots + a_n \cdot x + c$$

- Conjunto de las raíces reales del polinomio P

$$\text{Raíces}(P) = \{x \text{ de } \mathbb{R} / P(x) = 0\}$$

- Cardinal de raíces(P) $\# \text{Raíces}(P)$
- Grafo de un polinomio $\text{grafo}(P) = \{(x, P(x)) \text{ en } \mathbb{R}^2 / x \text{ en } \mathbb{R}\}$
- Semiespacio positivo $S^+ = \{(x, y) \text{ de } \mathbb{R}^2 / y \text{ positivo}\}$
- Semiespacio negativo $S^- = \{(x, y) \text{ en } \mathbb{R}^2 / y \text{ negativo}\}$

4. MODELO PROPIO - RESULTADOS TEÓRICOS

En todo el trabajo se está presuponiendo:

- La minimización de un polinomio $P(x)$ de grado par, dado que un polinomio de grado impar no está acotado inferiormente en \mathbb{R} . Por la misma razón, supondremos que el coeficiente a_1 es positivo.

- La minimización de $P(x)$ se hace en todo \mathbb{R} .
- Todas las raíces de $P(x)$ que se consideran son reales.
- Un polinomio puede alcanzar el mínimo global en más de un punto. El método desarrollado da como resultado solamente uno de estos puntos, aunque el método fácilmente se puede adaptar para dar todos los puntos donde el polinomio alcanza el mínimo absoluto.

A continuación veremos unas definiciones, unos procedimientos y unos resultados que están demostrados en el reporte de investigación Beltran (1997) y que usaremos en el algoritmo del MCD.

La idea del algoritmo es en síntesis: dado un polinomio $P(x)$ de grado par, siempre podemos trasladarlo verticalmente de forma que $\text{Grafo}(P)$ sea tangente al eje OX obteniendo así el polinomio $P^*(x)$. Este último polinomio tiene la característica de que cualquiera de sus raíces reales es un punto de mínimo global de $P^*(x)$ y de $P(x)$ en todo \mathbb{R} . La forma de buscar una raíz cualquiera de $P^*(x)$ se hace teniendo en cuenta que en cada punto donde se anula $P^*(x)$ también se anula su derivada $P^{*'}(x)$. Dado que los ceros comunes de $P^*(x)$ y $P^{*'}(x)$ coinciden con los ceros del polinomio $\text{MCD}(P^*, P^{*'})$, cualquier cero de este último polinomio será un mínimo global de $P^*(x)$ y por tanto de $P(x)$.

4.1. Polinomios pobres, ricos y buenos

Definición DES-1 (polinomio pobre, rico y bueno):

Consideremos un polinomio de grado par y con $a_1 > 0$:

$$P(x) = a_1 \cdot x^n + a_2 \cdot x^{n-1} + a_3 \cdot x^{n-2} + \dots + a_n \cdot x + a_{n+1}$$

- $P(x)$ es pobre si el eje OX no corta el grafo(P) (escribiremos P^-).
- $P(x)$ es rico si el eje OX corta el grafo (P) (escribiremos P^+).
- $P(x)$ es bueno si el eje OX es tangente al grafo(P) (escribiremos P^*).

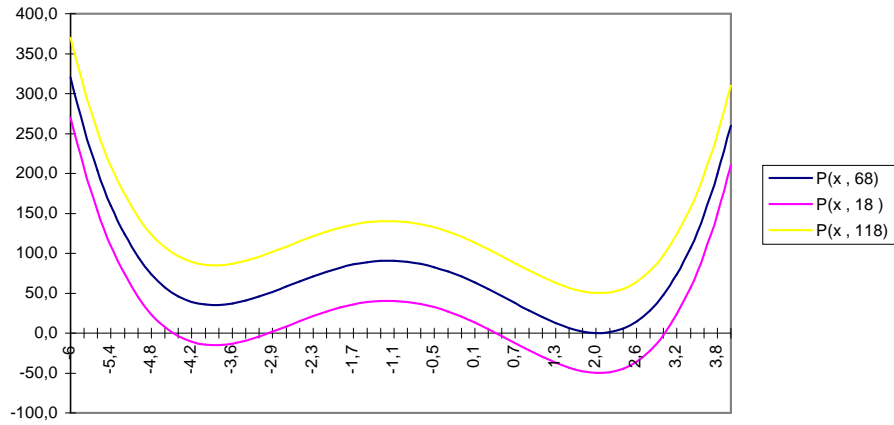
Corolario DES-1. Dado un polinomio $P(x)$ de grado n :

- Siempre existe una traslación vertical de $P(x)$ que lo convierte en polinomio bueno $P^*(x)$. Además $P(x)$ difiere de $P^*(x)$ solamente en el coeficiente a_{n+1} .
- x^* es un óptimo global de $P^*(x)$ si y solo si x^* es un óptimo global de $P(x)$. Por tanto podemos limitar la búsqueda de óptimos globales en el caso de polinomios buenos.

EJEMPLO REP-1.

$$P(x, c) = x^4 + 4x^3 - 11x^2 - 36x + c$$

$P(x, 68)$	polinomio bueno	(1 raíz real)
$P(x, 18)$	polinomio rico	(4 raíces reales)
$P(x, 118)$	polinomio pobre	(0 raíces reales)



4.2. Obtención de un óptimo global de un polinomio bueno $P^*(x)$

Teorema DES-1. (caracterización de óptimo global vía MCD) Sean:

- $P^*(x)$ polinomio bueno y $P^{*'}(x)$ su derivada,
- $M(x)$ polinomio definido como el MCD(P^* , $P^{*'}$),
- x^* punto de \mathbb{R} .

Entonces:

$$M(x^*) = 0 \quad \text{sii} \quad x^* \quad \text{mínimo global de} \quad P^*(x).$$

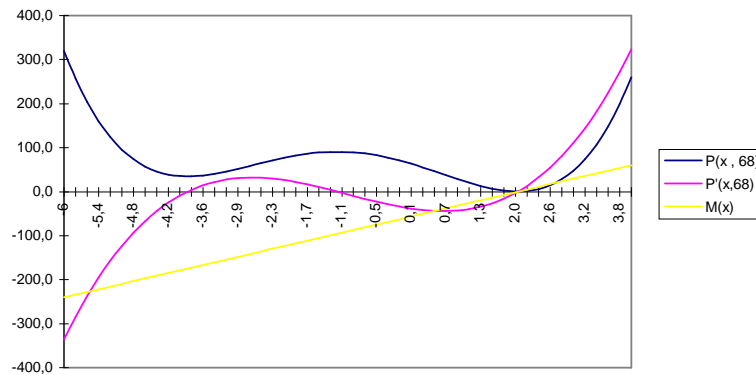
EJEMPLO REP-2. Continuamos con el polinomio del ejemplo REP-1.

$$\begin{aligned} P^*(x) &= x^4 + 4x^3 - 11x^2 - 36x + 68 \\ P^{*'}(x) &= 4x^3 + 12x^2 - 22x - 36 \\ M(x) &= \text{MCD}(P^*, P^{*'}) = (x - 2) \end{aligned}$$

En el siguiente gráfico se ve como:

- $x = 2$ es el mínimo global de $P(x, 68)$.
- $x = 2$ es el único real donde se anula simultáneamente P^* y $P^{*'}$.
- por lo tanto $(x - 2)$ es un factor de M (en realidad el único) y así $M(2) = 0$.
- Podemos concluir que $x^* = 2$ óptimo global de $P^*(x)$ y por lo tanto de $P(x)$.

NOTA: En lugar de representar $y = M(x)$ hemos representado $y = 30 \cdot M(x)$, pues la representación de $M(x)$ queda casi pegada al eje $O\bar{X}$.



4.3. Obtención del polinomio bueno P^* asociado a un polinomio P

Tenemos dos tipos de información que usaremos en el algoritmo OGP.

4.3.1. Primera información - $\#Raíces(P)$

El siguiente resultado puede encontrarse en Henrici (1974) pág. 448.

Teorema de Sturm DES-2. (Número de ceros reales y distintos de $P(x)$ en (a, b)). Definimos $P_0(x) := P(x)$ y $P_1(x) := P'(x)$. Sea $\{P_0, P_1, P_2, \dots, P_m\}$ la secuencia de polinomios obtenidos mediante el algoritmo de Euclides para calcular el $MCD(P_0, P_1)$ (ver apéndice). Entonces:

El número de ceros reales y distintos de P_0 en el intervalo (a, b) es igual a la diferencia entre el número de cambios de signo en $\{P_0(a), P_1(a), P_2(a), \dots, P_m(a)\}$ y el número de cambios de signo en $\{P_0(b), P_1(b), P_2(b), \dots, P_m(b)\}$ suponiendo que $P_0(a) \cdot P_0(b)$ sea diferente de cero.

Definición DES-2. (Secuencia de Sturm asociada a un polinomio). El conjunto de polinomios $\{P_0, P_1, P_2, \dots, P_m\}$ que aparece en el enunciado del teorema anterior, se denomina secuencia de Sturm asociada a P_0 . Escribiremos $\text{Sturm}(P)$ para hacer referencia a la secuencia de Sturm asociada al polinomio $P(x)$.

EJEMPLO REP-3. Continúa del ejemplo REP-2.

$$P^*(x) = x^4 + 4x^3 - 11x^2 - 36x + 68$$

Veamos $\text{Sturm}(P^*)$:

$$\begin{aligned} P_0(x) &:= P^*(x) = x^4 + 4x^3 - 11x^2 - 36x + 68 \\ P_1(x) &:= P^{*'}(x) = 4x^3 + 12x^2 - 22x - 36 \\ P_2(x) &= 8.5x^2 - 21.5x - 77 \\ P_3(x) &= -9.4x + 18.8 = -9.4 \cdot (x - 2) \\ P_4(x) &= 0 \end{aligned}$$

$$\begin{aligned} N(-\infty) &:= \text{Cambios de signo en } \{P_0(-\infty), P_1(-\infty), P_2(-\infty), P_3(-\infty), P_4(-\infty)\} = \\ &= \text{Cambios de signo en } \{+, -, +, +, 0\} = \\ &= 2 \end{aligned}$$

$$\begin{aligned} N(\infty) &:= \text{Cambios de signo en } \{P_0(\infty), P_1(\infty), P_2(\infty), P_3(\infty), P_4(\infty)\} = \\ &= \text{Cambios de signo en } \{+, +, +, -, 0\} = \\ &= 1 \end{aligned}$$

Entonces por el teorema de Sturm tenemos que $\#Raíces(P^*) = N(-\infty) - N(\infty) = 1$ y podemos asegurar que P^* es polinomio bueno. Podemos ver la representación de $P^*(x) = P(x, 68)$ en Ejemplo REP-1 donde efectivamente se ve que P^* tiene una sola raíz real.

Nótese que al ser $P_4(x) = 0$ entonces $P_3(x) = -9.4 \cdot (x-2)$ nos da el $MCD(P^*, P^{*'}) = (x - 2)$.

4.3.2. Segunda información - la función residuo

Definición DES-4. (función residuo asociada a un polinomio). Dada la familia de polinomios $F = \{P(x, c) : c \text{ de } \mathbb{R}\}$ definimos la función residuo $R_P(c)$ asociada a F , como el resto de la división $P_{m-1}(x, c) : P_m(x, c)$ donde P_{m-1} y P_m son los dos últimos polinomios no constantes de Sturm $(P(x, c))$.

Corolario DES-4. (Carácter de un polinomio vía función residuo). Sea c^* la mayor raíz real de $R_P(c)$. Entonces:

- a) $P(x, c)$ pobre sii $c > c^*$.
- b) $P(x, c)$ bueno sii $c = c^*$.
- c) $P(x, c)$ rico sii $c < c^*$.

EJEMPLO REP-4. Continúa del ejemplo REP-3.

$$P(x, c) = x^4 + 4x^3 - 11x^2 - 36x + c$$

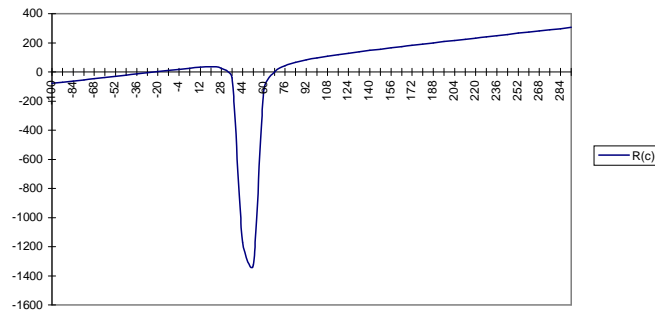
Con la ayuda del paquete «Mathematica» (Wolfram (1992)) hemos calculado $R_P(c)$ obteniendo la fracción algebraica siguiente:

$$R_P(c) = A(c)/B(c)$$

donde

$$\begin{aligned} A(c) &= 289 \cdot (16c^3 - 1256c^2 - 483c + 809676) \\ B(c) &= (68c - 3255)^2 \end{aligned}$$

En la siguiente representación de $R_P(c)$ se ve como su mayor raíz real es $c^* = 68$, que confirma el hecho de que $P(x, 68)$ sea un polinomio bueno (como vimos en el ejemplo REP-1).



5. MÉTODO DEL MCD PARA LA MINIMIZACIÓN GLOBAL DE UN POLINOMIO

Dado un polinomio de grado par $P(x)$, buscamos un x^* mínimo global de $P(x)$ en \mathbb{R} . Las etapas del método del MCD se explican en 5.1, 5.2 y 5.3.

Entonces, en virtud del teorema DES-1, x^* es un óptimo global de $P(x)$ en \mathbb{R} .

5.1. Obtención del polinomio bueno $P^*(x)$ asociado a un polinomio $P(x)$

Aplicando el corolario DES-4 es suficiente encontrar c^* raíz máxima de la función residuo $R_P(c)$ y entonces $P(x, c^*)$ polinomio bueno asociado a P^* . Estamos interesados, pues, en un intervalo (a_0, b_0) que contenga c^* . Dado $P(x)$ hacemos:

5.1.1. Cálculo del intervalo inicial (a_0, b_0) .

5.1.2. Cálculo de c^* .

5.1.1. Cálculo del intervalo inicial tal que contenga c^*

Vía el teorema de Sturm, podemos conocer $\#Raíces(P(x, c_k))$ $k = 0, 1, 2, 3, \dots$ y así determinar el carácter rico, pobre o bueno de cada $P(x, c_k)$. Podemos incrementar c_k hasta que lleguemos a un $P(x, c_p)$ polinomio pobre y así tendremos garantizado $c^* < c_p$.

Tomaremos $b_0 = c_p$ como extremo superior del intervalo inicial.

En resumen, hemos empobrecido el polinomio inicial $P(x, c_0)$ para buscar b_0 .

Por otra parte, dado que $P(x, 0)$ siempre tiene una raíz real en $x = 0$, podemos tomar como extremo inferior del intervalo inicial, cualquier valor a_0 positivo, lo más cerca de b_0 posible y de forma que $P(x, a_0)$ sea un polinomio rico.

EJEMPLO REP-5. Continúa del ejemplo REP-4.

Dado que $P(x, 18)$ es un polinomio rico, después de 2 iteraciones de la rutina EMPORBINT, obtenemos $P(x, 72)$ que es un polinomio pobre. De esta forma tenemos que $18 < c^* < 72$ con lo cual podemos tomar $(18, 72)$ como intervalo inicial.

A continuación tenemos Sturm ($P(x, \underline{72})$).

***** EMPOBRINT 0

Polinomio a optimizar $P(x, 18)$

1.00 4.00 -11.00 -36.00 18.00

OGP/out

sturm

1.00	4.00	-11.00	-36.00	<u>72.00</u>	P_0
.00	4.00	12.00	-22.00	-36.00	P_1
.00	.00	8.50	1.50	-81.00	P_2
.00	.00	.00	11.36	18.06	P_3
.00	.00	.00	.00	25.30	P_4

***** EMPOBRINT 2

5.1.2. Cálculo de c^*

5.1.2.1. *Primera información, #Raíces(P)*. Vía teorema de Sturm podemos conocer el número de raíces de cada polinomio $P(x, c_k)$. Al mismo tiempo por el corolario DES-4 sabemos que c^* es un valor límite en el sentido de que si $c < c^*$ entonces $P(x, c)$ es un polinomio rico, y en cambio, si $c > c^*$ entonces $P(x, c)$ es un polinomio pobre. Un primer algoritmo básico para calcular c^* , sería seguir un esquema tipo bisección con la información $\#Raíces(P(x, c_k))$ $k = 0, 1, 2, \dots$, de manera que si $P(x, c_k)$ es pobre, disminuiríamos c_k , y en cambio, si $P(x, c_k)$ es rico, aumentaríamos c_k i.e. vamos trasladando verticalmente el grafo de P hasta que sea tangente al eje OX . Siguiendo este proceso de forma iterativa, acabaríamos obteniendo $P(x, c_n)$ polinomio bueno.

5.1.2.2. *Segunda información, la función residuo*. Recordemos la definición de función residuo asociada a un polinomio (definición DES-4):

Dada la familia de polinomios $F = \{P(x, c) : c \text{ de } \mathbb{R}\}$ definimos la función residuo $R_P(c)$ asociada a F , como el residuo de la división $P_{m-1}(x, c) : P_m(x, c)$ donde P_{m-1} y P_m son los dos últimos polinomios no constantes de Sturm ($P(x, c)$).

La forma de utilizar la información de la función residuo $R_P(c)$ nos la da el siguiente teorema:

Teorema DES-3. (Caracterización de polinomio bueno vía la función de residuo $R_P(c)$).
Sea $F = \{P(x, c) : c \text{ de } \mathbb{R}\}$ y definimos $P^*(x) = P(x, c^*)$, entonces:

P^* polinomio bueno sii c^* es la mayor raíz real de $R_P(c)$.

Se trata pues de calcular la mayor raíz de $R_P(c)$.

Llegados hasta este punto, de las alternativas para calcular esta raíz hemos escogido el método de la secante por:

- No disponer de $R_P(c)$ de forma explícita ni de su derivada. El valor de $R_P(c_0)$ se obtiene como un subproducto al calcular Sturm ($P(x, c_0)$).
- El método se ha mostrado robusto y rápido en la práctica.

El método de la secante está descrito en el apéndice y programado en la rutina SECANT.

5.1.2.3. Combinación de las dos informaciones

ALGORITMO - 3. (Rutina BISSECCIO). Combinaremos las dos informaciones anteriores para obtener un algoritmo rápido y robusto.

0. Partimos del intervalo inicial (a_0, b_0) tal que $a_0 < c^* < b_0$. Hacemos $k = 0$.
1. Aplicamos el método de la secante y podemos tener los tres casos siguientes:
 - 1.1. Si encontramos la mayor raíz c de $R_P(c)$ STOP.
 - 1.2. Si encontramos una raíz c_0 de $R_P(c)$ pero no es máxima. Entonces, hacemos $k = k + 1$, tomando como nuevo intervalo $(a_k, b_k) = (c_0 + \epsilon, b_k)$ y volvemos al punto a 1.
 - 1.3. Si el algoritmo no converge o lo hace a un punto externo al intervalo (a_k, b_k) , entonces vamos a 2.
2. Ha fracasado el método de la secante, y aplicamos una etapa del método de la bisección al intervalo (a_k, b_k) .
3. Volvemos a 1.

EJEMPLO REP-5. El intervalo inicial es $(36, 72)$. La rutina BISSECCIO encuentra $c^* = 68$ con 8 iteraciones del método de la secante. Al final tenemos Sturm ($P(x, 68)$).

***** BISSECCIÓ

c	$R_P(c)$
<u>36.0000</u>	-39.4913
<u>72.0000</u>	25.3022
57.9418	-198.6757
70.4119	16.4875
69.4563	10.4699
67.7938	-1.6308
68.0178	.1392
68.0002	.0017
<u>68.0000</u>	<u>.0000</u>

BIS/SEC	iter	8				
OGP/out	sturm					
	1.00	4.00	-11.00	-36.00	<u>68.00</u>	P_0
	.00	4.00	12.00	-22.00	-36.00	P_1
	.00	.00	8.50	21.50	-77.00	P_2
	.00	.00	.00	-9.47	18.95	P_3
	.00	.00	.00	.00	<u>.00</u>	P_4

***** BISSECCIÓ

5.2. Calcular $M(x) := \text{MCD}(P^*, P^{*'})$

Una vez conseguido $P^* = P(x, c^*)$ polinomio bueno, necesitamos $M(x) := \text{MCD}(P^*, P^{*'})$.

Usaremos el método de Euclides. Construimos $\text{Sturm}(P^*) = \{P_0^*, P_1^*, \dots, P_m^*\}$ y tal como se describe en el apéndice, P_m^* coincide con el $\text{MCD}(P^*, P^{*'})$ salvo una constante y entonces podemos tomar $M(x) = P_m^*(x)$ ya que la constante no afecta a las raíces de $M(x)$. Nótese que para calcular $M(x)$ no necesitamos hacer ningún cálculo adicional una vez hemos construido $\text{Sturm}(P^*)$.

En el ejemplo anterior veíamos como al ser $P_4(x)$ el polinomio nulo, entonces $P_3(x) = -9.4x + 18.95 = -9.4 \cdot (x - 2)$ nos da el $\text{MCD}(P^*, P^{*'}) = (x - 2)$.

5.3. Calcular x^* una raíz de $M(x)$ (Rutina BIRGE)

El teorema DES-1 nos asegura que cualquier raíz real de $M(x)$ es un óptimo global. Tenemos que resolver pues la ecuación $M(x) = 0$. Lo haremos con el método de Birge-Vieta descrito en el apéndice. En muchos casos $M(x) = 0$ será una ecuación de primer grado y la rutina BIRGE no hará más que un cálculo directo.

EJEMPLO REP-6. Continúa del ejemplo REP-5.

La rutina BIRGE calcula la única raíz de $M(x) = -9.47x + 18.95$ que es $x = 2$.

En consecuencia ya tenemos el óptimo global de $P(x)$ i.e. $x^* = 2$.

```
***** BIRGE  0
OGP/in   mcd      -9.47  18.95
OGP/out  OPTIM                    2.0000
***** BIRGE  1
```

6. EJEMPLOS

6.1. Ejemplo completo

Ahora reunimos el ejemplo que hemos visto de forma fragmentada para dar una visión global del método.

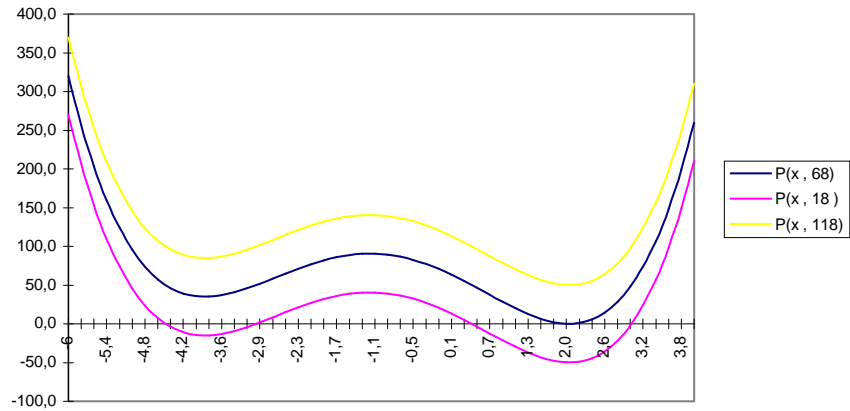
6.1.1. Ideas, gráficos y cálculos

Queremos calcular $\min_x P(x)$ donde $P(x) = x^4 + 4x^3 - 11x^2 - 36x + 18$.

Según el valor del término independiente c , tendremos un polinomio bueno, rico o pobre.

$$P(x, c) := x^4 + 4x^3 - 11x^2 - 36x + c.$$

$P(x, 68)$	polinomio bueno	(1 raíz real).
$P(x, 18)$	polinomio rico	(4 Raíces reales).
$P(x, 118)$	polinomio pobre	(0 Raíces reales).



Para obtener $P^*(x)$ aparte de calcular $\#Raíces(P)$ usamos la función residuo $R_P(c)$ que en este caso, dado que $P(x, c) = x^4 + 4x^3 - 11x^2 - 36x + c$ tiene la siguiente expresión:

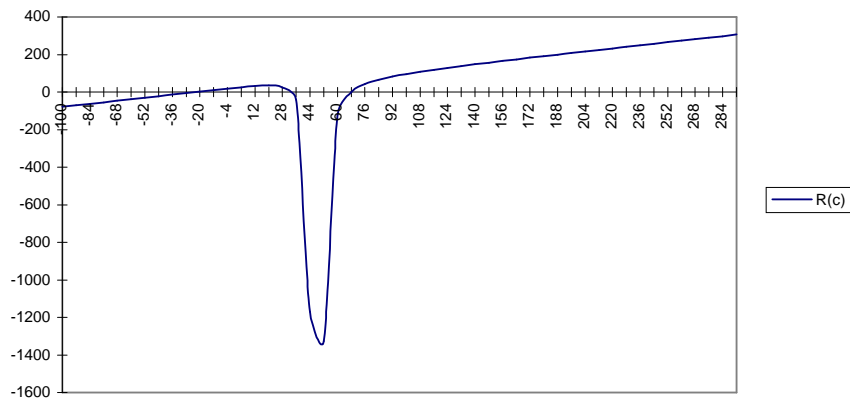
$$R_P(c) = A(c)/B(c)$$

donde

$$A(c) = 289 \cdot (16c^3 - 1256c^2 - 483c + 809676)$$

$$B(c) = (68c - 3255)^2$$

En la siguiente representación de $R_P(c)$ se ve como su mayor raíz real es $c^* = 68$ confirmando el hecho de que $P(x, 68)$ sea un polinomio bueno.



El método del MCD nos conduce a $P^*(x) = x^4 + 4x^3 - 11x^2 - 36x + 68$.

A modo de ejemplo veamos ahora Sturm (P^*) y el cálculo de $\#Raíces(P^*)$:

$$\begin{aligned} P_0(x) &= P^*(x) = x^4 + 4x^3 - 11x^2 - 36x + 68 \\ P_1(x) &= P^{*'}(x) = 4x^3 + 12x^2 - 22x - 36 \\ P_2(x) &= 8.5x^2 - 21.5x - 77 \\ P_3(x) &= -9.4x + 18.8 = -9.4 \cdot (x - 2) \\ P_4(x) &= 0 \end{aligned}$$

$$\begin{aligned} N(-\infty) &:= \text{Cambios de signo en } \{ P_0(-\infty), P_1(-\infty), P_2(-\infty), P_3(-\infty), P_4(-\infty) \} = \\ &= \text{Cambios de signo en } \{ +, -, +, +, 0 \} = \\ &= 2 \end{aligned}$$

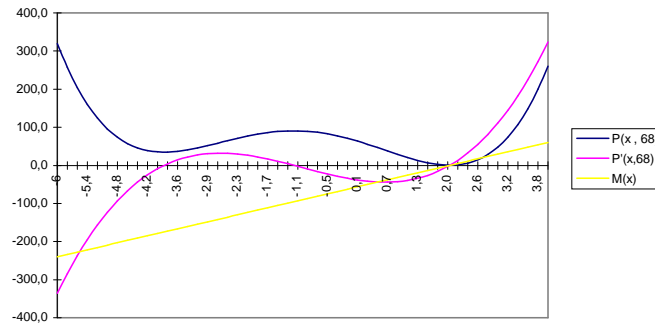
$$\begin{aligned} N(\infty) &:= \text{Cambios de signo en } \{ P_0(\infty), P_1(\infty), P_2(\infty), P_3(\infty), P_4(\infty) \} = \\ &= \text{Cambios de signo en } \{ +, +, +, -, 0 \} = \\ &= 1 \end{aligned}$$

Entonces por el teorema de Sturm tenemos que $\#Raíces(P^*) = N(-\infty) - N(\infty) = 1$ y podemos asegurar que $P^*(x)$ es un polinomio bueno. Al principio de este ejemplo podemos ver la representación de $P^*(x)$ que coincide con $P(x, 68)$ y confirmar que $P^*(x)$ tiene una única raíz real. Nótese que al ser $P_4(x) = 0$, entonces $P_3(x) = -9.4 \cdot (x - 2)$ coincide, excepto por una constante, con $\text{MCD}(P^*, P^{*'}) = (x - 2)$.

Una vez tenemos $P^*(x)$, calculamos los polinomios $P^{*'}(x)$ y $M(x)$ que representamos a continuación:

$$\begin{aligned} P^*(x) &= x^4 + 4x^3 - 11x^2 - 36x + 68 \\ P^{*'}(x) &= 4x^3 + 12x^2 - 22x - 36 \\ M(x) &= \text{MCD}(P^*, P^{*'}) = (x - 2) \end{aligned}$$

En el siguiente gráfico se ve como:



- $x = 2$ es el mínimo global de $P(x, 68)$.
- $x = 2$ es el único punto donde se anulan simultáneamente $P^*(x)$ y $P^{*'}(x)$.
- Por lo tanto $(x - 2)$ es un factor de M (en realidad el único) y así $M(2) = 0$.
- Podemos concluir que $x^* = 2$ es el óptimo global de $P^*(x)$ y por lo tanto de $P(x)$.

NOTA. En lugar de representar $y = M(x)$, hemos representado $y = 30 \cdot M(x)$ pues la representación de $M(x)$ no se distinguiría del eje OX .

6.1.2. Resultados con el programa OGP.EXE

Para correr el programa OCP.EXE necesitamos el fichero de datos OGP.DAT que consta de tres partes:

- Grado del polinomio (entero positivo).
- Precisión exigida (por ejemplo 0.0001).
- El polinomio que queremos minimizar escrito en forma de columna.

EJEMPLO REP-6. Para introducir el polinomio $P(x) = x^4 + 4x^3 - 36x + 18$ crearíamos un fichero OGP.DAT con el siguiente contenido.

4	(Grado del polinomio)
0.0001	(Precisión exigida)
1.1	(a_1)
4.0	(a_2)
0.0	(a_3)
-36.0	(a_4)
18.0	(a_5)

Dado que $P(x, 18)$ es un polinomio rico, después de 2 iteraciones de la rutina «empo-
brint» obtenemos $P(x, 72)$ que es un polinomio pobre. A continuación tenemos Sturm
($P(x, 72)$)

```

***** EMPOBRINT  0
Polinomio a optimizar P(x, 18)
      1.00   4.00  -11.00  -36.00   18.00

OGP/out      sturm
      1.00   4.00  -11.00  -36.00   72.00  P0
      .00   4.00   12.00  -22.00  -36.00  P1
      .00   .00   8.50   1.50  -81.00  P2
      .00   .00   .00  11.36  18.06  P3
      .00   .00   .00   .00  25.30  P4
***** EMPOBRINT  2

```


El intervalo inicial donde se encuentra c^* es (36, 72). La rutina BISSECCIÓ encuentra c^* con 8 iteraciones del método de la secante. Al final tenemos Sturm ($P(x, 68)$):

***** BISSECCIÓ

c_k	$R_P(c_k)$
<u>36.0000</u>	-39.4913
<u>72.0000</u>	25.3022
57.9418	-198.6757
70.4119	16.4875
69.4563	10.4699
67.7938	-1.6308
68.0178	.1392
68.0002	.0017
<u>68.0000</u>	<u>.0000</u>

BIS/SEC	iter	8				
OGP/out	sturm					
	1.00	4.00	-11.00	-36.00	<u>68.00</u>	P_0
	.00	4.00	12.00	-22.00	-36.00	P_1
	.00	.00	8.50	21.50	-77.00	P_2
	.00	.00	.00	-9.47	18.95	P_3
	.00	.00	.00	.00	<u>.00</u>	P_4

***** BISSECCIÓ

La rutina BIRGE calcula la única raíz de $M(x) = -9.47x + 18.95$ que es $x = 2$.

En consecuencia ya tenemos el óptimo global de $P(x)$ i.e. $x^* = 2$.

***** BIRGE 0

OGP/in mcd -9.47 18.95

OGP/out OPTIM 2.0000

***** BIRGE 1

6.2. Pruebas con otros polinomios

A continuación definimos unos polinomios que nos servirán para poner a prueba el método del MCD descrito anteriormente. Dado un polinomio

$$P(x) = a_1 \cdot x^n + a_2 \cdot x^{n-1} + a_3 \cdot x^{n-2} + \dots + a_n \cdot x + a_{n+1}$$

le haremos corresponder el vector formado por sus coeficientes

$$P(x) = (a_1, a_2, \dots, a_{n+1}).$$

La etiqueta del polinomio nos dirá el grado y el número de identificación del polinomio. Por ejemplo con 4POL1, designamos un polinomio de grado 4 y que identificamos con el número 1.

$$\begin{aligned}
 4\text{POL } 1 &= (1. & 0. & 0. & 0. & -1. &) \\
 4\text{POL } 2 &= (1. & 4. & -11. & -36. & 18. &) \\
 4\text{POL } 3 &= (1. & -3. & -1.5 & 10. & 0. &) \\
 4\text{POL } 4 &= (1. & -4. & 4. & 0. & 0. &) \\
 6\text{POL } 1 &= (1. & -6. & 15. & -20. & 15. & -6. & 1. &) \\
 6\text{POL } 2 &= (1. & 0. & -14. & 0. & 49. & 0. & -36. &) \\
 6\text{POL } 3 &= (0.16666 & -2.08 & 0.4875 & 7.1 & -3.95 & -1. & 0.1 &) \\
 8\text{POL } 1 &= (0.001 & 0. & -0.102 & 0.096 & 3.009 & -5.520 & -23.068 & 65.904 \\
 & & & & & & & -40.320 &) \\
 8\text{POL } 2 &= (1. & 0. & 0. & -20.320. & 12.6 & 1. & 0. & \\
 & & & & & & & -13. &) \\
 10 \text{ POL } 2 &= (0.001 & 0.019 & -0.012 & -1.842 & -4.347 & 60.291 & 142.862 & -869.188 \\
 & & & & & & & -864.264 & 5165.28 & -3628.8)
 \end{aligned}$$

En las siguientes tablas se muestra la información más relevante obtenida al minimizar los polinomios anteriores con una tolerancia = 0.0001. Hemos de notar que:

- La rutina Birge hace normalmente una sola iteración al tener que resolver una ecuación de primer grado (caso de un único óptimo global).
- La precisión obtenida es buena con unas pocas iteraciones.

	4POL1	1POL2	4POL3	4POL4	6POL1
Óptimos globales exactos	0,0000	2,0000	-1,0000	0,0000	1,0000
			2,0000		
Óptimo calculado	0,0000	2,0000	-0,9999	0,0000	1,0351
Iteraciones «EMPOBRINT»	1	2	1	1	1
Iteraciones «BISSECCIÓ»					
• Primer intento secante	2	8	5	2	2
• Segundo intento secante					
Iteraciones «BIRGE»	1	1	1	1	1290
Comentarios			(1)		(2)
	6POL2	6POL3	8POL1	8POL2	10POL2
Óptimos globales exactos	-2,6457	10,0000	-7,3400	2,2744	6,4347
	0,0000				
	2,6457				
Óptimo calculado	0,0000	10,0004	-7,3416	2,2400	6,4347
Iteraciones «EMPOBRINT»	1	14	4	6	4
Iteraciones «BISSECCIÓ»					
• Primer intento secante	2	7	14	4	27
• Segundo intento secante			11		
Iteraciones «BIRGE»	1	1	1	1	1
Comentarios	(1)		(3)		

Comentarios:

- (1) Hay polinomios con más de un óptimo global.
- (2) Aquí, excepcionalmente, la rutina BIRGE necesita 1290 iteraciones para alcanzar una aproximación al óptimo. Esto es debido a que 6POL1 es $(x - 1)^6$ polinomio de curvatura prácticamente nula en un entorno amplio del óptimo $x^* = 1$. El método Birge no es pues adecuado para este tipo de casos.
- (3) Aquí el método del MCD con 14 iteraciones ha calculado una primera raíz de $R_P(c)$ que no es la mayor; seguidamente, con 11 iteraciones más ya ha calculado la raíz máxima de $R_P(c)$.

7. CONCLUSIONES

Hemos presentado un nuevo algoritmo para minimizar globalmente en el conjunto de los reales un polinomio de grado par. La característica más importante del algoritmo es el uso de información global de la función a minimizar (número de raíces reales y función residuo) a diferencia de los métodos clásicos basados en información local (derivada en un punto). El enfoque aquí presentado parece más adecuado que métodos anteriores aunque falta hacer una prueba comparativa. Numéricamente el algoritmo se ha mostrado eficiente y robusto.

8. APÉNDICE

8.1. Algoritmo Euclides para calcular el máximo común divisor de dos polinomios $\text{MCD}(P_0, P_1)$

Dividimos P_0 entre P_1 y denotamos el resto $-P_2$. Seguidamente dividimos P_1 entre P_2 y denotamos el residuo $-P_3$. Continuamos este procedimiento hasta que obtengamos un resto igual a cero. Al final del algoritmo podremos establecer las siguientes relaciones:

$$\begin{aligned} P_0 &= Q_1 \cdot P_1 - P_2 \\ P_1 &= Q_2 \cdot P_2 - P_3 \\ &\vdots \\ P_{m-2} &= Q_{m-1} \cdot P_{m-1} - P_m \\ P_{m-1} &= Q_m \cdot P_m \end{aligned}$$

Entonces $P_m = \text{MCD}(P_0, P_1)$

8.2. (División sintética) Generalización de la «Regla de Ruffini» para dividir P_0 entre P_1

Escribimos horizontalmente los coeficientes del dividendo en orden decreciente (como en el método de Ruffini). Seguidamente colocamos verticalmente los coeficientes del divisor cambiados de signo (como en el método de Ruffini) con el coeficiente de máximo grado normalizado (igual a uno). Una referencia más detallada puede encontrarse en Acton (1990) pag. 181.

EJEMPLO APEN-1. Supongamos que queremos dividir P_1 entre P_2 con:

$$\begin{array}{r} P_1(x) = 2x^4 - 4x^3 + x^2 + 3 \\ P_2(x) = x^3 - 2x^2 - x + 1 \end{array}$$

Escribimos en forma de tabla:

P_2	P_1	2	-4	1	0	3	
+2		*	4	0	*	*	
+1		*	*	2	0	*	
-1		*	*	*	-2	0	
Cociente(x)	2	0	3	-2	3	Resto(x)	

$$\text{Cociente}(x) = 2x \qquad 3x^2 - 2x + 3 = \text{Resto}(x)$$

La regla de formación es:

1. Sumamos la columna.
2. Multiplicamos el resultado del paso 1 por cada coeficiente de P_2 , empezando por arriba, colocando el producto en sucesivas columnas hacia la derecha y en la misma fila que el coeficiente de P_2 correspondiente.
3. Volvemos al paso 1, pero omitiendo el paso 2 después de haber calculado el cociente.

Al final, tendremos dos triángulos en blanco que en el ejemplo hemos marcado con «*».

8.3. Método de la secante para el cálculo de raíces reales

Queremos resolver $g(x) = 0$.

Este método es una variante del método de Newton, donde en lugar de usar $g'(x_k)$, usamos una aproximación $(g(x_k) - g(x_{k-1})) / (x_k - x_{k-1})$.

0. Partimos del intervalo inicial (a, b) que contiene la raíz buscada.

1. Inicializamos $x_0 = a$ $x_1 = b$.

2. Mientras $(|x_k - x_{k-1}| > \epsilon)$ hacer:

$$\text{paso} = g(x_k) \cdot \{(x_k - x_{k-1}) / (g(x_k) - g(x_{k-1}))\}$$

$$x_{k+1} = x_k - \text{paso}.$$

Final mientras.

8.4. Método de Birge-Vieta para calcular una raíz de un polinomio

Queremos resolver $P(x) = 0$ siendo $P(x)$ un polinomio.

Este método es una especialización del método de Newton, donde calculamos $P(x_k)$ y $P'(x_k)$ de forma muy eficiente, aprovechando las propiedades algebraicas de los polinomios.

Teorema del residuo. Consideramos $P(x)$ y suponemos que expresamos $P(x) = (x - x_0) \cdot Q(x) + R_1$. Entonces:

- a) $P(x_0)$ coincide con el residuo R_1 del cociente $P(x) : (x - x_0)$
- b) $P'(x_0)$ coincide con el residuo R_2 del cociente $Q(x) : (x - x_0)$

ALGORITMO. (Método Birge-Vieta)

0. Partimos del intervalo inicial (a, b) .

1. Inicializamos $x_0 = a$ $x_1 = b$.

2. Mientras $(|x_k - x_{k-1}| > \epsilon)$ hacer

$$\text{paso} = R_1 / R_2$$

$$x_{k+1} = x_k - \text{paso}.$$

Final mientras.

BIBLIOGRAFÍA

- [1] **Acton, F.S.** (1990). «Numerical Methods that Work», EEUU, *The Mathematical Association of America*, 2ª Ed., 1990.
- [2] **Beltran, C.** (1997). «Globally Minimizing Even Degree Polynomials on the Real Line». *Research Report 97/01 of the Statistics and Operations Research Department-UPC*. Date 2/97.
- [3] **Bromberg, M.** and **Chang, T.** (1992). «One Dimensional Global Optimization Using Linear Lower Bonds» from the book *Recent Advances in Global Optimization*, New Jersey, EEUU, Princeton University Press, 1992, pp. 201-220.
- [4] **Floudas, C.A.** and **Visweswaran, V.** (1992). «Global Optimization of Problems with Polynomial Functions in one Variable» from the book *Recent Advances in Global Optimization*, New Jersey, EEUU, Princeton University Press, 1992, pp. 164-197.
- [5] **Goldstein, A.A.** and **Price, J.F.** (1971). «On descent from local minima», *Mathematics of Computation*, **25**, 569-574.
- [6] **Henrici, P.** (1974). «Applied and Computational Complex Analysis». Volume I, EEUU, John Wiley & Sons, 1974.

- [7] **Wingo, D.R.** (1985). «Globaly Minimizing Polinomials Without Evaluating Derivatives», *International Journal of Computer Mathematics*, **17**, 287.
- [8] **Wolfram, S.** (1992). «Mathematica: A System for Doing Mathematics by Computer», EEUU, Addison-Wesley Publishing Company, 2^a Ed.

ENGLISH SUMMARY

GLOBALLY MINIMIZING A POLYNOMIAL ON THE REAL LINE

C. BELTRÁN ROYO

Universitat Politècnica de Catalunya*

An algorithm for globally minimizing an even degree polynomial on the real line is proposed and tested. The algorithm is based on the idea of translating the polynomial graph vertically, until the OX axis is tangent to the graph of the translated polynomial. At this point, any root of the translated polynomial is a global minimizer of the original polynomial.

Keywords: Global optimization, greatest common divisor of two polynomials, Euclides algorithm, Sturm sequence, synthetic division of two polynomials.

AMS Classification (MSC 2000): 49J05

*Dept. d'Estadística i Investigació Operativa. Secció d'Informàtica. Universitat Politècnica de Catalunya. Pau Gargallo, 5. 08028 Barcelona (Espanya).

–Received October 1997.

–Accepted July 1998.

The objective of this work is to minimize an even degree polynomial globally i.e. we want to solve the problem:

$$\min\{P(x) : x \in \mathbb{R}\}$$

Traditionally this problem has been solved using local information such as the first derivative. Our approach will use global information (the total number of different real zeros of $P(x)$ in \mathbb{R}). This approach avoids an exhaustive search into an interval (a, b) carried out by local information based algorithms.

The algorithm is based on the idea of translating the polynomial graph vertically until the OX axis is tangent to the graph of the translated polynomial. At this point, any root of the translated polynomial is a global minimizer of the original polynomial.

First of all we define a *rich* polynomial as a polynomial that crosses the OX axis, a *good* polynomial as a polynomial that is tangent to the OX axis and a *poor* polynomial as a polynomial that does not intersect the OX axis.

The main results we have proved are:

- a) If $P^*(x)$ is a good polynomial, $P^{*'}(x)$ its derivative and $\text{GCD}(P^*, P^{*'})$ their greatest common divisor, then, the set of global minimizers of $P^*(x)$ is the set of real zeros of the $\text{GCD}(P^*, P^{*'})$.
- b) Given the polynomial $P(x, c) = a_1 \cdot x^n + a_2 \cdot x^{n-1} + \dots + a_n \cdot x + c$ we define an associated función $R_P(c)$ called the residual function. Then $P(x, c^*)$ is a good polynomial if and only if c^* is the greatest real zero of the residual function $R_P(c)$.

The GCD method for finding a global minimizer of $P(x)$ is:

- 1) $P(x) = P(x, a_{n+1})$ is translated vertically until it becomes a good polynomial. We use two sources of information: first, Sturm theorem tells us the total number of different real zeros of $P(x, c)$ in \mathbb{R} and second, solving the equation $R_P(c) = 0$ we find c^* the exact translation that converts $P(x, a_{n+1})$ into a good polynomial $P^*(x) = P(x, c^*)$.
- 2) Solving the equation $M(x) = 0$ where $M(x)$ is the $\text{GCD}(P, P^*)$, we will have a global minimizer of $P(x)$.

Computational experience has shown that the GCD method is an efficient and reliable method for globally minimizing a polynomial.