Revista Tecnocientífica URU
Universidad Rafael Urdaneta
Facultad de Ingeniería
Nº 3 Julio - Diciembre 2012
ISSN: 2244 - 775X

Estudio comparativo entre el método de Lemke y el método de los conjuntos activos para programación cuadrática

Marihebert Leal[1], Kilkenis Fuenmayor[1], Javier Bastidas[2] y Susana Salinas[1,3]

[1]Centro de Investigación de Matemática Aplicada. Facultad de Ingeniería.
Universidad del Zulia. Maracaibo, Estado Zulia.

[2]Laboratorio de Control e Instrumentación. Escuela de Ingeniería Mecánica. Facultad de Ingeniería.
Universidad del Zulia. Maracaibo, Estado Zulia.

[3]Departamento de Ciencias Básicas. Facultad de Ingeniería. Universidad Rafael Urdaneta.
Maracaibo, Estado Zulia.

Recibido: 01-07-2012 Aceptado: 16-11-2012

Resumen

En este trabajo se presentan dos métodos para resolver el problema general de programación cuadrática convexa con restricciones de igualdad y desigualdad. Uno es el método de los conjuntos activos y el otro es el de Lemke. Se realizaron varias pruebas variando el tamaño del problema y el número de restricciones de igualdad y desigualdad, mostrando que el método de los conjuntos activos es eficiente cuando el problema tiene un alto número de restricciones de igualdad, pero resulta lento cuando existe un alto número de restricciones de desigualdad. Mientras que el algoritmo de Lemke resultó eficiente en problemas cuadráticos convexos con un alto número de restricciones de desigualdad.

Palabras clave: Programación cuadrática, métodos de pivoteo, Lemke

Comparative study between Lemke's method and the active set method for quadratic programming

Abstract

In this paper, we present two different methods in order to solve a general convex quadratic programming problem with equality and inequality constraints: a pivot method (Lemke's algorithm) and the most used method, the strategies of active set. Active set method is efficient when the problem has many equality constraints, but it is inefficient when the problem has many inequalities constraints. In the search for a strategy that offers a best time of response for this kind of problems, we studied into the pivots methods, Lemke's algorithm. We provide numerical simulations and comparative results that show the Lemke's method is effective on problems with a large number of inequality constraints.

Key words: Quadratic programming, pivoting techniques, Lemke's methods.

Introducción

En este trabajo se considera el problema general de programación cuadrática. El interés en este tipo de problemas surge en vista que los modelos cuadráticos son ampliamente utilizados en aplicaciones de la vida real y particularmente en el uso del SQP (*Sequential Quadratic Programming*), [1], para la solución de un problema de programación no lineal, la dirección de búsqueda en cada iteración es determinada como la solución de un Problema de Programación Cuadrática (PPC).

Además, en el área de control de procesos, específicamente en problemas de control óptimo y modelos de control predictivo [2],[3], se hace necesaria la solución de un PPC con un alto número de restricciones de desigualdad en un tiempo aceptable.

Entre las estrategias para la solución de problemas de este tipo [4], se tienen los métodos de punto interior primal-dual [5], los métodos de pivoteo [6] y uno de los métodos más utilizados como lo es el método de los conjuntos activos [7].

Tanto los métodos de punto interior primal-dual como los métodos de pivoteo resuelven el sistema de ecuaciones que se obtiene al aplicarle al PPC las condiciones de Karush-Kunh-Tucker [8], es decir, resuelven un problema complementario lineal (PCL).

El método de los conjuntos activos [9], tiene como ventaja ofrecer un buen tiempo de desempeño cuando el PPC tiene un alto número de restricciones de igualdad, pero a medida que las restricciones de desigualdad aumentan el tiempo de ejecución aumenta.

Entre los métodos de pivoteo se seleccionó el algoritmo de Lemke [6], el cual permite resolver problemas de programación cuadrática convexa (PPCC), con sólo restricciones de desigualdad y las variables deben ser positivas. Esta característica permite que el método sea especialmente efectivo para resolver PPC con un alto número de restricciones de desigualdad. Para el caso de un PPCC con restricciones tanto de igualdad como de desigualdad y que las variables no tengan restricción de signo, el problema cuadrático necesita algunas modificaciones (ver sección 2) para que el algoritmo de Lemke pueda resolverlo.

Este trabajo presenta simulaciones numéricas comparativas para mostrar la importancia de la estabilidad y eficiencia del método de Lemke en la solución de un PPC de gran tamaño con un alto número de restricciones de desigualdad con respecto al método de los conjuntos activos.

El trabajo está estructurado de la siguiente forma: En la sección siguiente se hace una breve descripción de las modificaciones del PPCC para poder ser resuelto por el algoritmo de Lemke. En la sección 3 presentamos el Método de Lemke, un método conocido en la literatura para resolver PCL utilizando las condiciones de optimalidad de Kunh Tucker. La sección 4 está dedicada a simulaciones numéricas de gran tamaño y comparaciones de las predicciones numéricas entre los dos métodos.

Transformación de un problema cuadrático convexo en un problema complementario lineal

El problema cuadrático general se puede expresar como

$$\text{Min}, \frac{1}{2} x^t Hx + x^t c, \text{ sujeto a } A_i x = b_i, \ i \in E, \ A_i x \leq b_i, \ i \in I. \tag{1}$$

E e I son conjuntos de índices para restricciones de igualdad y desigualdad respectivamente. La matriz H es simétrica y semidefinida positiva (cuando no es definida positiva).

Este problema se resuelve por distintas estrategias, una de las más utilizadas es el método de los conjuntos activos.

El algoritmo de Lemke sólo resuelve PPCC, ecuación (2), donde solo existan restricciones de desigualdad y un punto solución estrictamente positivo

$$\text{Min, } \frac{1}{2}\, x^t Hx + x^t c, \text{ sujeto a } Ax \leq b\ , x \geq 0. \tag{2}$$

donde H es simétrica y definida positiva.

Las condiciones de Kunh-Tucker para el problema (2) son

$$\begin{aligned} &v = Hx + c^t + A^t u \geq 0 \\ &y = -(Ax - b) \geq 0 \\ &x^t v = 0 \\ &u^t y = 0 \\ &x, u, v, y \geq 0. \end{aligned} \tag{3}$$

En forma matricial

$$\begin{aligned} &\begin{bmatrix} v \\ y \end{bmatrix} = \begin{bmatrix} c \\ b \end{bmatrix}^t + \begin{bmatrix} H & A^t \\ -A & 0 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \\ &\begin{bmatrix} x \\ u \end{bmatrix}^t \begin{bmatrix} v \\ y \end{bmatrix} = 0 \\ &x, u, v, y \geq 0. \end{aligned} \tag{4}$$

De esta forma se reduce el PPC al siguiente PCL:

$$\begin{aligned} &w = q + Mz \\ &z^t w = 0 \\ &w, z \geq 0, \end{aligned} \tag{5}$$

donde *w, q, M* y *z* son:

$$w = \begin{bmatrix} v \\ y \end{bmatrix}\ q = \begin{bmatrix} c \\ b \end{bmatrix}\ M = \begin{bmatrix} H & A^t \\ -A & 0 \end{bmatrix}\ z = \begin{bmatrix} x \\ u \end{bmatrix} \tag{6}$$

Este problema se puede resolver utilizando el algoritmo de pivoteo propuesto por Lemke, el cual resuelve el problema anterior si la matriz M cumple con ciertas propiedades [6].

Para utilizar este algoritmo cuando en el PPC existen restricciones de igualdad y el punto solución es irrestricto en signo este requiere algunas modificaciones.

Transformación con restricciones de igualdad

Consideremos el siguiente problema cuadrático con solo restricciones de igualdad.

$$\text{Min, } \frac{1}{2}\, x^t Hx + x^t c, \text{ sujeto a } Ax = b\ \ x \geq 0 \tag{7}$$

donde (7) se puede reescribir de la siguiente forma:

$$\text{Min, } \frac{1}{2}\, x^t Hx + x^t c, \text{ sujeto a } Ax \leq b\ , -Ax \leq -b\ , x \geq 0 \tag{8}$$

Para este problema las condiciones de Kunh-Tucker son las siguientes

$$\begin{aligned} &v = Hx + c^t + \begin{bmatrix} A \\ -A \end{bmatrix}^t u = 0 \\ &y = -\left(\begin{bmatrix} A \\ -A \end{bmatrix} x - \begin{bmatrix} b \\ -b \end{bmatrix} \right) = 0 \\ &x^t v = 0 \\ &u^t y = 0 \\ &x, u \geq 0. \end{aligned} \tag{9}$$

Entonces, en forma matricial:

$$\begin{aligned} &\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} c \\ b \\ -b \end{bmatrix}^t + \begin{bmatrix} H & A^t & -A^t \\ -A & 0 & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ u_+ \\ u_- \end{bmatrix} \\ &\begin{bmatrix} x \\ u_+ \\ u_- \end{bmatrix}^t \geq 0, \end{aligned} \tag{10}$$

donde (10) se reduce al siguiente problema

$$\begin{aligned} &0 = q + Mz \\ &z^t w = 0 \\ &w = 0 \\ &z \geq 0, \\ &z = -M^{-1} q \end{aligned} \tag{11}$$

Siendo *M:*

$M = \begin{bmatrix} H & A^t & -A^t \\ -A & 0 & 0 \\ A & 0 & 0 \end{bmatrix}$, la cual es singular por tener filas linealmente dependientes, por tanto este PCL no tiene solución. Por ello, es necesario introducir un valor constante ε, que es una pequeña fracción apropiada de *abs(b)* como se muestra a continuación para que el problema tenga solución.

$$v = Hx + c^t + \begin{bmatrix} A \\ -A \end{bmatrix}^t u = 0$$

$$y = -\left(\begin{bmatrix} A \\ -A \end{bmatrix} x - \begin{bmatrix} b+\varepsilon \\ -b+\varepsilon \end{bmatrix} \right) = 0 \tag{12}$$

$$x^t v = 0$$

$$u^t y = 0$$

$$x, u \geq 0.$$

Entonces en forma matricial

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} c \\ b+\varepsilon \\ -b+\varepsilon \end{bmatrix}^t + \begin{bmatrix} H & A^t & -A^t \\ -A & 0 & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ u_+ \\ u_- \end{bmatrix}$$

$$\begin{bmatrix} x \\ u_+ \\ u_- \end{bmatrix}^t \geq 0. \tag{13}$$

De esta forma el problema siempre tiene solución única.

Transformación irrestricto en signo

Considerando el problema cuadrático (14) donde las variables no tienen restricción en signo,

Min, $\frac{1}{2}\, x^T Hx + x^T C$. Sujeto a $Ax \leq b$, x irrestricto. (14)

Este tipo de problema se presenta cuando se utiliza la estrategia del SQP.

Para este caso en que algunas variables no son restrictas en signo, se debe reformular el PPC, para que al convertirlo en un PCL, se cumplan las condiciones de complementariedad.

$$z_i^t\, w_i = 0,\ w_i, z_i \geq 0 \tag{15}$$

Por tanto, para resolver el siguiente PPC, donde H (definida positiva y simétrica) y A son matrices y C, b, y x son vectores, es necesario realizar el siguiente cambio de variables [10].

$$x = [w - pe],\ e = [1, 1 \ldots 1]^t,\ w \geq 0,\ p \geq 0 \tag{16}$$

Al sustituir (16) en (14) se obtiene el siguiente problema:

$$\text{Mín } c^T [w - pe] + \tfrac{1}{2}[w - pe] H [w - pe]^t \tag{17}$$

$$\text{s.a. } A[w - pe] \leq b, w \geq 0, p \geq 0$$

Operando algebraicamente se obtiene el siguiente sistema matricial:

$$\tfrac{1}{2}\begin{bmatrix} w & p \end{bmatrix}\begin{bmatrix} H_{11} & H_{12} & H_{1n} & -\sum_{i=1}^{m} H_{i1} \\ H_{21} & H_{22} & H_{2n} & -\sum_{i=1}^{m} H_{i2} \\ H_{m1} & H_{m2} & H_{mn} & -\sum_{i=1}^{m} H_{in} \\ -\sum_{i=1}^{m} H_{i1} & -\sum_{i=1}^{m} H_{i2} & -\sum_{i=1}^{m} H_{in} & \sum_{i=1,j=1}^{m,n} H_{ij} \end{bmatrix}\begin{bmatrix} w \\ p \end{bmatrix} + c^T \begin{bmatrix} w \\ p \end{bmatrix} \tag{18}$$

$$\text{s.a.} \begin{bmatrix} A_{11} & A_{12} & A_{1n} & -\sum_{i=1}^{n} A_{1i} \\ A_{21} & A_{22} & A_{2n} & -\sum_{i=1}^{n} A_{2i} \\ Am_1 & A_{22} & A_{2n} & -\sum_{i=1}^{n} A_{mi} \end{bmatrix}\begin{bmatrix} w \\ p \end{bmatrix} \leq \begin{bmatrix} b_1 \\ b_2 \\ b_m \end{bmatrix}$$

$$w \geq 0, p \geq 0$$

Donde w es un vector que depende del tamaño de x, p es un coeficiente y la nueva matriz H es semidefinida positiva donde el sistema resultante puede no tener solución o tener infinitas soluciones, dependiendo de las restricciones.

Algoritmo de Lemke

Muchos algoritmos, basados en estrategias de pivoteo, han sido propuestos para resolver el PCL [11], [12]. En esta sección se ofrece una breve explicación de la estrategia utilizada en el algoritmo de Lemke, que establece condiciones en la matriz M para garantizar la existencia de una solución [6] y la discusión de algunas herramientas para mejorar su estabilidad numérica.

Generalmente la solución de un PCL es expresado como la búsqueda de encontrar w y z tal que, satisfaga las condiciones del problema (5). Es obvio que si $q \geq 0$ la solución del problema es el conjunto básico $w=q$ y la matriz básica $B=I$. Mas general si q tiene componentes negativos, Lemke ofrece una estrategia para este caso extendiendo el problema (5), agregando una variable extra z_0 y un nuevo vector no negativo e, el cual es positivo en cada componente correspondiente a un negativo q_i.

$$Iw - Mz - ez_0 = q \tag{19}$$

$$z^t w = 0$$

$$w \geq 0, z \geq 0, z_0 \geq 0$$

Claramente cualquier solución de (19) con $z_0 = 0$ es también solución del problema original (5). Establecer $z = 0$ es también claro que $w \geq 0$ para todos los valores positivos suficientemente grandes de z_0. El algoritmo de Lemke comienza buscando el valor más pequeño de z_0 tal que $w \geq 0$, para el cual por supuesto por lo menos un elemento es exactamente cero.

Haciendo $z_0 = q_r = \max\{- q_i : \leq i \leq n\}$, $z = 0$ y $w = q + ez_0$, se realiza el pivote de la fila r y la columna correspondiente a la variable artificial z_0, se obtiene un nuevo conjunto de variables básicas intercambiando w_r por z_0 y una nueva matriz básica. Esta solución es llamada solución casi complementaria básica factible. Con esta solución inicial, a través de una secuencia de pivotes que son especificados, satisfaciendo (19) (Soluciones factibles), se intenta llevar la variable artificial a cero para mantener las complementariedades.

Sea B la actual base casi complementaria factible de (19) y suponga que (w_p, z_p) es el par de variables complementarias de las cuales una de ellas tiene que convertirse en no básica (en el primer paso considerando éste el que corresponde a w_r). Los pasos para las siguientes iteraciones son extremadamente simples:

1. Si z_0 se convierte en no-básica o $z_0 < \delta$, para algún valor pequeño δ, terminar.

2. Determinar cuál de las w_p, o z_p va a entrar a la base (el complemento de la variable que deja la base) y determinar α, la actualización de ésta está asociada al vector columna original $B\alpha = u_p$ (donde u_p es la p-ésima columna de I) si w_p va a entrar a la base o $B\alpha = m_p$ (donde m_p es la p-ésima columna de M) si z_p va a entrar a la base.

3. Actualizar el lado derecho de (19) resolviendo la ecuación $B\beta = q$.

4. Calcular $\dfrac{\beta_r}{\alpha_r} = \min\limits_{\sigma_i > 0} \dfrac{\beta_i}{\alpha_i}$. El nuevo par de variables no básicas es ahora (w_p, z_p) y el nuevo par de variables básicas que dejan la base son (w_p, z_p).

5. Dependiendo del método utilizado para resolver el sistema de ecuaciones en los pasos 2 y 3 se hace necesario actualizar la forma del producto estándar o la descomposición de B.

Se puede mostrar [13].que el algoritmo de pivoteo complementario de Lemke siempre se detiene en un número finito de pasos con una solución complementaria básica factible o una terminación rayo y si la matriz M asociada al problema complementario lineal es copositiva-plus, luego el algoritmo produce una solución complementaria básica factible. Este resultado también garantiza la convergencia del método de Lemke a una solución del problema cuadrático asociado, desde la matriz M formada usando una aproximación definida positiva del Hessiano de la matriz, que es siempre copositiva-plus.

Desde el punto de vista computacional existen algunas consideraciones importantes que deben tomarse en cuenta para mejorar la estabilidad del algoritmo. Para comenzar, tenemos que en cada iteración solo dos vectores α y β del sistema transformado son usados, por lo que se puede buscar economizar en cálculos generando estos vectores de los datos originales, en lugar de transforman todo el sistema en cada iteración.

Considerando ahora la solución $B\alpha = u_p$, $B\beta = q$, éstos pueden ser calculados usando también B^{-1} explícitamente o como una secuencia de matrices pivotes. La mayor desventaja con esta aproximación es que los errores de redondeo son acumulados muy rápido. Para aliviar este problema, mucha de la preocupación por la exactitud se puede eliminar resolviendo las ecuaciones con un método de descomposición de matrices como la técnica de descomposición LU y utilizando técnicas estables para actualizar los factores triangulares de la base. En la implementación del algoritmo de Sauders [14] una implementación eficiente del método de Batels-Golub fue utilizada.

Otro aspecto que es necesario considerar cuando se implementa el método de Lemke es el hecho que a diferencia del método simplex, la variable que entra a la base es determinada únicamente por la iteración anterior. Esto es que en general no hay posibilidad de rechazar la columna a causa de un pivote no satisfactorio utilizando alguna otra columna. Por tanto, se debe ser muy cuidadoso en lo posible al elegir el pivote en el paso 3, en cada iteración. La primera y la más obvia precaución es utilizar la técnica de la selección de la fila pivote de Harris [15] para seleccionar el pivote más grande disponible, esto es elegir los pivotes que mantengan factibilidad dentro de una tolerancia dada ε_0. En pocas palabras esto involucra dos etapas del proceso:

1. $\theta = \min\limits_{\alpha_i > 0} \dfrac{\beta_r + \varepsilon_0}{\alpha_i}$; 2. Elegir $\dfrac{\beta_r}{\alpha_r}$, tal que $\alpha_r = \max\left\{ \alpha_i / \alpha_i \geq \varepsilon_p, \dfrac{\beta_i}{\alpha_i} \leq \theta \right\}$

Donde ε_p es una tolerancia de pivote absoluto.

Como un chequeo adicional se puede imponer una prueba de tolerancia de pivote relativo sobre α_r. Sea $\alpha_{\max} = \max|\alpha_r|$ y ε_r la tolerancia. Luego si $\alpha_r \geq \varepsilon_r \varepsilon_{\max}$ el pivote es aceptado. Sino la elección es suspendida y la base es reinvertida, entonces α será recalculado tan exacto como sea posible (o práctico), si el recálculo de α_r falla de nuevo la prueba de pivote relativo, éste debe ser aceptado, pero con una mayor confianza. Claro, si α_r es muy pequeño la prueba será ineficiente y si es muy grande, la reinversión se realizará muchas veces más de lo necesario. Tomlin recomienda un valor de 10^{-8} [13].

Como una medida final Tomlin [13] recomienda chequear periódicamente el error relativo en β llevando a cabo una iteración de refinamiento iterativo con la actual representación de B^{-1}. Si el error relativo no es satisfactorio se llama a una reinversión.

Resultados numéricos

Para estudiar el comportamiento de los métodos de Lemke y de los Conjuntos Activos se generaron problemas cuadráticos convexos (*H* definida positiva) aleatorios, para asegurar la convergencia del método de Lemke [16]. Se desarrolló un alto número de pruebas de diferentes tipos de problemas, donde se varió el número de variables así como el número de las restricciones de desigualdad, activas e igualdad. A continuación se presentan algunos de los resultados obtenidos, un conjunto más completo de los resultados se muestra en [17]. En todas las tablas la primera columna representa el número de variables, la segunda el tiempo de ejecución en seg. (CPU) para el método de los conjuntos activos, la tercera el tiempo de ejecución del método de Lemke y la última la relación del tiempo de ejecución de Lemke con respecto al método de los conjuntos activos (CA) , es decir, $\frac{t_{LEMKE}}{t_{CA}}$.

Para las tablas 1 y 2, se tienen 10 y 50 restricciones activas respectivamente, se puede observar que a medida que aumenta el número de variables el tiempo de ejecución del algoritmo de Lemke es menor con respecto al algoritmo de los conjuntos activos.

Tabla 1. Tiempos de comparación de los dos métodos

N	Tq	Tl	tl/tq
100	1,4331	2,3087	1,61
200	4,9779	4,6497	0,93
300	16,0095	7,9473	0,49
400	31,9265	13,6128	0,42
500	59,0302	21,1908	0,35
600	100,9205	29,8442	0,29

Figura 1. Desempeño para el problema de la tabla 1

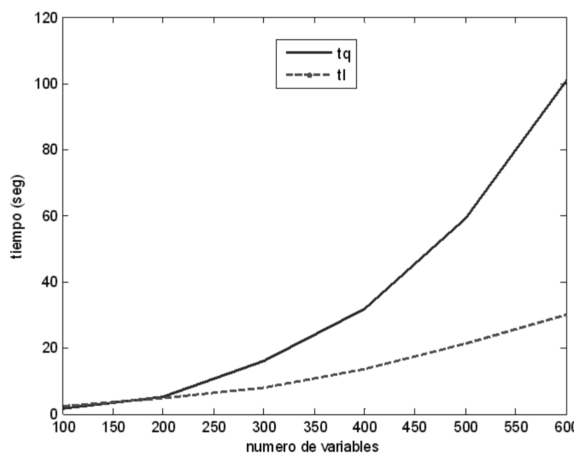
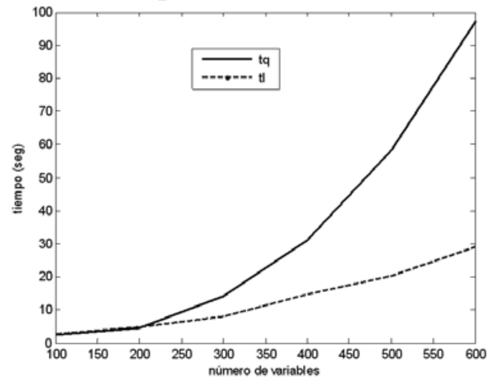


Tabla 2. Tiempos de comparación de los dos métodos

N	tq	tl	tl/tq
100	2,5157	2,6495	1,05
200	4,6168	4,7254	1,02
300	14,1521	8,0157	0,57
400	31,265	14,7722	0,47
500	58,447	20,4268	0,35
600	97,254	29,0439	0,30

Figura 2. Desempeño para el problema de la tabla 2



En la tabla 3 se puede observar que al agregar restricciones de igualdad siendo el número de restricciones de desigualdad mayor, el tiempo de desempeño del algoritmo de Lemke con respecto al de los conjuntos activos es mejor, mientras que en la tabla 4 se observa que al ser mayor el número de restricciones de igualdad el algoritmo de los conjuntos activos ofrece mejores tiempos de respuesta. Además, en la tabla 3 se observa que el método de Lemke presenta aproximadamente el mismo tiempo de solución sin importar el tipo de restricciones, mientras que para el algoritmo de los conjuntos activos, su tiempo de ejecución varía según el tipo de restricción.

Tabla 3. Tiempos de solución para QP de 200 variables con 10 restricciones de igualdad y 90 desigualdad y variando el número de restricciones activas

mde	tq	tl	tl/tq
10	1,8859	0,4967	0,26
20	1,5273	0,5013	0,32
30	1,4237	0,4923	0,34
40	1,2816	0,5084	0,39

Figura 3. Desempeño para el problema de la tabla 3

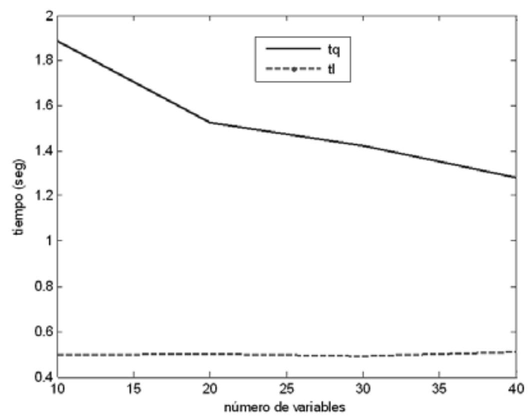
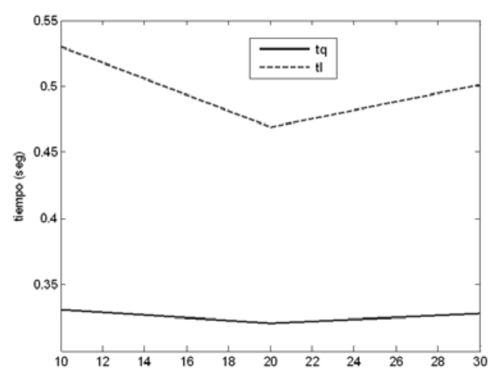


Tabla 4. Tiempos de solución para QP de 200 variables con 70 restricciones de igualdad y 30 desigualdad y variando el numero de restricciones activas

mde	tq	tl	tl/tq
10	0,3308	0,5296	1,60
20	0,3205	0,4689	1,46
30	0,3283	0,5011	1,52

Figura 4. Desempeño para el problema de la tabla 5



Conclusiones

Se pudo comprobar que las modificaciones realizadas al método de Lemke permiten su utilización en problemas de programación cuadrática con restricciones de igualdad, desigualdad e irrestricto en signo. Con el estudio comparativo realizado se pudo determinar que un método alternativo al de los conjuntos activos para la solución de problemas cuadráticos con un alto número de restricciones de desigualdad y algunas de ellas activas es el método de Lemke, ya que ofrece un buen desempeño.

Referencias bibliográficas

1. Boggs P. and Tolle J. Sequential Quadratic programming, Acta numérica, (1996), p. 1-57.
2. Wright S. Applying new optimization algorithms to model predictive control, Chemical process control-V, CACHE, AIChE symposium series, No 316, Volume 93, (1997), p. 147-155.
3. Gopal V. and Biegler L. Large scale Inequality Constrained Optimization and Control, IEEE Control Systems Magazine,(1998), p. 59-68.
4. Freund R. (Massachusetts Institute of technology), Solution Methods for quadratic Optimization, Massachusetts (2004).
5. Domínguez J. and González-Lima M. A primal Dual Interior Point algorithm for quadratic programming, Numerical Algorithms, Vol. 42, (2006), p. 1-30.
6. Bazaraa M., Sherali H. and Shetty, C. (John Wiley and Sons), Nonlinear Programming, Theory and Algorithms, New York (1993).
7. Nocedal J. and Wright S. (Springer Verlag), Numerical Optimization, New York, (1999).
8. Yassine A. Comparative study between Lemke's methods and the interior point method for the monotone linear complementary problem, Studia Univ. Babes-Bolyai, Mathematica, vol. 53,No. 3, September, (2008), p. 119-132.
9. Luenberger D. (Addison-Wesley Iberoamericana), Programación Lineal y No Lineal, Wilmington, Delaware, (1989).
10. Schaester M. Technical note: Unrestricted Variables in Linear Programming, Journal of Optimization Theory and Applications, Vol. 69, Nº 3, (1991), p. 605-610.
11. Judice J. and Mitra G. Reformulation of Mathematical Programming Problems as Linear Complementary Problems and Investigation of their Solutions methods, Journal of Optimization Theory and Applications, Vol. 57, (1988), p. 123-149.
12. Ferris M. Iterative Linear Programming Solution of Convex Programs, Journal of Optimization Theory and Applications, Vol. 65, N°1 April (1990), p. 53-65.
13. Tomlin J. Robust implementation of Lemke's methods for the linear complementary problem, Mat. Prog. Study, No. 7, (1978), p. 55-60.
14. Saunders M. (Academic Press) A fast, stable implementation of the simplex methods using Bartels-Golub updating, Sparse matrix computations, New York, (1976).
15. Harris P. Pivot selection methods of the Devex I. P. Code, Mathematical programming Study, (1975), p. 30-57.
16. Schittokowski K., NLP Notes: Information, Tests, Performance. Lecture Notes in economics and Mathematical Systems, Springer Verlag, NY (1980).
17. Leal M. and Vinante C. "Implementación del método de Lemke para resolución de Problemas Cuadráticos". Reporte Técnico. Laboratorio de Controles e Instrumentación. Julio (2008).