

UNA INTRODUCCIÓN AL MÉTODO DEL ESCALADO
AFÍN PARA PROGRAMACIÓN LINEAL

Jordi Castro
Statistics & Operations Research Dept.
UPC

DATE 2/2000
DR 2000/02

Una introducción al método del escalado afín para programación lineal

Jordi Castro

Dept. d'Estadística i Investigació Operativa
Universitat Politècnica de Catalunya
Pau Gargallo 5, 08028 Barcelona

Abstract: *El método del escalado afín fue el primero de los algoritmos denominados de punto interior para la solución de problemas de programación lineal. A pesar de que puede ser introducido mediante sencillos razonamientos geométricos, los aspectos más teóricos de dicho algoritmo (como la demostración de su convergencia) suelen ser más sutiles que los de métodos alternativos. Este hecho ha motivado que la mayoría de textos bien presenten el método de forma muy superficial, bien reduzcan su exposición a la demostración de una lista de teoremas y proposiciones. En este trabajo, de carácter principalmente docente, se ha pretendido compensar ambas carencias, de forma que, manteniendo el rigor, la presentación sea adecuada tanto para estudios de Ingeniería como de Matemáticas. Se incluye un apartado con una serie de problemas resueltos sobre el tema, entre los que aparece una implementación del algoritmo en el lenguaje Matlab. Esta implementación se utiliza para solucionar un problema de ingeniería de pocas variables, el cual se propone como ejercicio práctico a realizar. También se presenta una comparativa computacional entre una implementación eficiente del método del escalado afín y una del algoritmo del símplex en la solución de un total de 90 problemas estándar de programación lineal. Los resultados obtenidos ratifican la eficiencia del algoritmo del escalado afín.*

An introduction to the affine scaling algorithm for linear programming.

Keywords: Algoritmo de escalado afín, Métodos de punto interior, Programación lineal.

Clasificación AMS: 90C05, 90C06.

1. Introducción

Los métodos de punto interior han sido utilizados durante los últimos años con gran éxito para solucionar problemas de programación lineal. Este éxito ha provocado que actualmente se consideren una alternativa más efectiva que el método del símplex para una gran variedad de problemas lineales. En este documento vamos a desarrollar uno de estos métodos: el del escalado afín.

Este método fue el primero de los denominados de punto interior, y data del 1967 (Dikin (1967)), aunque no fue conocido en Occidente hasta dos décadas más tarde. En la actualidad existen ya diversos textos que describen dicho algoritmo. La mayoría de ellos, sin embargo, no realizan una presentación que balancee los aspectos teóricos con los más intuitivos y geométricos del algoritmo. Así, por ejemplo, mientras las obras de Arbel (1993) y Vanderbei (1996) realizan una presentación de carácter muy introductorio, en Tsuchiya (1996) la exposición se reduce principalmente a un conjunto de teoremas y proposiciones. Algunos textos, como Bertsimas y Tsitsiklis (1997), sí realizan el esfuerzo de presentar ambos aspectos, aunque, en este caso, la presentación se centra en la versión del algoritmo del escalado afín denominada de paso corto, la cual ha mostrado ser la menos eficiente. A diferencia de estos textos, en este documento nos hemos centrado en la variante más eficiente —de paso largo—, presentando

una demostración de convergencia de la misma, sin olvidar los aspectos más geométricos del algoritmo.

Iniciaremos la exposición presentando la notación que se utilizará (sección 2) e introduciendo el problema lineal que pretendemos solucionar así como los requisitos necesarios para el seguimiento de la exposición (sección 3). En la sección 4 se dará una visión global de las diferentes técnicas de punto interior para programación lineal. Acto seguido pasaremos a exponer en la sección 5 el método del escalado afín. Se realiza una presentación constructiva del método, introduciendo en cada apartado sus diferentes etapas algorítmicas. Se incluyen diversos ejemplos para ilustrar cada una de estas etapas. La sección 5 finaliza presentando los principales resultados sobre la convergencia del algoritmo, y una comparativa de su rendimiento computacional respecto el algoritmo del símplex. En la sección 6 se listan una serie de ejercicios propuestos, con sus respectivas soluciones. Durante la presentación del método se hacen diversas referencias a estos ejercicios. En la sección 7 se sugiere un trabajo práctico donde debe solucionarse un problema de optimización real de pequeña dimensión, utilizando una implementación del algoritmo del escalado afín en el lenguaje Matlab. Se adjunta la solución de dicho trabajo práctico. Las secciones 8 y 9 listan determinados aspectos del algoritmo no tratados en el documento y la bibliografía utilizada, respectivamente.

2. Notación

Se seguirá el siguiente convenio a lo largo del documento: Los términos con supraíndice (como x^k) harán referencia a los diferentes puntos obtenidos por el algoritmo. Los términos que tengan un * como supraíndice (como x^*) denotarán puntos óptimos o puntos solución. Las expresiones con subíndices (como x_i) denotarán componentes de vectores. Las letras griegas representarán escalares. Las letras mayúsculas (A, X , etc.) serán matrices. Las letras minúsculas podrán representar tanto vectores (x) como dimensiones de vectores (n, m), y se distinguirá según el contexto. La letra k se reserva para indicar el número de iteración del algoritmo (x^k denotará, por tanto, la solución que se tiene en la iteración k -ésima). La letra e denotará el vector todo de unos $e = (1, \dots, 1)^T$. Se seguirá la convención de que los vectores serán vectores columna, y sus traspuestos vectores fila.

3. Presentación del problema y requisitos necesarios

El problema lineal que queremos solucionar (y que denotaremos por (P)) estará en forma estándar:

$$\begin{aligned} \min_x \quad & c^T x \\ \text{sujeto a} \quad & Ax = b \\ & x \geq 0 \end{aligned} \tag{P}$$

donde $x \in \mathbb{R}^n$ es nuestro vector de variables, $A \in \mathbb{R}^{m \times n}$ es la matriz de restricciones del problema (donde $m < n$), $b \in \mathbb{R}^m$ el vector de términos independientes, y $c \in \mathbb{R}^n$ el vector de costes lineales. El conjunto de puntos $\{x \mid Ax = b, x \geq 0\}$ se denomina *región factible* (o *polígono factible*). Un punto que pertenece a este conjunto se denomina *punto factible*. Consideraremos las siguientes hipótesis acerca de (P) .

Hipótesis 1. La matriz A es de rango completo, esto es, $\text{rango}(A) = m$.

Hipótesis 2. El problema (P) no es degenerado. Esto significa que todo punto x factible para (P) tiene como mínimo m componentes diferentes de 0.

La interpretación de no-degeneración coincide con la del método del símplex, donde se denomina degenerada a una solución básica $x = [x_B^T \ x_N^T]^T$ ($x_B \in \mathbb{R}^m$, $x_N \in \mathbb{R}^{n-m}$, $x_N = 0$, $x_B \geq 0$) que tiene alguna componente de x_B igual a 0. Esta hipótesis facilita la presentación y justificación de determinadas etapas del método, así como su convergencia.

Entre los requisitos necesarios para desarrollar el método del escalado afín hallamos algunos conceptos básicos de teoría de la dualidad. En primer lugar debemos saber que el problema (P) tiene asociado un problema dual (D) , que tiene la siguiente formulación:

$$\begin{aligned} \max_{y,z} \quad & b^T y \\ \text{sujeto a} \quad & A^T y + z = c \\ & z \geq 0, \end{aligned} \tag{D}$$

donde $y \in \mathbb{R}^m$ es el vector de variables duales, y $z \in \mathbb{R}^n$ es el vector denominado de holguras duales. Al igual que para el problema primal, consideraremos que el dual no es degenerado (esta hipótesis será únicamente utilizada para demostrar la convergencia del método).

Hipótesis 3. El problema (D) no es degenerado. Esto significa que todo punto (y, z) factible para (D) tiene como máximo m componentes de z iguales a 0.

Las Hipótesis 2 y 3 garantizan que los problemas primal y dual, si son factibles, tienen un único punto óptimo.

Las relaciones entre los problemas (P) y (D) nos vienen dadas por los Teoremas de la Dualidad Débil, Dualidad Fuerte, y de la Holgura Complementaria, que también serán utilizados durante la exposición:

Teorema 1. (Teorema de la Dualidad Débil) Si x es factible para el problema primal (P) , e y es factible para el problema dual (D) , entonces

$$c^T x \geq b^T y.$$

Teorema 2. (Teorema de la Dualidad Fuerte) Si el problema primal (P) tiene una solución óptima x^* , entonces el problema dual también tiene una solución óptima y^* de forma que

$$c^T x^* = b^T y^*.$$

Teorema 3. (Teorema de la Holgura Complementaria) Si x es factible para el problema primal (P) , y el par (y, z) es factible para el problema dual (D) , entonces son óptimos de sus respectivos problemas si y solamente si

$$x_i z_i = 0 \quad \text{para todo } i = 1, \dots, n.$$

La distancia $c^T x - b^T y$ entre las funciones objetivo primal y dual se denominará *gap dual*. En el óptimo de ambos problemas, el gap dual es 0.

En determinados puntos del desarrollo se utilizarán también algunos conceptos de programación no lineal, como las condiciones de optimalidad de problemas sujetos a restricciones de igualdad (método de los multiplicadores de Lagrange). Estos conceptos se usarán para justificar formalmente determinados pasos del algoritmo. En particular, usaremos el siguiente resultado:

Teorema 4. (Condiciones necesarias y suficientes de optimalidad para un problema con restricciones de igualdad) Dado el problema

$$\begin{aligned} \min_x \quad & f(x) \\ \text{sujeto a} \quad & h(x) = 0, \end{aligned}$$

donde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ y $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ son continuas hasta sus segundas derivadas, y considerando la función Lagrangiana definida como

$$L(x, y) = f(x) - y^T h(x),$$

donde $y \in \mathbb{R}^m$ recibe el nombre de vector de multiplicadores de Lagrange, entonces es condición necesaria para que x^* sea mínimo del problema que, si los gradientes de las restricciones en x^* son independientes, exista un vector de multiplicadores y^* tal que

$$\nabla_{(x,y)} L(x^*, y^*) = 0 \iff \begin{cases} \nabla f(x^*) - \nabla h(x^*)^T y^* = 0 \\ h(x^*) = 0 \end{cases} .$$

Y es condición suficiente para que sea mínimo que, además de la condición necesaria anterior, se verifique

$$v^T [\nabla_{xx}^2 L(x^*, y^*)] v > 0 \quad \text{para todo } v \neq 0 \text{ tal que } \nabla_x h(x^*) v = 0 .$$

4. Métodos de punto interior para programación lineal

Los métodos de punto interior constituyen una familia de técnicas no-síplex para programación lineal. De hecho, puede decirse que se basan en la aplicación a problemas lineales de métodos que clásicamente habían sido considerados de programación no lineal. Por ello la mayoría de obras recientes de programación no lineal incluyen este tipo de técnicas (Bertsekas (1995)).

Se denominan métodos de punto interior precisamente porque los puntos generados por estos algoritmos se hallan en el interior de la región factible. Esta es una clara diferencia respecto el método del símplex, el cual avanza por la frontera de dicha región moviéndose de un punto extremo a otro. La Fig. 1 muestra gráficamente ambos comportamientos. Puede comprobarse como el método de punto interior requiere que el punto inicial x^0 sea también un punto interior a la región factible.

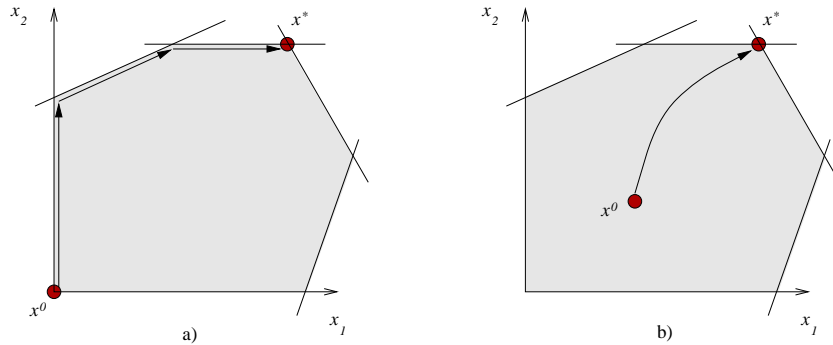


Figura 1. Trayectoria seguida para alcanzar el punto óptimo x^* desde el punto inicial x^0 por

- a) el método del símplex
- b) un método de punto interior

Otra de las diferencias fundamentales respecto el método del simplex es que existen algoritmos de punto interior polinómicos. Esto significa que obtienen el punto óptimo en un número de iteraciones que es función polinómica del tamaño del problema. El método del simplex, en el peor de los casos, tiene un coste exponencial. En la actualidad los métodos de punto interior más eficientes tienen una complejidad de $O(\sqrt{n}L)$, donde n es el número de variables del problema y L una medida del tamaño del problema (a saber, el número de bits necesarios para representar los datos del problema).

Podemos clasificar los métodos de punto interior en cuatro grandes categorías:

- **Métodos de escalado afín.** Son los más sencillos, y transforman el problema original mediante un escalado afín de las variables. En la sección 5 se presentan con todo detalle.
- **Métodos basados en transformaciones proyectivas.** Al igual que los anteriores transforman el problema original, aunque de forma más elaborada, utilizando lo que se denomina una *transformación proyectiva*. Estas transformaciones son del tipo:

$$\tilde{x} = \frac{X^{-1}x}{e^T X^{-1}x} \quad ,$$

donde $x \in \mathbb{R}^n$ es el punto actual, \tilde{x} representa la imagen de x en el nuevo espacio de variables, X es una matriz diagonal que tiene como términos diagonales las componentes de x ($X = \text{diag}(x_1, \dots, x_n)$), y e^T es el vector n -dimensional de unos ($e^T = (1_1, \dots, 1_n)$). El ahora ya famoso primer algoritmo de punto interior, desarrollado por Karmarkar en 1984, pertenece a esta categoría.

- **Los denominados métodos *path-following*** (se hace difícil traducir este término, por lo que preferimos mantener el original en inglés). Estos métodos utilizan una técnica para problemas no lineales, denominada *barrera logarítmica*. En vez de solucionar (P) directamente, solucionan una secuencia de problemas de la forma

$$\begin{aligned} \min_x \quad & c^T x - \mu \sum_{j=1}^n \ln(x_j) \\ \text{sujeto a} \quad & Ax = b \quad . \end{aligned} \tag{P_\mu}$$

Puede probarse que, a medida que $\mu \rightarrow 0$ la secuencia de soluciones de (P_μ) se acerca a la solución del problema (P) original. El conjunto de soluciones de (P_μ) para diferentes μ proporciona el denominado camino central (*central path*). Los métodos *path-following* siguen una trayectoria cercana a este camino central en su búsqueda del óptimo. El algoritmo primal-dual de punto interior pertenece a esta categoría. Este algoritmo ha mostrado ser, en general, el método de punto interior más eficiente. Hoy en día es considerado como la mejor opción para solucionar problemas lineales de gran dimensión.

- **Los métodos de reducción de potencial**, que intentan reducir una determinada función de mérito o de potencial.

Nosotros nos centraremos en la primera categoría de métodos: los de escalado afín.

5. Método del escalado afín

Este tipo de métodos son los más sencillos de todos los de punto interior. A pesar de ello, muestran un excelente rendimiento en la solución de problemas de gran dimensión. El método del escalado afín fue sugerido originalmente por el matemático ruso Dikin (Dikin (1967)) en 1967. Este trabajo, sin embargo, pasó desapercibido hasta que un par de décadas más tarde el método presentado fue redescubierto en Occidente de forma independiente por diferentes

investigadores como una simplificación del algoritmo de las transformaciones proyectivas de Karmarkar (Barnes (1986), Vanderbei *et al.* (1986)).

Las principales características del método del escalado afín son:

- Es de comprensión sencilla.
- Puede ser fácilmente implementado.
- Por lo general, proporciona muy buenos resultados.
- Puede verse como una aplicación de una variante de un método de programación no lineal —denominado del *gradiente proyectado*— a la solución de un problema lineal.

5.1. Algoritmo básico

Nuestro objetivo es solucionar el problema lineal formulado en $(P) \min_x c^T x$ sujeto a $Ax = b$, $x \geq 0$ (recordemos que contamos con n variables y m restricciones). Para desarrollar el algoritmo del escalado afín consideraremos que disponemos de un punto interior y factible $x^0 \in \mathbb{R}^n$. Que sea factible significa que satisface las restricciones del problema. Que sea interior quiere decir que no contacta con las paredes de la región factible, esto es, que $x > 0$. Por tanto, nuestro punto interior y factible x^0 inicial garantiza:

$$Ax^0 = b \quad \text{y} \quad x^0 > 0.$$

EJEMPLO 1

Dado el problema lineal siguiente (a la izquierda se muestra el problema original y a la derecha su forma estándar tras haber introducido las variables x_3 , x_4 y x_5)

$\begin{aligned} \min \quad & -3x_1 - 2x_2 \\ \text{sujeto a} \quad & \\ & 4x_1 - 2x_2 \leq 5 \\ & 3x_1 + 4x_2 \geq 1 \\ & x_1 + x_2 \leq 2 \\ & x_1 \geq 0 \quad x_2 \geq 0 \end{aligned}$	\implies	$\begin{aligned} \min \quad & -3x_1 - 2x_2 \\ \text{sujeto a} \quad & \\ & 4x_1 - 2x_2 + x_3 = 5 \\ & 3x_1 + 4x_2 - x_4 = 1 \\ & x_1 + x_2 + x_5 = 2 \\ & x_i \geq 0 \text{ para toda } i = 1, \dots, 5 \end{aligned}$
---	------------	--

puede comprobarse como el punto $x^0 = (1/2 \ 1/2 \ 4 \ 5/2 \ 1)^T$ es interior (todas las componentes son mayores que 0) y factible (satisface las restricciones del problema en forma estándar):

$$\begin{array}{rclcl} 4 \cdot 1/2 & -2 \cdot 1/2 & +4 & & = 5 \\ 3 \cdot 1/2 & +4 \cdot 1/2 & & -5/2 & = 1 \\ 1/2 & + & 1/2 & & + 1 = 2 \end{array}$$

La Fig. 2 muestra la situación del punto x^0 dentro de la región factible, resaltada en oscuro. La frontera de la región factible viene delimitada por las tres restricciones (de desigualdad). Se consideran únicamente las dos primeras componentes de x^0 , puesto que usamos la formulación original del problema, no la forma estándar, para permitir su representación en el plano.

El algoritmo básico seguido por el método del escalado afín consiste en, comenzando por x^0 , generar una secuencia de puntos $\{x^k\}$ que nos lleven hasta el punto óptimo x^* del problema. Así, partiendo de x^0 avanzaremos hasta x^1 , de éste hasta x^2 , y así sucesivamente, hasta que se verifique alguna condición que nos indique que el punto actual x^k es solución de nuestro problema. Cada nuevo punto se obtendrá del anterior, siguiendo el siguiente procedimiento iterativo

$$x^{k+1} = x^k + \alpha^k \Delta x^k, \quad \alpha^k \geq 0. \tag{1}$$

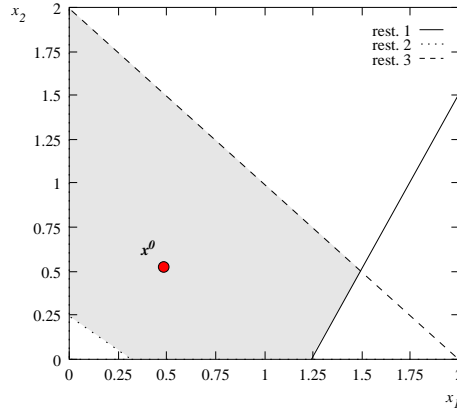


Figura 2. Situación del punto x^0 dentro de la región factible delimitada por las tres restricciones del ejemplo 1: rest. 1 $\equiv 4x_1 - 2x_2 \leq 5$, rest. 2 $\equiv 3x_1 + 4x_2 \geq 1$, rest. 3 $\equiv x_1 + x_2 \leq 2$.

Aquí $\Delta x^k \in \mathbb{R}^n$ representa nuestro *vector o dirección de movimiento*, mientras que el escalar α^k , denominado *longitud de paso*, es positivo y nos indica cuanto nos alejamos de x^k a lo largo de la dirección calculada. Este esquema es común a la mayoría de métodos de Optimización.

Por tanto debemos centrarnos en las dos siguientes cuestiones: i) ¿Cómo obtener la dirección de movimiento Δx^k ? y ii) ¿Cómo calcular la longitud de paso α^k ? La primera cuestión va a ocupar la mayor parte del resto de secciones. El cálculo de la longitud de paso es sencillo una vez se ha obtenido la dirección de movimiento, tal y como veremos.

5.2. Cálculo de la dirección de movimiento

La dirección de movimiento Δx calculada a cada iteración del algoritmo no puede ser una dirección cualquiera. Concretamente, Δx debe verificar dos condiciones para ser considerada una “buena” dirección:

- 1) Debe preservar la factibilidad del nuevo punto (dirección factible).
- 2) Debe mejorar el valor de la función objetivo (mejorar significa disminuir en este caso, puesto que el problema (P) es de minimización).

Estas condiciones son tratadas en los dos siguientes apartados.

(Nótese que en el párrafo anterior hemos escrito Δx y no Δx^k , como en la sección previa. Para simplificar la notación preferimos eliminar el supraíndice k . Por tanto Δx indicará la dirección de movimiento en el punto actual x^k . Seguiremos este acuerdo para la mayoría de términos que aparecerán de ahora en adelante —entre ellos la longitud de paso, que se denotará por α .)

5.2.1. Obtención de una dirección factible

Supongamos que nos hallamos en el punto factible e interior x^k . En el siguiente punto x^{k+1} queremos que se mantenga la factibilidad de nuestra solución. De momento nos preocuparemos únicamente por las restricciones de igualdad $Ax = b$; las restricciones $x \geq 0$ pueden ser satisfechas de forma simple, y serán consideradas más adelante. Por tanto, el nuevo punto x^{k+1} debe garantizar

$$Ax^{k+1} = b.$$

Sabemos por (1) que el punto x^{k+1} se obtiene como

$$x^{k+1} = x^k + \alpha \Delta x,$$

y que el punto x^k verifica $Ax^k = b$ por ser factible. Entonces, la condición de factibilidad de x^{k+1} se reduce a

$$\left. \begin{aligned} b &= Ax^{k+1} \\ &= A(x^k + \alpha \Delta x) \\ &= Ax^k + \alpha A \Delta x \\ &= b + \alpha A \Delta x \end{aligned} \right\} \Rightarrow A \Delta x = 0.$$

Concluimos que, para que x^{k+1} sea factible, debe verificarse que la dirección de movimiento debe pertenecer al espacio nulo de A definido como $N(A) = \{v \mid Av = 0, v \in \mathbb{R}^n\}$. Sin embargo, dado un vector cualquiera $d \in \mathbb{R}^n$, éste siempre puede ser proyectado sobre $N(A)$, si A tiene rango completo, premultiplicándolo por la siguiente matriz de proyección ortogonal

$$P = I_n - A^T(AA^T)^{-1}A, \quad (2)$$

donde I_n denota la matriz identidad de dimensión n . En el ejercicio 1 se plantea la deducción de este resultado. Esta matriz de proyección ortogonal verifica las siguientes propiedades

$$\begin{aligned} AP &= \mathbf{0} \\ P &= P^T \quad (\text{simetría}) \\ P^2 &= P \quad (\text{idempotencia}). \end{aligned} \quad (3)$$

(ver el ejercicio 2 para una comprobación de las mismas).

EJEMPLO 2

Consideremos el siguiente problema lineal

$$\begin{aligned} \min \quad & -\frac{1}{3}x_1 - \frac{2}{3}x_2 \\ \text{sujeto a} \quad & x_1 + x_2 + x_3 = 1 \\ & x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \end{aligned}$$

En este problema $m = 1$, $n = 3$ y $A = (1 \ 1 \ 1)$, y el óptimo viene dado claramente por $x^* = (0 \ 1 \ 0)^T$. El punto actual factible es $x^k = (1/3 \ 1/3 \ 1/3)^T$. La matriz de proyección ortogonal sobre el espacio nulo de A se calcula como:

$$\begin{aligned} P &= I_n - A^T(AA^T)^{-1}A = \begin{pmatrix} 1 & & \\ & 1 & \\ & & 1 \end{pmatrix} - \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \left((1 \ 1 \ 1) \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right)^{-1} (1 \ 1 \ 1) \\ &= \begin{pmatrix} 1 & & \\ & 1 & \\ & & 1 \end{pmatrix} - 1/3 \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} = 1/3 \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}. \end{aligned}$$

Dado un vector cualquiera $d = (0 \ -1/2 \ 0)^T$, podemos calcular su proyección sobre el espacio nulo de A

$$Pd = 1/3 \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \begin{pmatrix} 0 \\ -1/2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1/6 \\ -1/3 \\ 1/6 \end{pmatrix}.$$

Podemos comprobar que la dirección Pd así obtenida es factible observando que el punto $x^k + Pd$ satisface la restricción de igualdad del problema:

$$x^k + Pd = \begin{pmatrix} 1/3 \\ 1/3 \\ 1/3 \end{pmatrix} + \begin{pmatrix} 1/6 \\ -1/3 \\ 1/6 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 0 \\ 1/2 \end{pmatrix} \quad 1/2 + 0 + 1/2 = 1.$$

La Fig. 3 a) ilustra gráficamente este ejemplo.

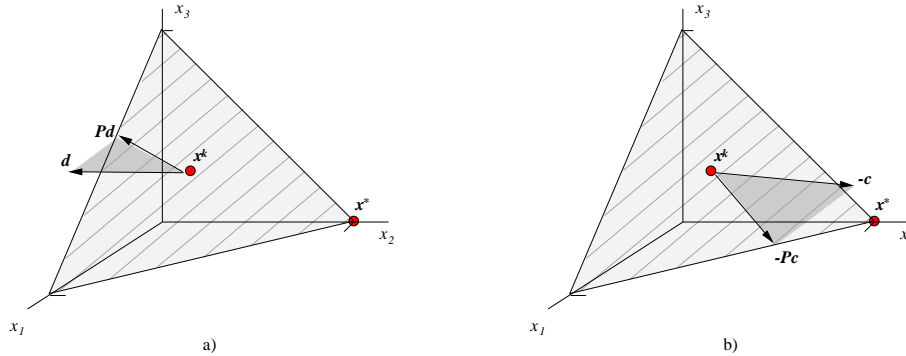


Figura 3. Representación gráfica de los ejemplos 2 y 3. Se detalla la región factible $\{x_1 + x_2 + x_3 = 1, x_i \geq 0\}$, y sobre ésta el punto actual x^k , y las curvas de nivel de la función objetivo.

- Proyección ortogonal del vector $d = (0 \ -1/2 \ 0)^T$. El nuevo punto $x^k + Pd$ es $(1/2 \ 0 \ 1/2)^T$ y pertenece a la región factible.
- Proyección ortogonal del vector $-c = (1/3 \ 2/3 \ 0)^T$. El nuevo punto $x^k - Pc$ es $(1/3 \ 2/3 \ 0)^T$ y mejora el valor de la función objetivo.

Por tanto, dada una dirección de movimiento cualquiera $d \in \mathbb{R}^n$, siempre se verificará que su proyección ortogonal sobre el espacio nulo de A , Pd , es una dirección factible para nuestro problema. Esto es, que $APd = 0$ (se deduce inmediatamente de la primera propiedad de (3)). Sin embargo, cualquier dirección no siempre nos permitirá mejorar nuestra función de coste. Así, el punto $x^k + Pd$ obtenido en el ejemplo 2 tiene un coste superior al del punto original x^k :

$$c^T(x^k + Pd) = -1/3 \cdot 1/2 - 2/3 \cdot 0 = -1/6 > c^T x^k = -1/3 \cdot 1/3 - 2/3 \cdot 1/3 = -1/3.$$

También se observa este hecho en la Fig. 3 a): nos hemos movido hacia curvas de nivel de coste superior al del punto x^k . Nuestro objetivo es ahora escoger, de entre todas las direcciones posibles, una que nos garantice una mejora de la función objetivo.

5.2.2. Mejora de la función objetivo

En un problema de minimización como el que estamos considerando, en el nuevo punto x^{k+1} debemos tener un coste inferior al del punto x^k . Esto significa que

$$c^T x^{k+1} \leq c^T x^k.$$

Utilizando que $x^{k+1} = x^k + \alpha \Delta x$, y teniendo en cuenta que α es siempre positivo, podemos escribir la condición anterior como

$$c^T x^{k+1} = c^T(x^k + \alpha \Delta x) = c^T x^k + \alpha c^T \Delta x \leq c^T x^k \implies c^T \Delta x \leq 0.$$

Esta condición se denomina *condición de descenso*.

Consideremos la siguiente dirección Δx

$$\Delta x = -Pc, \quad (4)$$

que no es sino la proyección de $-c$ (menos el gradiente de nuestra función lineal) sobre el espacio nulo de A . Por el apartado anterior sabemos que esta dirección es factible. Y utilizando la simetría e idempotencia de P puede demostrarse que también es de descenso:

$$c^T \Delta x = -c^T Pc = -c^T P^2 c = -c^T P^T Pc = -\|Pc\|^2 \leq 0.$$

A pesar de aparecer un \leq en la última desigualdad anterior, en la práctica siempre será de $<$. Esto nos evita la situación de que el algoritmo avanzase sin disminuir la función objetivo, lo que podría dar lugar a un caso de ciclaje y pérdida de convergencia. El ejercicio 3 plantea la cuestión de bajo qué circunstancias se tiene que $\|Pc\|^2 = 0$.

EJEMPLO 3

Consideremos de nuevo el problema del ejemplo 2, utilizando ahora como dirección de movimiento $-c = (1/3 \ 2/3 \ 0)^T$ (si este movimiento fuera excesivo y no garantizase la factibilidad del siguiente iterado, utilizaríamos $d = -\beta c$, con $0 < \beta < 1$). El menos gradiente proyectado sobre $N(A)$ es ahora

$$-Pc = 1/3 \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \begin{pmatrix} 1/3 \\ 2/3 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1/3 \\ -1/3 \end{pmatrix}.$$

El nuevo punto obtenido es

$$x^k - Pc = \begin{pmatrix} 1/3 \\ 1/3 \\ 1/3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1/3 \\ -1/3 \end{pmatrix} = \begin{pmatrix} 1/3 \\ 2/3 \\ 0 \end{pmatrix} \quad 1/3 + 2/3 + 0 = 1.$$

A diferencia del ejemplo 2, observamos como ahora sí hemos disminuido el valor de nuestra función objetivo

$$c^T(x^k - Pc) = -1/3 \cdot 1/3 - 2/3 \cdot 2/3 = -5/9 \leq c^T x^k = -1/3.$$

La Fig. 3 b) ilustra gráficamente esta situación.

Por tanto hemos obtenido una dirección adecuada a nuestros propósitos: factible y de descenso. La elección de $-c$ para proyectar sobre el espacio nulo de A no ha sido totalmente arbitraria: la dirección de menos el gradiente es precisamente la utilizada en métodos básicos de Optimización sin restricciones como el de la *máxima pendiente* o del *gradiente*, introducido ya por Cauchy en 1847. El siguiente apartado da una justificación más formal de dicha elección.

5.2.3. Justificación de la dirección obtenida

Supongamos que queremos hallar la dirección factible Δx que nos proporciona el máximo descenso (o disminución del valor de la función objetivo) a partir del punto x^k . Para ello planteamos el siguiente problema de optimización:

$$\begin{aligned} \min_{\Delta x} \quad & c^T(x^k + \Delta x) \\ \text{sujeto a} \quad & A(x^k + \Delta x) = b \\ & \|\Delta x\|^2 = \rho^2. \end{aligned} \quad (5)$$

La segunda restricción nos indica que la dirección que se obtendrá debe tener una longitud (norma) de ρ , donde este valor se ajusta para evitar que el nuevo punto $x^k + \Delta x$ viole la no-negatividad de las variables. Es, por tanto, una forma de tratar las restricciones $x \geq 0$. También garantiza que el problema planteado esté acotado (esto es, que no tienda hacia $-\infty$).

Para nuestros propósitos, sin embargo, no nos importa la longitud de Δx (siempre podemos acortarla posteriormente), sino su dirección. Notemos que el problema planteado no es más que la minimización de la función objetivo dentro de la región formada por la intersección del subespacio $A(x^k + \Delta x) = b$ y una esfera centrada en x^k y de radio ρ , tal y como muestra la Fig. 4 para un caso de 3 variables y una restricción.

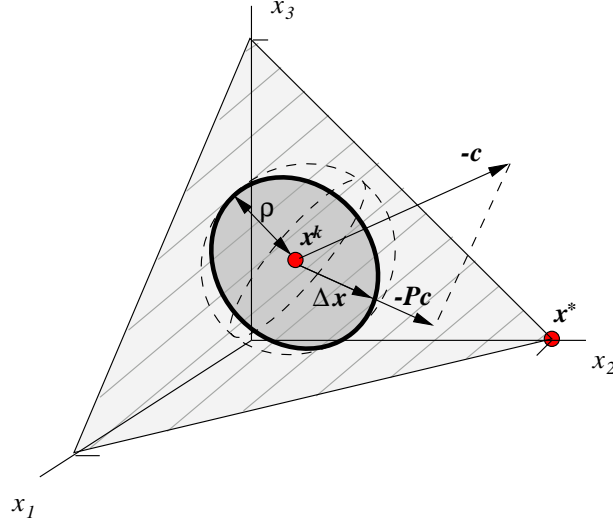


Figura 4. Región factible (línea de trazo grueso) consistente en la intersección del subespacio $A(x^k + \Delta x) = b$ (en gris claro) y una esfera centrada en x^k y de radio ρ (línea discontinua). También se muestra el gradiente proyectado $-Pc$ y la dirección $\Delta x = -\rho Pc / \|Pc\|$ que proporciona el máximo descenso desde el punto x^k hasta un punto de la región factible.

El problema (5) anterior puede ser solucionado mediante el método de los multiplicadores de Lagrange (Teorema 4), considerando la función Lagrangiana

$$L(\Delta x, y, \lambda) = c^T(x^k + \Delta x) - y^T(A(x^k + \Delta x) - b) + \lambda(\Delta x^T \Delta x - \rho^2).$$

El vector $y \in \mathbb{R}^m$ corresponde a los multiplicadores del primer conjunto de restricciones, mientras que el escalar λ es el multiplicador de la restricción de la norma de Δx . La razón por la que el término con el multiplicador λ es positivo, a diferencia del de los multiplicadores y , será mostrada posteriormente. Derivando la función para Δx , y y λ , e igualando a 0, obtenemos

$$\begin{aligned} c - A^T y + 2\lambda \Delta x &= 0 \\ A(x^k + \Delta x) - b &= 0 \\ \Delta x^T \Delta x - \rho^2 &= 0. \end{aligned}$$

Como $Ax^k = b$ por ser x^k factible, la segunda ecuación anterior nos indica que $A\Delta x = 0$. Premultiplicando la primera ecuación por A y utilizando que $A\Delta x = 0$ obtenemos

$$Ac = AA^T y.$$

Como A tiene rango completo, podemos aislar y de la expresión anterior:

$$y = (AA^T)^{-1} Ac. \quad (6)$$

Sustituyendo (6) en la primera ecuación, y utilizando la definición de P en (2), obtenemos

$$\begin{aligned} 2\lambda\Delta x &= -c + A^T(AA^T)^{-1}Ac \\ &= -(I_n - A^T(AA^T)^{-1}A)c = -Pc \\ \Delta x &= -\frac{1}{2\lambda}Pc. \end{aligned}$$

Usando la expresión anterior, el valor de λ puede ajustarse para garantizar que $\|\Delta x\|^2 = \rho^2$:

$$\rho^2 = \|\Delta x\|^2 = \frac{1}{4\lambda^2}\|Pc\|^2 \Rightarrow \lambda = \frac{\|Pc\|}{(2\rho)}.$$

Sustituyendo esta expresión en la anteriormente hallada de Δx se obtiene que

$$\Delta x = -\frac{\rho}{\|Pc\|}Pc. \quad (7)$$

Para estar seguros de que este punto corresponde a un mínimo (y no a un máximo) debemos verificar la condición suficiente de mínimo del Teorema 4. Derivando dos veces la Lagrangiana respecto Δx obtenemos

$$\nabla_{\Delta x \Delta x} L(\Delta x, y, \lambda) = 2\lambda.$$

Este término siempre es positivo, ya que hemos obtenido anteriormente que $\lambda = \|Pc\|/(2\rho) > 0$, por lo que la expresión (7) nos proporciona el mínimo de nuestro problema. Esto justifica el por qué hemos considerado un signo positivo para el término λ en la función Lagrangiana: si hubiéramos utilizado un signo menos, la solución obtenida hubiera sido la dirección contraria a (7), verificándose entonces que $\nabla_{\Delta x \Delta x} = -2\lambda$, lo que nos indicaría que esta dirección contraria corresponde a un máximo de nuestro problema en vez de a un mínimo.

Dejando a parte el escalar $\rho/\|Pc\|$, la dirección obtenida coincide con la presentada en (4). Ésta es la dirección que nos garantiza el máximo descenso a partir de un punto manteniendo la factibilidad del siguiente. Recibe el nombre de dirección de (menos) *el gradiente proyectado* (omitiremos el “menos” al referirnos a esta dirección de aquí en adelante), y coincide con la utilizada en el método del *gradiente proyectado* para Programación No Lineal. La Fig. 4 muestra la proyección del gradiente sobre la región factible de nuestro problema.

5.3. Escalado

El gradiente proyectado nos proporciona una excelente dirección, en el sentido que entre todas las direcciones factibles es la que tiene la máxima disminución de $c^T x$ por unidad de longitud de paso. Sin embargo, y según la expresión (1) que rige el proceso iterativo del método, si nos hallamos muy cerca de las paredes de la región factible definidas por las restricciones $x \geq 0$, esta longitud de paso α que podremos realizar será pequeña (de otro modo, violaríamos la no-negatividad de alguna variable). No es suficiente, por tanto, con tener una buena dirección de descenso; conviene también estar en un punto *centrado* respecto de la región factible (esto es, alejado de las paredes delimitadas por $x \geq 0$), lo que nos permitirá tener $\alpha \gg 0$. Este hecho se observa claramente en la Fig. 5, donde el movimiento desde el punto x^a es pequeño comparado con el realizado desde el punto x^b .

El método del escalado afín solventa la dificultad de no tener un punto centrado mediante el siguiente procedimiento de 3 etapas, que se repiten a cada iteración del algoritmo:

1. Escalar el problema de forma que el punto actual se encuentre alejado de las paredes delimitadas por $x \geq 0$. El punto actual x^k queda transformado en \tilde{x}^k , en el nuevo espacio de variables \tilde{x} .

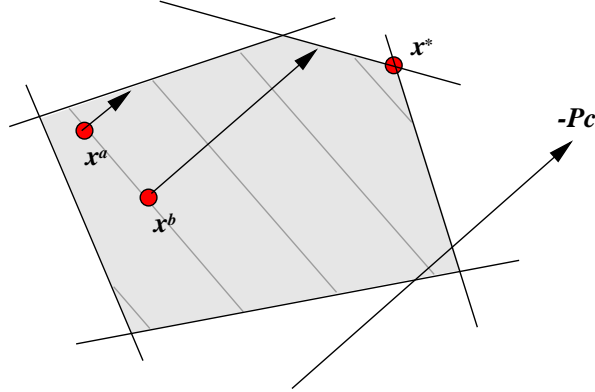


Figura 5. Movimiento realizado en la dirección del gradiente proyectado $-Pc$ desde un punto “centrado” x^b y desde un punto “no-centrado” x^a .

2. Calcular la dirección del gradiente proyectado en el nuevo problema escalado. En este nuevo espacio de variables el gradiente proyectado se denota por $\Delta\tilde{x}$.
3. Trasladar la dirección $\Delta\tilde{x}$ obtenida al problema original, deshaciendo el escalado previo. A partir de $\Delta\tilde{x}$ obtenemos por tanto la dirección Δx en el espacio original de variables.

Ahora debemos determinar qué tipo de escalado debe utilizarse. Cualquier escalado no sirve. Así, por ejemplo, un escalado que multiplicase todas las componentes x_i^k $i = 1, \dots, n$ por un mismo valor no sería válido para nuestros propósitos, porque se mantendría la distancia relativa de todas las variables respecto las paredes definidas por $x \geq 0$.

La estrategia seguida por el método del escalado afín consiste en transformar cada componente x_i^k $i = 1, \dots, n$ del espacio de variables x en un 1 en el nuevo espacio \tilde{x} . En este nuevo espacio todas las componentes del punto \tilde{x}^k se encuentran alejadas por igual de 0. Para conseguir esto debemos dividir cada componente x_i^k por ella misma. El cambio de variables que se realiza es por tanto:

$$\tilde{x}_i = \frac{x_i}{x_i^k} \quad i = 1, \dots, n,$$

donde x^k es el punto actual de iteración. Este cambio se deshace de forma trivial:

$$x_i = \tilde{x}_i x_i^k \quad i = 1, \dots, n.$$

Si denotamos por X^k la matriz diagonal que tiene por términos diagonales las componentes del punto actual x_i^k ,

$$X^k = \begin{pmatrix} x_1^k & & & \\ & x_2^k & & \\ & & \ddots & \\ & & & x_n^k \end{pmatrix},$$

podemos escribir las operaciones anteriores de cambio de variable en forma matricial:

$$\begin{aligned} \tilde{x} &= (X^k)^{-1}x \\ x &= X^k \tilde{x}. \end{aligned} \tag{8}$$

Observemos que este escalado está bien definido porque siempre nos movemos por puntos interiores de la región factible, con lo que $x_i^k > 0$, siendo entonces X^k no singular y definida positiva.

Veamos ahora en los tres apartados siguientes como se realizan a cada iteración del algoritmo las 3 etapas antes expuestas. Supondremos que nos hallamos en un punto x^k factible e interior determinado.

1. Escalado del problema original: $x \rightarrow \tilde{x}$

El problema primal (P) que estamos solucionando se escala utilizando el cambio de variables $x = X^k \tilde{x}$ de la siguiente manera:

$$\begin{array}{ccc} \min_x c^T x & \min_{\tilde{x}} c^T X^k \tilde{x} & \min_x \tilde{c}^T \tilde{x} \\ \text{sujeto a } Ax = b & \text{sujeto a } AX^k \tilde{x} = b & \text{sujeto a } \tilde{A} \tilde{x} = b \quad (\tilde{P}) \\ x \geq 0 & X^k \tilde{x} \geq 0 & \tilde{x} \geq 0, \end{array}$$

donde, en el último paso, se ha utilizado la notación

$$\begin{aligned} \tilde{c} &= X^k c \\ \tilde{A} &= AX^k. \end{aligned} \quad (9)$$

También conviene notar que en el último paso las restricciones $X^k \tilde{x} \geq 0$ se han transformado en $\tilde{x} \geq 0$. Ambos grupos de restricciones son equivalentes gracias a que las componentes del punto actual de iteración x^k son positivas.

El punto x^k que teníamos en (P) se transforma en el punto $\tilde{x}^k = e = (1, \dots, 1)^T$ que tiene una posición centrada en (\tilde{P}). La etapa 1 de la Fig. 6 muestra este proceso de escalado. Se observa como el escalado modifica las curvas de nivel de la función objetivo a lo largo de la región factible.

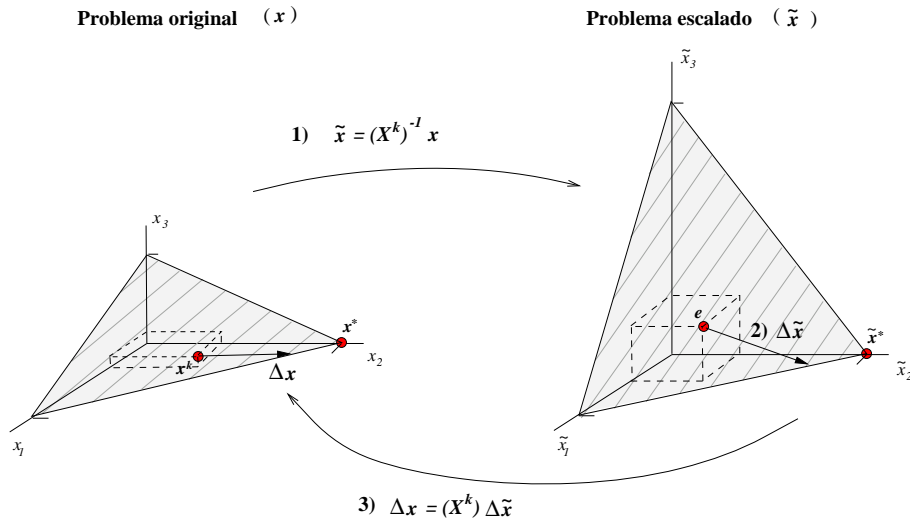


Figura 6. Representación gráfica de las tres etapas del proceso de escalado realizado a cada iteración del método del escalado afín.

- 1) Escalado del problema original, pasando del espacio de variables x al nuevo espacio \tilde{x} . El punto x^k queda transformado en el punto $e = (1, \dots, 1)^T$ en el nuevo espacio.
- 2) Cálculo de la dirección de movimiento $\Delta \tilde{x} = -\tilde{P} \tilde{c}$ en el nuevo espacio de variables.
- 3) Retorno al problema original, obteniendo la dirección de movimiento Δx a partir de $\Delta \tilde{x}$ aplicando el escalado inverso al utilizado en la etapa 1.

2. Cálculo de $\Delta\tilde{x}$

En el nuevo espacio de variables \tilde{x} la dirección de descenso $\Delta\tilde{x}$ será la del gradiente proyectado, calculada según la expresión (4). Sin embargo ahora utilizaremos la nueva matriz de restricciones \tilde{A} y el nuevo vector de costes \tilde{c} . Utilizando la definición de \tilde{A} y \tilde{c} en (9), y teniendo en cuenta que X^k es una matriz diagonal, el cálculo de $\Delta\tilde{x}$ es como sigue:

$$\begin{aligned}
\Delta\tilde{x} &= -\tilde{P}\tilde{c} \\
&= -(I_n - \tilde{A}^T(\tilde{A}\tilde{A}^T)^{-1}\tilde{A})\tilde{c} \\
&= -(I_n - (AX^k)^T(AX^k(AX^k)^T)^{-1}AX^k)X^k c \\
&= -X^k(I_n - A^T(A(X^k)^2A^T)^{-1}A(X^k)^2)c.
\end{aligned} \tag{10}$$

La etapa 2 de la Fig. 6 muestra la dirección $\Delta\tilde{x}$ en el problema escalado.

3. Retorno al problema original: $\tilde{x} \rightarrow x$

Si nos moviésemos en el problema escalado lo haríamos de la siguiente forma (ignoremos por el momento la longitud de paso):

$$\tilde{x}^{k+1} = \tilde{x}^k + \Delta\tilde{x}.$$

Si multiplicamos por X^k la expresión anterior deshacemos el cambio de variables realizado, según (8):

$$\begin{aligned}
X^k\tilde{x}^{k+1} &= X^k\tilde{x}^k + X^k\Delta\tilde{x} \\
x^{k+1} &= x^k + X^k\Delta\tilde{x}.
\end{aligned}$$

Por tanto, el movimiento Δx realizado en el espacio original de variables es igual a $X^k\Delta\tilde{x}$. Utilizando la expresión (10) de $\Delta\tilde{x}$ obtenida en la etapa 2 tenemos:

$$\begin{aligned}
\Delta x &= X^k\Delta\tilde{x} \\
&= -(X^k)^2(I_n - A^T(A(X^k)^2A^T)^{-1}A(X^k)^2)c \\
&= -D(c - A^T(ADA^T)^{-1}ADc) \quad [D = (X^k)^2] \\
&= -D(c - A^T y) \quad [y = (ADA^T)^{-1}ADc] \\
&= -Dz \quad [z = c - A^T y],
\end{aligned} \tag{11}$$

donde D , y y z se han definido como como

$$\begin{aligned}
D &= (X^k)^2 \\
y &= (ADA^T)^{-1}ADc \\
z &= c - A^T y.
\end{aligned} \tag{12}$$

La etapa 3 de la Fig. 6 muestra la dirección Δx obtenida en el problema original. En el ejercicio 4 se pide comprobar que esta dirección del gradiente proyectado con escalado es factible y de descenso.

Nótese que para obtener la dirección Δx debemos factorizar la matriz $ADA^T = A(X^k)^2A^T$. Para ello debe verificarse que AX^k sea una matriz de rango completo. Sin embargo, sabemos que A es de rango completo (Hipótesis 1) y que X^k tiene como mínimo m componentes > 0 por ser el problema no-degenerado (Hipótesis 2), concluyendo que $A(X^k)^2A^T$ es no singular. La siguiente proposición, que no demostramos, recoge este resultado.

Proposición 1.

Si (P) no es degenerado y la matriz de restricciones A tiene rango completo, entonces $A(X^k)^2 A^T$ es no singular. Además, los términos y y z definidos en (12) son una función continua de x^k .

A cada iteración del algoritmo las tres etapas anteriores se realizan implícitamente, y únicamente es necesario calcular la dirección Δx según la expresión (11). Esta será la dirección a utilizar en el proceso iterativo del método presentado en (1). El siguiente ejemplo ilustra todo lo expuesto hasta el momento.

EJEMPLO 4

Consideremos el problema siguiente (que ya fue utilizado en el ejemplo 1):

$$\begin{array}{ll} \min & -3x_1 - 2x_2 \\ \text{sujeto a} & \\ & 4x_1 - 2x_2 \leq 5 \\ & 3x_1 + 4x_2 \geq 1 \\ & x_1 + x_2 \leq 2 \\ & x_1 \geq 0 \quad x_2 \geq 0 \end{array} \quad \implies \quad \begin{array}{ll} \min & -3x_1 - 2x_2 \\ \text{sujeto a} & \\ & 4x_1 - 2x_2 + x_3 = 5 \\ & 3x_1 + 4x_2 - x_4 = 1 \\ & x_1 + x_2 + x_5 = 2 \\ & x_i \geq 0 \text{ para toda } i = 1, \dots, 5 \end{array}$$

y el punto actual $x^k = (1.25 \ 0.1 \ 0.2 \ 3.15 \ 0.65)^T$. Este punto es factible e interior —ver Fig. 7 a)—, y tiene un coste de $-3 \cdot 1.25 - 2 \cdot 0.1 = -3.95$. En este problema tenemos que

$$A = \begin{pmatrix} 4 & -2 & 1 & 0 & 0 \\ 3 & 4 & 0 & -1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix} \quad c = \begin{pmatrix} -3 \\ -2 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad X^k = \begin{pmatrix} 1.25 & & & & \\ & 0.1 & & & \\ & & 0.2 & & \\ & & & 3.15 & \\ & & & & 0.65 \end{pmatrix}.$$

Utilizando la matriz diagonal X^k anterior, y aplicando la ecuación (9), podemos construir nuestro problema escalado:

$$\begin{array}{ll} \min & -3.75 \tilde{x}_1 - 0.2 \tilde{x}_2 \\ \text{sujeto a} & \\ & 5 \tilde{x}_1 - 0.2 \tilde{x}_2 + 0.2 \tilde{x}_3 = 5 \\ & 3.75 \tilde{x}_1 + 0.4 \tilde{x}_2 - 3.15 \tilde{x}_4 = 1 \\ & 1.25 \tilde{x}_1 + 0.1 \tilde{x}_2 + 0.65 \tilde{x}_5 = 2 \\ & \tilde{x}_i \geq 0 \text{ para toda } i = 1, \dots, 5. \end{array}$$

Utilizando las expresiones anteriores de A , c y X^k , la dirección de movimiento en este nuevo espacio de variables se calcula según (10)

$$\Delta \tilde{x} = -X^k (I_n - A^T (A(X^k)^2 A^T)^{-1} A(X^k)^2) c = \begin{pmatrix} 0.0183 \\ 0.319 \\ -0.140 \\ 0.0623 \\ -0.0844 \end{pmatrix}.$$

La región factible del problema escalado y la dirección de movimiento $\Delta \tilde{x}$ obtenida se representan en la Fig. 7 b), considerando las variables \tilde{x}_3 , \tilde{x}_4 y \tilde{x}_5 como holguras. Nótese la diferencia de escala entre el eje de ordenadas y el de abscisas, debido al valor tan cercano a 0 que tiene la componente $x_2^k = 0.1$.

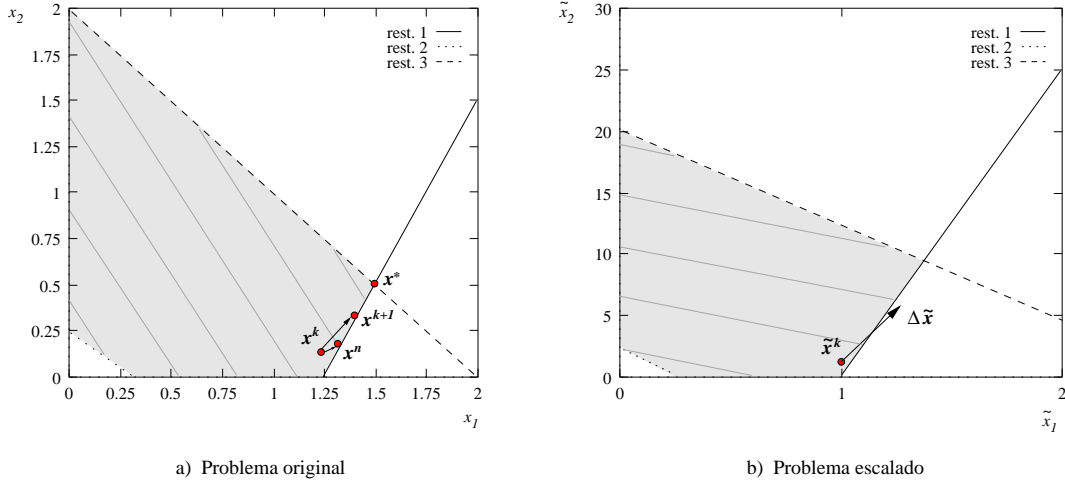


Figura 7. Representación gráfica del ejemplo 4.

- a) Problema original. Situación de los puntos x^k , x^{k+1} y x^n dentro de la región factible (en oscuro). En gris se resaltan las curvas de nivel de la función objetivo $-3x_1 - 2x_2$.
- b) Problema escalado. Se muestra el punto $\tilde{x}^k = (1, \dots, 1)^T$, la dirección $\Delta\tilde{x}$, y las curvas de nivel de la función objetivo del problema escalado $-3.75\tilde{x}_1 - 0.2\tilde{x}_2$.

Como ya contamos con $\Delta\tilde{x}$, la dirección de movimiento en el espacio de variables original la obtenemos de la siguiente forma:

$$\Delta x = X^k \Delta\tilde{x} = \begin{pmatrix} 0.0229 \\ 0.0318 \\ -0.0280 \\ 0.196 \\ -0.0548 \end{pmatrix}.$$

Podríamos haber calculado directamente Δx a través de (11). En esta dirección podemos avanzar, sin violar la no-negatividad de las variables, hasta el punto

$$x^{k+1} = \begin{pmatrix} 1.414 \\ 0.328 \\ 5 \cdot 10^{-7} \\ 4.55 \\ 0.258 \end{pmatrix}.$$

Veremos más adelante en la subsección 5.4 como se determina en la práctica este nuevo punto. El coste de la función objetivo en x^{k+1} es menor que en el punto x^k :

$$c^T x^{k+1} = -3 \cdot 1.414 - 2 \cdot 0.328 = -4.8984 < c^T x^k = -3.95.$$

Puede comprobarse que si no hubiéramos realizado ningún tipo de escalado en el problema original, hubiéramos obtenido la siguiente dirección de movimiento (del gradiente proyectado):

$$\Delta x^n = -Pc = \begin{pmatrix} 0.0984 \\ 0.0685 \\ -0.257 \\ 0.569 \\ -0.167 \end{pmatrix}.$$

Y avanzando por esta dirección, hasta donde nos permitan los límites $x \geq 0$ de las variables, hubiéramos obtenido el punto

$$x^n = \begin{pmatrix} 1.32 \\ 0.153 \\ 5 \cdot 10^{-7} \\ 3.59 \\ 0.520 \end{pmatrix},$$

que tiene un coste superior al del punto x^{k+1}

$$c^T x^n = -3 \cdot 1.32 - 2 \cdot 0.153 = -4.2870 > c^T x^{k+1} = -4.8984.$$

Por tanto, el escalado ha sido una estrategia efectiva, puesto que nos ha permitido reducir más nuestra función objetivo. En la Fig. 7 a) se muestran los puntos x^{k+1} y x^n obtenidos.

Del ejemplo anterior debemos destacar dos cosas. La primera es que el gradiente proyectado sin escalado nos ha proporcionado una dirección de movimiento peor que con escalado, en el sentido que hemos obtenido un coste superior. Sin embargo tiene un inconveniente todavía más grave. Si volviéramos a iterar con el gradiente proyectado, obtendríamos la misma dirección de movimiento, puesto que ésta es siempre constante e igual a $-Pc$. La consecuencia directa de esto es que no podremos avanzar más en esta dirección una vez hallamos llegado a alguna pared definida por $x \geq 0$. Así, pues, en el punto x^n obtenido en el ejemplo 4 se habrían acabado todas nuestras opciones de mejorar la función objetivo. El escalado, por contra, evita esta situación, ajustando la dirección de movimiento con la topología de la región factible.

El segundo aspecto a destacar del ejemplo anterior es que no se ha explicitado como calcular el nuevo punto x^{k+1} una vez se tiene Δx , esto es, qué longitud de paso α utilizar. Sin embargo lo que sí hemos observado es que se ha avanzado tanto como se ha podido en la dirección Δx , hasta llegar a anular la componente de alguna variable. Esta decisión parece razonable, ya que la disminución de coste que obtenemos al pasar de x^k a x^{k+1} es de $\alpha c^T \Delta x$ (se deduce directamente premultiplicando a ambos lados de (1) por c^T), y cuanto mayor sea α , más disminuiríamos nuestro coste. Utilizando este hecho, la siguiente subsección nos muestra cómo obtener la longitud de paso.

5.4. Determinación de la longitud de paso

Nuestro objetivo será movernos desde x^k a lo largo de Δx tanto como podamos mientras no violemos la no-negatividad de las variables. Esto es, debe garantizarse que

$$x^{k+1} \geq 0 \quad \implies \quad x^k + \alpha \Delta x \geq 0.$$

Como $\alpha > 0$, las componentes $\Delta x_i \geq 0$ no pueden llevar la variable x_i a 0. Por tanto únicamente debemos preocuparnos por los términos $\Delta x_i < 0$. El paso máximo, denotado por $\bar{\alpha}$ que podemos realizar viene dado entonces por

$$\bar{\alpha} = \min \left\{ -\frac{x_i^k}{\Delta x_i} \text{ para toda } i \text{ tal que } \Delta x_i \leq 0 \right\}. \quad (13)$$

Sin embargo, si definiésemos α de esta manera la componente i -ésima de x^{k+1} , donde i es el índice de la componente que ha proporcionado el mínimo $-x_i^k/\Delta x_i$ con $\Delta x_i < 0$, se anularía. Esto haría que x^{k+1} no fuese interior, y por tanto la matriz X^{k+1} de escalado sería singular. La estrategia a seguir para evitar este inconveniente consiste en reducir el paso máximo calculado

según la expresión anterior multiplicándolo por $\rho \in (0, 1)$ (en la práctica se utilizan valores entre 0.95 y 0.9995). Entonces α queda definido como

$$\alpha = \rho \cdot \bar{\alpha} = \rho \cdot \min \left\{ -\frac{x_i^k}{\Delta x_i} \text{ para toda } i \text{ tal que } \Delta x_i \leq 0 \right\} \quad \rho \in [0.95, 0.9995]. \quad (14)$$

El ejercicio 5 presenta una estrategia de comprobación de la ilimitación de un problema que está estrechamente relacionada con el cálculo de α .

EJEMPLO 5

En el ejemplo 4 se iteró a partir del punto $x^k = (1.25 \ 0.1 \ 0.2 \ 3.15 \ 0.65)^T$ obteniéndose una dirección $\Delta x = (0.0229 \ 0.0318 \ -0.0280 \ 0.196 \ -0.0548)^T$. Utilizando un valor de $\rho = 0.95$, la longitud de paso α se calcularía como:

$$\begin{aligned} \alpha &= \rho \cdot \min \left\{ -\frac{x_3^k}{\Delta x_3}, -\frac{x_5^k}{\Delta x_5} \right\} = 0.95 \cdot \min \left\{ -\frac{0.2}{-0.0280}, -\frac{0.65}{-0.0548} \right\} \\ &= 0.95 \cdot \min \{7.143, 11.861\} = 0.95 \cdot 7.143 = 6.7885. \end{aligned}$$

El nuevo punto sería por tanto

$$x^{k+1} = x^k + \alpha \Delta x = \begin{pmatrix} 1.25 \\ 0.1 \\ 0.2 \\ 3.15 \\ 0.65 \end{pmatrix} + 6.7885 \begin{pmatrix} 0.0229 \\ 0.0318 \\ -0.0280 \\ 0.196 \\ -0.0548 \end{pmatrix} = \begin{pmatrix} 1.406 \\ 0.316 \\ 0.0101 \\ 4.483 \\ 0.278 \end{pmatrix}.$$

Nótese como en el nuevo punto se mantiene más alejada la tercera componente x_3^{k+1} de 0 que en el ejemplo 4. Esto se debe a la utilización del término ρ de reducción de la longitud de paso.

5.5. Una digresión: el escalado afín de paso corto

Se ha observado como el método del escalado afín presentado intenta avanzar tanto como puede en la dirección de movimiento calculada. Por este motivo dicho método se conoce como *escalado afín de paso largo*. Es la variante más eficiente y fue descubierta hacia mediados de los ochenta (Vanderbei *et al.* (1986)). Recibe este nombre para diferenciarse del denominado *escalado afín de paso corto*, que, cronológicamente, apareció en primer lugar. Esta variante de paso corto es, de hecho, el método del escalado afín original, presentado por Dikin hacia finales de los 60 (Dikin (1967)). Algunos textos actuales utilizan esta versión para introducir las técnicas del escalado afín (por ejemplo, Bertsimas y Tsitsiklis (1997)). En esta subsección presentaremos brevemente dicho método. En el resto de la presentación, sin embargo, continuaremos tratando únicamente la versión de paso largo.

El método del escalado afín de paso corto, al igual que el de paso largo introducido con anterioridad, utiliza el cambio de variables $\tilde{x} = (X^k)^{-1}x$ presentado en (8), obteniendo entonces el siguiente problema en el espacio de variables \tilde{x}

$$\begin{aligned} \min_{\tilde{x}} \quad & \tilde{c}^T \tilde{x} \\ \text{sujeto a} \quad & \tilde{A} \tilde{x} = b \\ & \tilde{x} \geq 0, \end{aligned} \quad (15)$$

donde los términos \tilde{c} y \tilde{A} se han definido en (9). Este problema ya fue denotado con anterioridad como (\tilde{P}) . Sabemos que con este escalado el punto x^k en el problema original queda

transformado en $\tilde{x}^k = e^T = (1, \dots, 1)^T$ en el nuevo espacio. Consideremos ahora la siguiente esfera

$$\tilde{B}(\tilde{x}, \rho) = \{\tilde{x} \mid \|\tilde{x} - e\| \leq \rho, \tilde{A}\tilde{x} = b\}. \quad (16)$$

formada por la intersección de $\|\tilde{x} - e\| \leq \rho$ y el subespacio $\tilde{A}\tilde{x} = b$. Dado que e es factible para (15), tenemos que $\tilde{B}(\tilde{x}, \rho)$ estará incluido dentro de la región factible $\{\tilde{A}\tilde{x} = b, \tilde{x} \geq 0\}$ de (15) si $\rho \leq 1$. El conjunto de puntos $\tilde{B}(\tilde{x}, \rho)$ puede ser transformado al espacio de variables original x deshaciendo el escalado mediante (8) y (9):

$$\begin{aligned} B(x, \rho) &= \{x \mid \|(X^k)^{-1}x - e\| \leq \rho, AX^k(X^k)^{-1}x = b\} \\ &= \{x \mid \|(X^k)^{-1}x - e\| \leq \rho, Ax = b\}. \end{aligned} \quad (17)$$

Lo que en el espacio \tilde{x} era una esfera se ha convertido en un elipsoide en el espacio original x , gracias a la matriz de escalado $(X^k)^{-1}$. Además, al igual que $\tilde{B}(\tilde{x}, \rho = 1)$ estaba incluido en la región factible de (15), se verifica que el elipsoide $B(x, \rho = 1)$ también está incluido en la región factible de (P). Esto lo garantiza el siguiente resultado:

Proposición 2.

El conjunto de puntos definidos por el elipsoide $B(x, \rho)$ son factibles para el problema (P) si $\rho \leq 1$. Esto es

$$B(x, \rho \leq 1) \subseteq \{x \mid Ax = b, x \geq 0\}.$$

Si $\rho < 1$, además de factibles son puntos interiores.

Demostración.

Por definición de $B(x, \rho)$ sabemos que todos sus puntos x verifican que $Ax = b$. Debemos ver que además $x \geq 0$. Sabemos que si $x \in B(x, \rho)$ entonces $\|(X^k)^{-1}x - e\| \leq \rho$. Utilizando que para cualquier vector v , $v_i \leq \|v\|$, tenemos

$$\left| \frac{x_i}{x_i^k} - 1 \right| \leq \|(X^k)^{-1}x - e\| \leq \rho \leq 1.$$

Multiplicando la relación anterior por x_i^k podemos escribirla como

$$|x_i - x_i^k| \leq \rho x_i^k \leq x_i^k,$$

obteniendo el resultado deseado: $x_i \geq 0$. Si $\rho < 1$ las relaciones anteriores se verifican como estrictas desigualdades, teniendo que $x_i > 0$ y por tanto interior. ■

La Fig. 8 muestra gráficamente la relación entre la esfera (16) y el elipsoide (17) en sus respectivos espacios para $\rho = 1$. Se observa como el elipsoide es factible para (P), tal y como se ha demostrado en la Proposición 2.

Como los puntos del elipsoide (17) son factibles para (P), el método del escalado afín de paso corto itera sustituyendo las restricciones $x \geq 0$ por las de pertenencia a dicho elipsoide. En el problema escalado, en vez del elipsoide considera una esfera, planteando el siguiente problema de optimización para obtener $\Delta\tilde{x}$.

$$\begin{aligned} \min_{\Delta\tilde{x}} \quad & \tilde{c}^T(\tilde{x}^k + \Delta\tilde{x}) \\ \text{sujeto a} \quad & \tilde{A}(\tilde{x}^k + \Delta\tilde{x}) = b \\ & \|\Delta\tilde{x}\|^2 = \rho^2. \end{aligned} \quad (18)$$

Notemos que la región factible de este problema coincide con la frontera de la esfera $\tilde{B}(\tilde{x}, \rho)$, dado que $\tilde{x}^k = e$. Si consideramos $\rho \leq 1$ dicha región está además incluida en la región

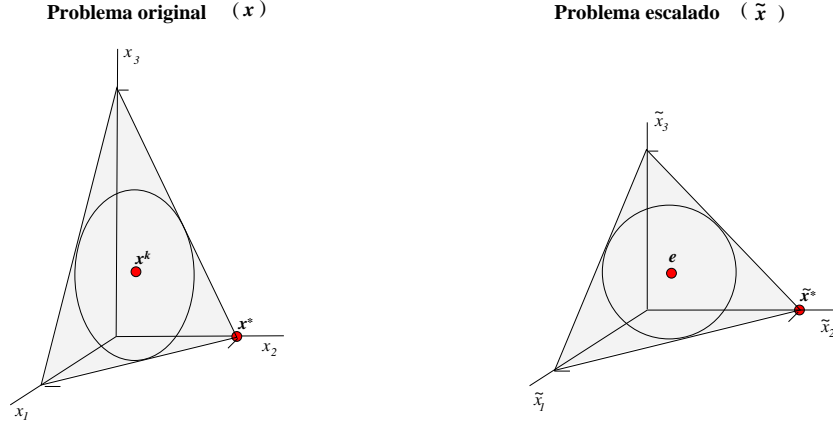


Figura 8. Representación de la esfera $\tilde{B}(\tilde{x}, \rho = 1) = \{\tilde{x} \mid \|\tilde{x} - e\| \leq \rho, \tilde{A}\tilde{x} = b\}$ en el problema escalado y el elipsoide $B(x, \rho = 1) = \{x \mid \|(X^k)^{-1}x - e\| \leq \rho, Ax = b\}$ en el problema original. Se considera un problema con una única restricción y tres variables. Se observa como los puntos del elipsoide son factibles para el problema original.

factible de (15), como hemos argumentado anteriormente. Podríamos haber escrito la segunda restricción como $\|\Delta\tilde{x}\|^2 \leq \rho^2$, teniendo entonces exactamente como región factible de (18) la esfera $\tilde{B}(\tilde{x}, \rho)$. Sin embargo el óptimo del problema se hallará en la frontera, dado que la función objetivo es lineal, por lo que preferimos formular directamente la segunda restricción en forma de igualdad. En este problema estamos buscando la dirección $\Delta\tilde{x}$ que nos proporciona la máxima disminución de nuestra función objetivo en un radio de longitud ρ alrededor de $\tilde{x}^k = e$. Puede verse que es equivalente al problema (5) que planteamos en el apartado 5.2.3. Y sabemos por (7) que su solución viene dada por

$$\Delta\tilde{x} = -\rho \frac{\tilde{P}\tilde{c}}{\|\tilde{P}\tilde{c}\|}.$$

El problema (18) podría haber sido planteado en el espacio de variables x directamente, deshaciendo el escalado, obteniendo entonces:

$$\begin{aligned} \min_{\Delta x} \quad & c^T(x^k + \Delta x) \\ \text{sujeto a} \quad & A(x^k + \Delta x) = b \\ & \|(X^k)^{-1}\Delta x\|^2 = \rho^2. \end{aligned} \tag{19}$$

La región factible corresponde ahora a la frontera del elipsoide $B(x, \rho)$, y sabemos por la Proposición 2 que este elipsoide está incluido en la región factible del problema original (P) si $\rho \leq 1$, y que es interior a ésta si $\rho < 1$. La solución de (19) se obtiene a partir de la de (18), deshaciendo el escalado:

$$\Delta x = X^k \Delta\tilde{x} = -\rho \frac{X^k \tilde{P}\tilde{c}}{\|\tilde{P}\tilde{c}\|}.$$

El nuevo punto será entonces

$$x^{k+1} = x^k - \rho \frac{X^k \tilde{P}\tilde{c}}{\|\tilde{P}\tilde{c}\|}.$$

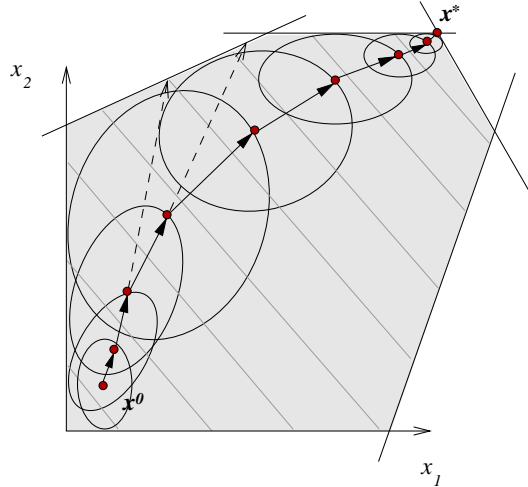


Figura 9. Secuencia de puntos generados por el algoritmo del escalado afín de paso corto. Se muestran los diferentes elipsoides considerados a cada iteración, y como el nuevo punto se halla sobre la frontera de los mismos. Con línea discontinua se muestra, en dos casos, el movimiento que realizaría la variante de paso largo, avanzando hasta encontrar alguna de las paredes de la región factible.

Para garantizar que sea interior, únicamente es necesario utilizar un $\rho < 1$. En (10) vimos como desarrollar el término $X^k \tilde{P} \tilde{c}$.

Observemos que en el algoritmo del escalado afín de paso corto no se realiza el cálculo de α , puesto que la longitud de paso viene determinada por $\rho / \|\tilde{P} \tilde{c}\|$ y nos conduce hasta la frontera del elipsoide. El término ρ juega un papel similar al mismo término del algoritmo de paso largo: nos acorta el paso realizado, aquí alejándonos de la frontera del elipsoide, y en el algoritmo de paso largo de alguna pared definida por $x \geq 0$. La Fig. 9 muestra la secuencia de puntos que generaría el método para un problema concreto. Los elipsoides se adaptan, según el escalado, a la forma de la región factible. Naturalmente, si en vez de quedarnos en la frontera del elipsoide continuamos avanzando hasta alcanzar alguna pared definida por $x \geq 0$, tal y como se realiza en la variante de paso largo, disminuiríamos nuestra función objetivo.

5.6. Detección del óptimo (finalización del algoritmo)

Sabemos como calcular Δx y α , pudiendo generar la sucesión de puntos $\{x^k\}$ a través del proceso iterativo (1). Ahora debemos establecer un criterio para determinar cuando el punto actual x^k está lo suficientemente próximo a x^* . Para ello utilizaremos el Teorema 2 (de la Dualidad Fuerte). Si somos capaces de obtener un estimador de la solución dual y asociada con x^k , entonces podremos detener nuestro proceso iterativo cuando el *gap dual* esté por debajo de una determinada tolerancia ε . Nuestra condición para detener el algoritmo será entonces:

$$\frac{|gap\ dual^k|}{1 + |c^T x^k|} = \frac{|c^T x^k - b^T y|}{1 + |c^T x^k|} \leq \varepsilon. \quad (20)$$

El denominador en la expresión anterior se incluye para considerar el *gap dual* relativo al valor de la función objetivo, y se le suma 1 para evitar problemas de precisión cuando el coste óptimo sea cercano a 0. En la práctica se utilizan valores $\varepsilon \in [10^{-6}, 10^{-8}]$.

Ahora debemos obtener un estimador razonable de y . Y para ello utilizaremos el Teorema 3 (de la Holgura Complementaria). Sabemos que las holguras duales z verifican $z = c - A^T y$, $z \geq$

0, y que en el óptimo de los problema primal y dual $x_i z_i = 0$, $i = 1, \dots, n$. Dejando a parte la restricción $z \geq 0$, podemos estimar y mediante el valor proporcionado por el siguiente problema

$$\begin{aligned} \min_{z,y} \quad & \frac{1}{2} \|X^k z\|^2 \\ \text{sujeto a} \quad & z = c - A^T y. \end{aligned}$$

Introduciendo la función Lagrangiana

$$L(z, y, u) = \frac{1}{2} \|X^k z\|^2 - u^T (z - c + A^T y),$$

donde los multiplicadores $u \in \mathbb{R}^n$, e igualando a 0 sus derivadas respecto z , y y u , obtenemos las condiciones que debe satisfacer el punto solución:

$$\begin{aligned} (X^k)^2 z - u &= 0 \\ Au &= 0 \\ z - c + A^T y &= 0. \end{aligned}$$

Multiplicando la primera ecuación por A , y utilizando que $Au = 0$, obtenemos

$$A(X^k)^2 z = Au = 0.$$

Multiplicando la tercera ecuación por $A(X^k)^2$ y utilizando el resultado anterior obtenemos

$$\left. \begin{aligned} 0 &= A(X^k)^2 z - A(X^k)^2 c + A(X^k)^2 A^T y \\ &= -A(X^k)^2 c + A(X^k)^2 A^T y \end{aligned} \right\} \Rightarrow y = (A(X^k)^2 A^T)^{-1} A(X^k)^2 c. \quad (21)$$

Esta será nuestra estimación de y . Análogamente, el vector $z = c - A^T y$ correspondería a la estimación de las holguras duales.

Haciendo uso de la notación $D = (X^k)^2$ introducida en apartados anteriores, la estimación y puede escribirse como

$$y = (ADA^T)^{-1} ADc.$$

Observemos que coincide con el vector y definido en (12), que interviene en el cálculo de la dirección de movimiento Δx . Por tanto no es necesario hacer ninguna operación adicional para obtener dicho estimador. También es interesante notar que la expresión (21) coincide con el vector de multiplicadores (6) de los problemas de optimización (5), (18) y (19), que nos proporcionaban la dirección de movimiento a cada x^k . Por tanto, nuestra secuencia de estimaciones duales no es más que la secuencia de multiplicadores de las ecuaciones $Ax = b$ de estos problemas.

En el ejercicio 6 se pide comprobar que esta estimación de las variables duales equivale a las del método del símplex a medida que nos acercamos a un punto extremo de la región factible (en este caso, sin embargo, nos acercamos por puntos interiores, en vez de por la frontera como en el símplex).

Dado que consideramos que el problema no es degenerado, la Proposición 1 nos garantiza que $A(X^k)^2 A^T$ es no singular, con lo cual el cálculo de los estimadores duales y está bien definido. Si, por contra, el problema fuese degenerado, podemos hallarnos con una matriz $(A(X^k)^2 A^T)$ casi singular, y si el punto solución x^* es degenerado, esta situación irá a peor a medida que iteremos (en la subsección 5.9 se presenta muy brevemente detalles sobre la convergencia del método y se comentan los problemas derivados por la degeneración). En este caso las estimaciones de y obtenidas por (21) no serán buenas (pueden no converger al y^* de (D)), y nuestro criterio de parada (20) poco acertado. Un criterio alternativo que puede

usarse en estas situaciones consiste en parar cuando la mejora relativa en la función objetivo sea pequeña, esto es

$$\frac{|c^T x^k - c^T x^{k+1}|}{1 + |c^T x^k|} \leq \varepsilon. \quad (22)$$

5.7. Obtención de una solución inicial factible

Sabemos como iterar y como finalizar el método. Sin embargo hemos supuesto que partíamos de un punto inicial x^0 factible e interior. Hallar tal punto para (P) no es trivial. A continuación presentamos un método para solventar esta dificultad, denominado *método Big-M* (equivalente a la técnica del mismo nombre para el algoritmo del simplex).

Consideremos un punto $x^0 > 0$ cualquiera (por ejemplo, $x^0 = (1, \dots, 1)^T$). Este punto escogido al azar no verificará probablemente $Ax^0 = b$. Podemos calcular su vector de infactibilidades como

$$r = b - Ax^0 \quad r \in \mathbb{R}^m.$$

Ahora, utilizando un valor $M \in \mathbb{R}$, donde $M \gg 0$, definamos el siguiente problema alternativo a (P)

$$\begin{aligned} \min_x \quad & [c^T \ M] \begin{bmatrix} x \\ x_{n+1} \end{bmatrix} \\ \text{sujeto a} \quad & [A \ r] \begin{bmatrix} x \\ x_{n+1} \end{bmatrix} = b \\ & \begin{bmatrix} x \\ x_{n+1} \end{bmatrix} \geq 0. \end{aligned} \quad (23)$$

Este nuevo problema tiene una variable más que (P) , y gracias a ello podemos obtener un punto factible e interior de forma sencilla. Este punto será

$$\begin{pmatrix} x_1^0 \\ x_2^0 \\ \vdots \\ x_n^0 \\ 1 \end{pmatrix}. \quad (24)$$

Por ser x^0 interior, (24) también lo es. Y utilizando la definición del vector r comprobamos su factibilidad para (23):

$$[A \ r] \begin{bmatrix} x^0 \\ 1 \end{bmatrix} = Ax^0 + r = Ax^0 + (b - Ax^0) = b.$$

Como M es un valor grande, es de esperar que en el punto óptimo de (23) se verifique $x_{n+1}^* = 0$. Cuando esto no sea así concluiremos que el problema original (P) es infactible, puesto que no hemos podido eliminar la componente asociada con el vector de infactibilidades r . Los principales inconvenientes de este método son:

- Debemos obtener un valor M adecuado, ni demasiado grande, ni demasiado pequeño. Si es muy grande podemos introducir inestabilidad numérica en los cálculos. Si es muy pequeño, penalizaremos poco la variable asociada al vector r , con lo que podemos obtener soluciones con $x_{n+1}^* \neq 0$ sin ser el problema infactible. En la literatura se han propuesto técnicas de adaptación de M a medida que avanza el algoritmo.
- La columna r introducida en la matriz de restricciones de (23) por lo general será densa. Esto provoca que la matriz ADA^T sea también densa. La factorización de ADA^T a cada

iteración del algoritmo es el paso más costoso a realizar, desde un punto de vista computacional. El hecho que esta matriz sea densa incrementa considerablemente el tiempo de ejecución, especialmente en la resolución de problemas de gran dimensión.

Entre sus ventajas destacamos que proporciona el óptimo solucionando un único problema. En esto se diferencia de métodos alternativos que realizan la optimización en dos fases.

5.8. Algoritmo del escalado afín

Estamos ahora en disposición de presentar todas las etapas del algoritmo del escalado afín primal. Éstas se presentan en el Algoritmo 1, el cual no requiere prácticamente comentarios. En la zona de inicializaciones se amplía el problema según el método Big-M. El punto inicial de iteración utilizado —correspondiente a (24)— tiene todas las componentes a 1 (línea 4 del algoritmo). En las inicializaciones también se obtiene la estimación dual de x^0 (línea 5). En la zona iterativa se comprueba en primer lugar la condición de finalización del algoritmo (línea 7), para pasar a calcular el nuevo punto (líneas 8–12). A continuación se estiman las variables duales para el nuevo punto (líneas 14–16), y se vuelve a iterar. Al finalizar el proceso iterativo se comprueba si el problema es infactible o si hemos hallado su óptimo, según el valor de la componente de x asociada con la columna de infactibilidades. En el ejercicio 7 se pide programar el Algoritmo 1 en un lenguaje de cálculo matricial como Matlab.

Veamos un ejemplo de aplicación del Algoritmo 1 a la resolución de un problema lineal de pequeña dimensión.

EJEMPLO 6

Solucionemos el problema lineal

$$\begin{aligned} \min \quad & -3x_1 - 2x_2 \\ \text{sujeto a} \quad & \\ & 4x_1 - 2x_2 + x_3 = 5 \\ & 3x_1 + 4x_2 - x_4 = 1 \\ & x_1 + x_2 + x_5 = 2 \\ & x_i \geq 0 \text{ para toda } i = 1, \dots, 5 \end{aligned}$$

aplicando el algoritmo del escalado afín primal. Este problema ya fue utilizado en los ejemplos 1 y 4, y sabemos que $x^0 = (1/2 \ 1/2 \ 4 \ 5/2 \ 1)^T$ es un punto interior factible. Usaremos este punto evitando tener que aplicar el método Big-M. Utilizaremos un valor de $\rho = 0.95$.

La primera iteración del algoritmo sería como sigue:

- La función objetivo en el punto actual es $c^T x^0 = -2.5$.
- Calculamos y solucionando $(ADA^T)y = ADc$, donde $D = (X^k)^2$, $X^k = \text{diag}(0.5, 0.5, 4, 2.5, 1)$.

$$\begin{pmatrix} 21 & 1 & 0.5 \\ 1 & 12.5 & 1.75 \\ 0.5 & 1.75 & 1.5 \end{pmatrix} y = \begin{pmatrix} -2 \\ -4.25 \\ -1.25 \end{pmatrix} \implies y = \begin{pmatrix} -0.070718 \\ -0.264115 \\ -0.501627 \end{pmatrix}.$$

- Comprobamos que el punto actual no es óptimo.

$$\text{gap dual} = \frac{|c^T x^0 - b^T y|}{1 + |c^T x^0|} = |-2.5 - (-1.620)|/3.5 = 0.25114.$$

- Calculamos $z = c - A^T y$:

$$z = c - A^T y = \begin{pmatrix} -3 \\ -2 \\ 0 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} -1.576842 \\ -1.416651 \\ -0.070718 \\ 0.264115 \\ -0.501627 \end{pmatrix} = \begin{pmatrix} -1.423158 \\ -0.583349 \\ 0.070718 \\ -0.264115 \\ 0.501627 \end{pmatrix}.$$

Algoritmo 1. Algoritmo del escalado afín

INICIALIZACIONES	
1	Calcular infactibilidades: $r = b - Ae_n$, donde $e_l = (1_1, \dots, 1_l)^T$
2	Ampliar A con la columna adicional r : $A \leftarrow [A \ r]$, $n \leftarrow n + 1$
3	Ampliar vector de costes: $c \leftarrow [c^T \ M]^T$, $M \in \mathbb{R}$ grande
4	$x^0 = e_{n+1}$; x^0 es interior ($x^0 > 0$) y factible ($Ax^0 = b$)
5	$D = I$, $y = (ADA^T)^{-1}ADc$
6	$k = 0$, $\varepsilon = 10^{-6}$, $\rho \in [0.95, 0.9995]$
PROCESO ITERATIVO	
7	<u>mientras</u> $\frac{ c^T x^k - b^T y }{1 + c^T x^k } > \varepsilon$ <u>hacer</u>
8	Calcular z : $z = c - A^T y$
9	Calcular Δx : $\Delta x = -Dz$
10	<u>si</u> ($\Delta x \geq 0$) <u>entonces</u> FIN: Problema ilimitado
11	Calcular α : $\alpha = \rho \cdot \min \left\{ -\frac{x_i^k}{\Delta x_i} \ \forall i \ \Delta x_i \leq 0 \right\}$
12	Calcular x^{k+1} : $x^{k+1} = x^k + \alpha \Delta x$
13	$k \leftarrow k + 1$
14	Calcular X^k : $X^k = \text{diag}(x_1^k, \dots, x_n^k)$
15	Calcular D : $D = (X^k)^2$
16	Calcular y : $(ADA^T)y = ADc$
17	<u>fin_mientras</u>
FINALIZACIÓN	
18	<u>si</u> $x_{n+1}^k \neq 0$ <u>entonces</u>
19	FIN: Problema infactible
20	<u>sino</u>
21	FIN: Óptimo hallado: $x^* \leftarrow (x_1^k, \dots, x_n^k)^T$
22	<u>fin_si</u>

- Calculamos $\Delta x = -Dz$:

$$\Delta x = -Dz = \begin{pmatrix} 0.35579 \\ 0.14584 \\ -1.13148 \\ 1.65072 \\ -0.50163 \end{pmatrix}.$$

- Determinamos α :

$$\begin{aligned} \alpha &= \rho \cdot \min\{-x_i^0/\Delta x_i \ \forall i \ \text{tal que } \Delta x_i < 0\} \\ &= 0.95 \cdot \min\{4/1.13148, 1.0/0.50163\} = 0.95 \cdot 1.9935. \end{aligned}$$

- Finalmente obtenemos el nuevo punto $x^1 = x^0 + \alpha \Delta x$:

$$x^1 = \begin{pmatrix} 0.5 \\ 0.5 \\ 4 \\ 2.5 \\ 1 \end{pmatrix} + 0.95 \cdot 1.9935 \begin{pmatrix} 0.35579 \\ 0.14584 \\ -1.13148 \\ 1.65072 \\ -0.50163 \end{pmatrix} = \begin{pmatrix} 1.173803 \\ 0.776190 \\ 1.857169 \\ 5.626170 \\ 0.050007 \end{pmatrix}.$$

- La función objetivo en este nuevo punto es $c^T x_1 = -5.0738 < c^T x^0 = -2.5$.
- El vector de estimadores duales y en el nuevo punto x^1 viene dado por

$$y = (-0.1361242 \quad 0.0017763 \quad -2.4025868)^T.$$

- El gap dual continúa siendo elevado

$$\frac{|c^T x^1 - b^T y|}{1 + |c^T x^1|} = 0.067538,$$

por lo que deberíamos seguir iterando.

La secuencia de puntos obtenida, junto con el gap dual, es la siguiente

k	x_1^k	x_2^k	x_3^k	x_4^k	x_5^k	gap dual
0	0.5	0.5	4	2.5	1	0.25116
1	1.173808	0.776192	1.857154	5.626192	0.050000	0.06753
2	1.475381	0.497191	0.092858	5.414905	0.027429	0.01181
3	1.487639	0.510988	0.071418	5.506874	0.001371	0.00235
4	1.499064	0.499914	0.003570	5.496851	0.001020	0.00045
5	1.4995	0.50042	0.002723	5.500300	0.000051	$8.82 \cdot 10^{-5}$
6	1.5000	0.50000	0.000103	5.500000	0.000038	$1.72 \cdot 10^{-5}$

y su representación gráfica se muestra en la Fig. 10. En ésta, las tres últimas variables se toman como holguras, considerando las restricciones del problema como desigualdades.

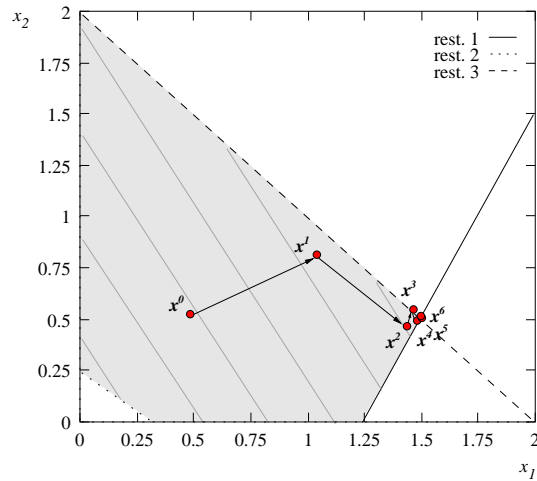


Figura 10. Representación gráfica de la secuencia de puntos $\{x^k\}$ obtenidos al solucionar el problema del ejemplo 6.

5.9. Convergencia del algoritmo

El análisis de convergencia del algoritmo del escalado afín es más complejo que el de otros métodos de punto interior más “elaborados” y eficientes, contrariamente a lo que se podría suponer. Aquí únicamente demostraremos su convergencia para el caso particular que estamos considerando, donde los problemas primal y dual son no degenerados (Hipótesis 2 y 3). En las Proposiciones 3 y 4 se presenta dicha demostración de convergencia. El resto de resultados únicamente serán presentados, indicando donde pueden hallarse sus respectivas demostraciones.

En primer lugar vamos a mostrar que si la secuencia de puntos x^k , y^k y z^k converge, lo hace a los puntos óptimos de (P) y (D) (los términos y^k y z^k son los estimadores de las variables duales definidos en (12), añadiendo ahora por conveniencia el supraíndice k de la iteración).

Proposición 3.

Dado un problema no ilimitado (P) y su dual (D), si la secuencia de puntos x^k , y^k y z^k generada por el método del escalado afín converge hacia x^* , y^* y z^* , entonces x^* es óptimo para (P), y el par (y^*, z^*) es óptimo para (D).

Demostración.

Consideremos dos puntos x^k y x^{k+1} de la secuencia. Como ambos puntos son factibles tenemos que $A(x^k - x^{k+1}) = b - b = 0$. Utilizando este hecho y las definiciones de z^k en (12), Δx en (11), y del proceso iterativo (1), la disminución de coste que se obtiene al pasar de x^k a x^{k+1} puede escribirse como

$$\begin{aligned} \Delta \text{coste} &= c^T(x^k - x^{k+1}) = c^T(x^k - x^{k+1}) - (y^k)^T A(x^k - x^{k+1}) \\ &= (c - A^T y^k)^T(x^k - x^{k+1}) = (z^k)^T(x^k - x^{k+1}) \\ &= -(z^k)^T \rho \bar{\alpha} \Delta x = (z^k)^T \rho \bar{\alpha} (X^k)^2 z^k \\ &= \rho \bar{\alpha} \|X^k z^k\|^2. \end{aligned} \tag{25}$$

Por la definición de $\bar{\alpha}$ en (13), y de Δx en (11), tenemos

$$\begin{aligned} \bar{\alpha} &= \min \left\{ -\frac{x_i^k}{\Delta x_i} \text{ para toda } i \text{ tal que } \Delta x_i \leq 0 \right\} = \min \left\{ \frac{1}{x_i^k z_i^k} \text{ para toda } i \text{ tal que } z_i^k \geq 0 \right\} \\ &= \frac{1}{\max \{x_i^k z_i^k \text{ para toda } i \text{ tal que } z_i^k \geq 0\}}. \end{aligned} \tag{26}$$

Utilizando que para cualquier vector v , $v_i \leq \|v\|$, podemos acotar inferiormente $\bar{\alpha}$:

$$\bar{\alpha} = \frac{1}{\max \{x_i^k z_i^k \text{ para toda } i \text{ tal que } z_i^k \geq 0\}} \geq \frac{1}{\|X^k z^k\|}.$$

Sustituyendo esta última expresión en (25) se tiene

$$\Delta \text{coste} = \rho \bar{\alpha} \|X^k z^k\|^2 \geq \rho \frac{\|X^k z^k\|^2}{\|X^k z^k\|} = \rho \|X^k z^k\|.$$

Sabemos que la sucesión de costes $c^T x^k$ es decreciente y que está acotada inferiormente, por ser (P) ilimitado. Por tanto es una sucesión convergente, verificándose que

$$\lim_{k \rightarrow \infty} \Delta \text{coste} = \lim_{k \rightarrow \infty} c^T(x^k - x^{k+1}) = 0.$$

Combinando las dos expresiones anteriores se tiene que

$$\lim_{k \rightarrow \infty} \rho \|X^k z^k\| = 0 \Rightarrow \lim_{k \rightarrow \infty} x_i^k z_i^k = 0 \text{ para todo } i = 1, \dots, n. \tag{27}$$

Como x^k converge hacia x^* y z^k hacia z^* tenemos que

$$x_i^* z_i^* = 0 \text{ para todo } i = 1, \dots, n. \tag{28}$$

Ademas, por ser x^k una secuencia de puntos factibles,

$$Ax^* = b. \tag{29}$$

Comprobemos ahora que $z^* \geq 0$. Supongamos que existe un $z_i^* < 0$. Por tanto existe un índice K tal que para todo $k > K$ se verifica $z_i^k < 0$. Escribiendo el proceso iterativo que genera la secuencia de puntos x^k para la componente i -ésima como

$$x_i^{k+1} = x_i^k - \alpha(x_i^k)^2 z_i^k,$$

y considerando un $k > K$, concluimos que $x_i^{k+1} > x_i^k > 0$ (ya que $\alpha > 0$, $x_i^k > 0$, y hemos supuesto que $z_i^k < 0$). Por tanto $x_i^* z_i^* < 0$ contradiciendo (28). Concluimos entonces que

$$z^* \geq 0. \quad (30)$$

Como $z^* = c - A^T y^*$ (por (12)), por las condiciones (28), (29) y (30), y el Teorema 3 de la Holgura Complementaria, se concluye directamente que x^* y el par (y^*, z^*) son los óptimos de (P) y (D) respectivamente. ■

El resultado anterior supone que las secuencias x^k , y^k y z^k convergen. La siguiente Proposición demuestra que, si se verifican las Hipótesis 2 y 3 de no-degeneración de los problemas primal y dual, las secuencias anteriores son, en efecto, convergentes.

Proposición 4.

Si los problemas (P) y (D) no son degenerados, y (P) no es ilimitado, entonces para todo $\rho < 1$ la secuencias x^k , y^k generadas por el método del escalado afín son convergentes.

Demostración.

La demostración consta de tres partes. En la primera se muestra que la secuencia x^k está acotada. En la segunda, que cualquier subsecuencia de x^k converge hacia una solución básica factible. En la tercera se demuestra que la solución básica factible a la cual converge cualquier subsecuencia es única, garantizándose entonces que x^k (y a su vez también y^k) es convergente.

En primer lugar mostraremos que la secuencia x^k está acotada. Dado que (P) no es ilimitado y que, al igual que (D), es no degenerado, tiene una única solución óptima. Sea x^* dicha solución. Consideremos ahora el siguiente problema:

$$\begin{aligned} \max_x \quad & \sum_{i=1}^n x_i \\ \text{sujeto a} \quad & Ax = b \\ & c^T x \leq c^T x^* \\ & x \geq 0. \end{aligned}$$

El problema anterior tiene una solución óptima finita, puesto que x^* es el único punto factible. Esto implica que el siguiente problema también tiene una solución óptima finita:

$$\begin{aligned} \max_x \quad & \sum_{i=1}^n x_i \\ \text{sujeto a} \quad & Ax = b \\ & c^T x \leq c^T x^0 \\ & x \geq 0, \end{aligned}$$

donde x^0 es el primer punto generado por el algoritmo del escalado afín (la implicación anterior puede comprobarse intuitivamente utilizando propiedades de los polítopos convexos y observando que en el segundo problema únicamente hemos desplazado el hiperplano de la restricción $c^T x \leq c^T x^*$, de forma que si el primer problema no era ilimitado, el segundo tampoco puede

serlo; una demostración formal de esta afirmación puede hallarse en el Teorema 4.14 del texto Bertsimas y Tsitsiklis (1997)). Por tanto el conjunto de puntos $\{x|Ax = b, c^T x \leq c^T x^0, x \geq 0\}$ está acotado (de lo contrario el máximo de $\sum_{i=1}^n x_i$ no sería finito). Dado que la secuencia de puntos x^k generada por el algoritmo del escalado afín pertenece al conjunto anterior (ya que todos ellos son factibles y verifican $c^T x^k \leq c^T x^0$), concluimos que es una secuencia acotada.

Por ser x^k una secuencia acotada, y utilizando un resultado básico de análisis matemático, podemos concluir que existe una subsecuencia x^{k_j} que converge a un punto límite \bar{x} . Vamos a comprobar que este punto límite es una solución básica factible. Consideremos el conjunto de índices $\mathcal{B} = \{i|\bar{x}_i > 0\}$. Como (P) es no degenerado, la cardinalidad de \mathcal{B} es mayor que m , el número de restricciones del problema. El Teorema fundamental de la Programación Lineal nos garantiza que si existe una solución factible, existe también una solución básica factible, y las componentes de la solución básica factible son un subconjunto de las componentes positivas de la solución factible. Por tanto podemos escoger un conjunto de m índices $B_1, \dots, B_m \in \mathcal{B}$, tal que las columnas A_{B_1}, \dots, A_{B_m} son linealmente independientes y la solución básica asociada \hat{x} es factible, y además no degenerada. Sea $B = [A_{B_1} | \dots | A_{B_m}]$ la base asociada y $c_B = (c_{B_1}, \dots, c_{B_m})$. Usando la definición de z^k , $z^k = c - A^T y^k$, y la ecuación (27) de la Proposición 3, tenemos que $\lim_{j \rightarrow \infty} x_i^{k_j} (c_i - A_i^T y^{k_j}) = 0$ para toda componente i . Las componentes $x_{B_i}^{k_j}$, $i = 1 \dots m$, convergen hacia \bar{x}_{B_i} y son positivas porque $B_i \in \mathcal{B}$, $i = 1 \dots m$. Por tanto $c_B - B y^{k_j}$ debe converger hacia 0 para garantizar (27). Puesto que B es no singular, se tiene que y^{k_j} converge hacia $B^{-1} c_B$. Por tanto, para toda componente i , $c_i - A_i^T y^{k_j}$ converge hacia $c_i - A_i^T B^{-1} c_B$, valor que coincide con el coste reducido de la variable i -ésima en la solución básica \hat{x} . La no-degeneración del problema dual (D) nos garantiza que $c_i - A_i^T B^{-1} c_B \neq 0$ para toda variable i no básica ($i \notin \{B_1, \dots, B_m\}$). Por tanto, para las variables no básicas $i \notin \{B_1, \dots, B_m\}$ se tiene que $c_i - A_i^T y^{k_j}$ converge a un valor diferente de 0, por lo que, usando una vez más (27), tenemos que $\bar{x}_i = 0$ para $i \notin \{B_1, \dots, B_m\}$. Por tanto concluimos que el conjunto de índices de las variables con valor estrictamente positivo es $\mathcal{B} = \{B_1, \dots, B_m\}$, y que $\bar{x} = \hat{x}$, garantizando que el punto límite \bar{x} es una solución básica factible.

Hemos comprobado que x^k tiene como mínimo un punto límite, y que cada punto límite es una solución básica factible. Ahora comprobaremos que éste es el único punto límite que existe. Sea $\delta > 0$ un valor real tal que cualquier variable básica en cualquier solución básica factible es mayor o igual que δ . Este valor δ siempre existirá ya que, por la no-degeneración de (P) , las variables básicas son estrictamente positivas. Sea $\epsilon = \delta/3$. Dado que cualquier punto límite de x^k es una solución básica factible, existe un valor K tal que para todo $k \geq K$, x^k se halla a una distancia inferior a ϵ de una determinada solución básica \tilde{x} (esto es, $\|x^k - \tilde{x}\| \leq \epsilon$, $k \geq K$). Supongamos ahora que existen dos soluciones básicas diferentes \bar{x} y \tilde{x} tales que para $k \geq K$,

$$\|x^k - \bar{x}\| \leq \epsilon \quad \text{y} \quad \|x^{k+1} - \tilde{x}\| \leq \epsilon. \quad (31)$$

Consideremos que la variable i es no básica en \bar{x} ($\bar{x}_i = 0$) y básica en \tilde{x} ($\tilde{x}_i \geq \delta$). Sustituyendo $\bar{x}_i = 0$ en la primera expresión de (31) (y usando que $v_i \leq |v_i| \leq \|v\|$ para cualquier vector v) tenemos que

$$x_i^k \leq \epsilon. \quad (32)$$

Análogamente, sustituyendo $\tilde{x}_i \geq \delta$ en la segunda expresión de (31), obtenemos que

$$|x_i^{k+1} - \tilde{x}_i| \leq \epsilon \Rightarrow -\epsilon \leq x_i^{k+1} - \tilde{x}_i \leq \epsilon \Rightarrow x_i^{k+1} \geq \tilde{x}_i - \epsilon \geq \delta - \epsilon = 2\epsilon. \quad (33)$$

Utilizando la ecuación de cálculo del nuevo iterado (1), la definición de Δx en (11), y la expresión de la longitud de paso máximo (26), obtenemos

$$x_i^{k+1} = x_i^k + \rho \bar{\alpha} (-(x_i^k)^2 z_i^k) = x_i^k \left(1 - \rho \frac{1}{\max \{x_i^k z_i^k \text{ para toda } i \text{ tal que } z_i^k \geq 0\}} x_i^k z_i^k\right). \quad (34)$$

Usando un argumento similar al utilizado al final de la Proposición 3 se puede afirmar que $z_i^k \geq 0$ (de otra forma no se garantizaría que $\bar{x}_i = 0$). A partir de (34), sabiendo por las hipótesis que $\rho < 1$, y utilizando (32) y (33) obtenemos finalmente

$$x_i^{k+1} \leq x_i^k(1 + \rho) < 2x_i^k \leq 2\epsilon \leq x_i^{k+1},$$

llegando a una contradicción. Por tanto sólo existe un único punto límite \bar{x} y la secuencia x^k converge a él. Tal y como se ha mostrado anteriormente, y^k converge también a $B^{-1}c_B$, la solución dual asociada a \bar{x} . ■

Sin las hipótesis de no-degeneración, también puede demostrarse la convergencia del algoritmo, aunque en este caso nos vemos obligados a reducir más la longitud de paso a través de la disminución de ρ . El siguiente resultado puede hallarse demostrado en Tsuchiya (1996):

Proposición 5.

Si $\rho \leq 2/3$, y (P) no es ilimitado, la secuencias x^k, y^k generadas por el método del escalado afín son convergentes (independientemente de la degeneración o no de los problemas primal y dual).

A continuación apuntamos un resultado sobre el orden de convergencia del método. Dado que avanzamos en la dirección de un gradiente proyectado (con un escalado) no debemos esperar un orden de convergencia superior al lineal. El siguiente resultado, de nuevo demostrado en Tsuchiya (1996), ratifica esta afirmación

Proposición 6.

Si $\rho \leq 1$, y (P) no es ilimitado, la secuencia $c^T x^k$ generada por el método del escalado afín converge linealmente hacia $c^T x^$, con una velocidad de convergencia no inferior a $1 - \rho/(2\sqrt{n})$.*

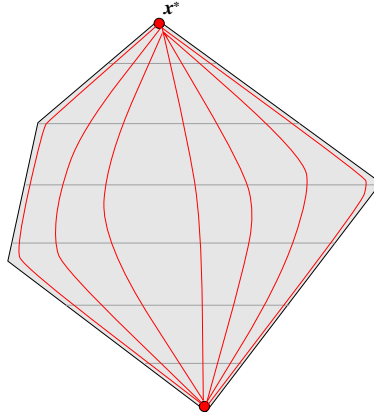


Figura 11. Trayectoria seguida por el método del escalado afín para alcanzar el punto óptimo x^* desde diferentes puntos iniciales, considerando valores $0 < \rho \approx 0$.

Acabamos este apartado indicando que todavía no está resuelto el problema de demostrar si este método de punto interior, al igual que otros, tiene un coste polinómico (esto es, si obtiene una solución en un número de iteraciones que está acotado por un polinomio función del número de variables n). Existe la sospecha, sin embargo, de que esto no es así. Tal argumento se basa en el hecho que, para valores de ρ infinitesimalmente pequeños la trayectoria seguida por el algoritmo del escalado afín consistiría en una curva que une el punto inicial x^0 con el punto solución x^* . La Fig. 11 muestra el aspecto típico que tendrían algunas de estas curvas partiendo desde puntos iniciales distintos aunque próximos. La principal característica que tienen estas

trayectorias es que, cuando se acercan a una faceta de la región factible se mantienen tangentes a ellas. Esto significa que cuando una variable x_i tiende hacia 0, su componente en la dirección de movimiento también se hace 0:

$$\lim_{x_i \rightarrow 0} \Delta x_i = 0.$$

Puede comprobarse utilizando la definición de Δx en (11). En estas situaciones la trayectoria realizada por el método del escalado afín es similar a la del algoritmo del símplex, avanzando a través de la frontera de la región factible de vértice en vértice, tal y como muestra la Fig. 11. Esto hace suponer que, al igual que el símplex, el escalado afín puede no tener una complejidad polinómica.

5.10. Rendimiento computacional del algoritmo

Se ha comparado el rendimiento computacional de una implementación del escalado afín con el de una implementación del algoritmo del símplex, en la solución de una batería estándar de 90 problemas de programación lineal, conocida como Netlib (Gay (1985)). La implementación del algoritmo del símplex utilizada ha sido la del paquete Minos (Murtagh y Saunders (1983)). Como implementación del algoritmo del escalado afín se ha tomado la del sistema Lpabo (Park *et al.* (1997)). El sistema Lpabo se distribuye en código fuente, está totalmente escrito en C, y permite al usuario escoger entre diferentes algoritmos de punto interior, lo cual hace de él un código especialmente apropiado para tareas docentes.

La Tabla 1 muestra los resultados obtenidos en la solución de los 90 problemas utilizados. Para cada problema se indica su nombre (columna “Problema”), y el número de restricciones, variables y elementos diferentes de cero de la matriz A (columnas “ m ”, “ n ”, y “ ne ” respectivamente). Para los dos códigos utilizados, Lpabo y Minos, se muestra el número de iteraciones realizado (columnas “N. iter”) y el tiempo de CPU en segundos requerido para la solución del problema (columna “CPU”). Todas las ejecuciones han sido realizadas sobre un ordenador personal con un procesador Pentium a 133MHz, utilizando el sistema operativo Linux.

La Tabla 2 resume la información presentada en la Tabla 1. Para cada código se indica el número total de problemas que pudieron ser solucionados, y se realiza una pequeña estadística descriptiva donde se muestra la media, desviación estándar, valor máximo y valor mínimo del número de iteraciones y tiempo de CPU de los dos códigos. Para obtener estos estadísticos únicamente se han utilizado los 85 problemas que pudieron ser solucionados por los dos códigos. Puede comprobarse como Lpabo no proporcionó la solución óptima en 4 problemas (debido fundamentalmente, al uso del método Big-M), mientras que Minos tuvo un comportamiento más robusto y únicamente en un caso no obtuvo el punto óptimo. También se observa como Lpabo tuvo un mejor comportamiento que Minos en lo que a tiempo de ejecución se refiere. La comparación del número de iteraciones invertido por cada código destaca una de las propiedades de los métodos de punto interior: el número de iteraciones es muy inferior al requerido por el algoritmo del símplex.

Finalmente, en las Figuras 12 y 13 se muestra la evolución del número de iteraciones de los dos algoritmos y de su cociente de tiempos de CPU (tiempo de CPU de Minos dividido por el de Lpabo), según el número de variables de cada problema, respectivamente. En la Fig. 12 se observa claramente una de las propiedades del algoritmo del escalado afín, y en general de los métodos de punto interior: el número de iteraciones realizado no aumenta con la dimensión del problema, y normalmente es un valor relativamente bajo (en pocas ocasiones se superan las 100 iteraciones). En la Fig. 13 se observa como Lpabo fue, en la mayoría de problemas, más eficiente que Minos (en la figura se separan las iteraciones favorables a Lpabo y Minos mediante la línea horizontal situada en el valor de abscisas 1, donde ambos códigos tienen idéntico rendimiento).

Se observa también una ligera tendencia positiva, a favor de Lpabo, en los problemas de mayor dimensión.

Tabla 1. Rendimiento de Lpabo y Minos

Problema	Dimensiones			Lpabo		Minos	
	m	n	ne	N. Iter.	CPU	N. Iter.	CPU
25fv47	822	1571	11127	41	10.6	7901	91.0
80bau3b	2263	9799	29063	59	23.3	12230	267.6
adlittle	57	97	465	25	0.1	145	0.4
afro	28	32	88	19	0.0	15	0.1
agg	489	163	2541	33	1.8	167	2.0
agg2	517	302	4515	38	4.7	247	2.9
agg3	517	302	4531	78	9.1	310	3.1
bandm	306	472	2659	32	0.7	505	3.2
beaconfd	174	262	3476	27	0.2	118	1.4
blend	75	83	521	24	0.1	101	0.4
bnl1	644	1175	6129	51	4.3	1513	12.6
bnl2	2325	3489	16124	52	44.3	7164	151.9
boeing1	351	384	3865	40	1.5	692	3.7
boeing2	167	143	1339	31	0.4	194	0.9
bore3d	234	315	1525	47	0.3	175	1.3
brandy	221	249	2150	26	0.5	354	1.9
capri	272	353	1786	37	0.8	288	1.8
cycle	1904	2857	21322	53	16.5	2818	59.2
czprob	930	3523	14173		—	2001	25.3
d2q06c	2172	5167	35674		—	58928	1078.4
d6cube	416	6184	43888	57	52.3	106649	1288.9
degen2	445	534	4449	31	3.6	1331	8.9
degen3	1504	1818	26230	39	61.5	8040	194.7
df001	6072	12230	41873	158	19668.3	—	
e226	224	282	2767	34	1.1	618	2.7
etamacro	401	688	2489	37	3.6	849	4.7
ffff800	525	854	6235	67	6.0	290	4.1
finnis	498	614	2714	45	2.0	609	4.2
fit1d	25	1026	14430	33	4.5	3125	8.3
fit2d	26	10500	138018	32	41.5	36532	336.2
forplan	162	421	4916	45	2.4	347	2.5
ganges	1310	1681	7021	36	10.9	782	12.9
gfrd-pnc	617	1092	3467	49	1.4	719	7.0
greenbea	2393	5405	31499	147	71.6	31008	902.4
greenbeb	2393	5405	31499	70	34.6	16309	516.1

Tabla 1. (cont.) Rendimiento de Lpabo y Minos

Problema	Dimensiones			Lpabo		Minos	
	m	n	ne	N. Iter.	CPU	N. Iter.	CPU
grow15	301	645	5665	22	1.9	555	6.1
grow22	441	946	8318	24	3.1	895	12.0
grow7	141	301	2633	22	0.9	230	2.0
israel	175	142	2358	76	11.2	328	1.7
kb2	44	41	291	30	0.2	62	0.2
lotfi	154	308	1086	36	0.5	284	1.2
maros	847	1443	10006	51	7.0	2524	28.4
modszk1	688	1620	4158		—	1269	12.2
nesm	663	2923	13988	38	12.8	3319	31.2
perold	626	1376	6026	107	34.5	4807	43.7
pilot	1442	3652	43220	64	444.8	18709	794.1
pilot.ja	941	1988	14706	350	220.2	8531	107.2
pilot.we	723	2789	9218	55	9.6	6357	62.1
pilot4	411	1000	5145	79	14.9	1753	13.2
pilot87	2031	4883	73804	58	1422.7	28105	1876.3
pilotnov	976	2172	13129	42	31.2	2830	39.8
recipe	92	180	752	19	0.2	29	0.5
sc105	106	103	281	20	0.2	44	0.4
sc205	206	203	552	20	0.3	94	0.8
sc50a	51	48	131	21	0.1	35	0.2
sc50b	51	48	119	18	0.1	15	0.2
scagr25	472	500	2029	28	0.8	407	3.3
scagr7	130	140	553	22	0.2	125	0.6
scfxm1	331	457	2612	38	1.5	482	2.8
scfxm2	661	914	5229	42	3.4	890	7.8
scfxm3	991	1371	7846	43	5.7	1369	15.7
scorpion	389	358	1708	25	0.3	153	1.7
scrs8	491	1169	4029	43	2.5	967	7.0
scsd1	78	760	3148	28	0.8	549	2.5
scsd6	148	1350	5666	27	1.4	1315	6.8
scsd8	398	2750	11334	25	2.5	3470	31.9
sctap1	301	480	2052	32	0.9	304	2.0
sctap2	1091	1880	8124	34	4.9	812	10.9
sctap3	1481	2480	10734	35	6.4	1123	18.1
seba	516	1028	4874	34	47.6	397	4.2

Tabla 1. (cont.) Rendimiento de Lpabo y Minos

Problema	Dimensiones			Lpabo		Minos	
	m	n	ne	N. Iter.	CPU	N. Iter.	CPU
share1b	118	225	1182	156	1.7	285	1.0
share2b	97	79	730	24	0.3	132	0.5
shell	537	1775	4900	39	1.9	302	4.8
ship04l	403	2118	8450	34	2.0	388	5.9
ship04s	403	1458	5810	32	1.3	209	4.0
ship08l	779	4283	17085	40	4.1	617	13.5
ship08s	779	2387	9501	35	1.9	331	7.6
ship12l	1152	5427	21597	37	4.9	1212	26.6
ship12s	1152	2763	10941	34	2.0	561	11.9
sierra	1228	2036	9252	82	10.6	1378	18.2
stair	357	467	3857	27	6.1	726	6.4
standata	360	1075	3038	55	2.1	171	2.6
standgub	362	1184	3147	55	2.0	116	2.6
standmps	468	1075	3686	75	3.4	460	4.0
stocfor1	118	111	474	27	0.2	113	0.6
stocfor2	2158	2031	9492	44	8.8	3399	68.3
tuff	334	587	4523	27	2.4	771	4.6
vtp.base	199	203	914	20	0.2	172	0.9
wood1p	245	2594	70216		—	1081	39.3
woodw	1099	8405	37478	47	19.5	5375	118.4

Tabla 2. Resumen de los resultados obtenidos con Lpabo y Minos

	Lpabo		Minos	
problemas solucionados	86		89	
	N. Iter	CPU	N. Iter	CPU
media	46.3	32.8	4126.3	86.3
desviación estándar	41.1	161.9	13009.0	279.5
máximo	350	1422.7	106649	1876.3
mínimo	18	0.04	15	0.12

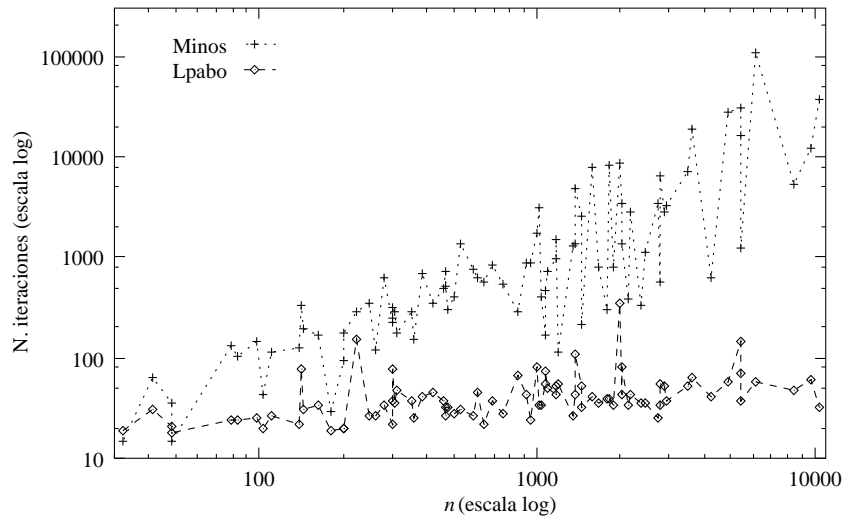


Figura 12. Número de iteraciones requerido por Lpabo y Minos en la solución de los problemas Netlib, según el número de variables de cada problema. Ambos ejes están en escala logarítmica.

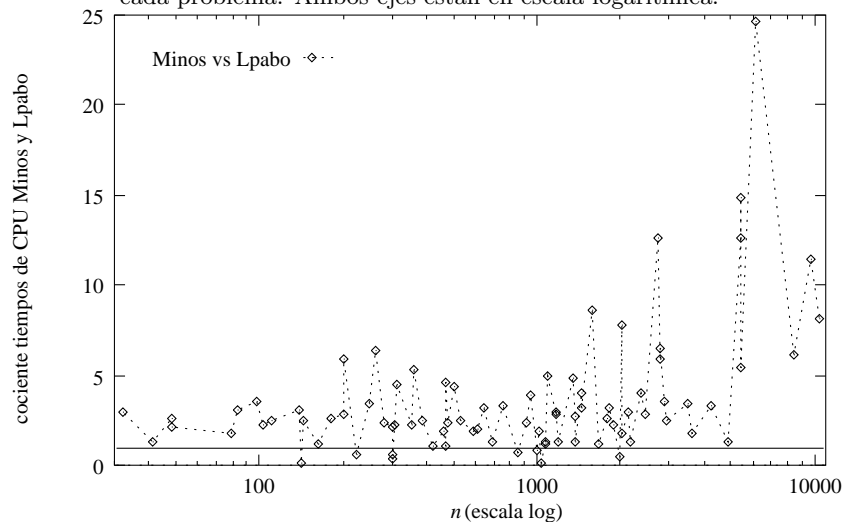


Figura 13. Cociente (tiempo de CPU de Minos)/(tiempo de CPU de Lpabo), según el número de variables de cada problema. El eje horizontal está en escala logarítmica.

6. Ejercicios propuestos

1. Deducid la expresión de la matriz de proyección ortogonal $P = I_n - A^T(AA^T)^{-1}A$ sobre el espacio nulo de $A \in \mathbb{R}^{m \times n}$, que fue definida en (2). Para ello utilizad que un vector cualquiera $d \in \mathbb{R}^n$ puede descomponerse en $d = u + v$, donde u pertenece al espacio nulo de A , y v al espacio de rango, tal y como muestra la Fig. 14.

SOLUCIÓN

Como u y v pertenecen al espacio nulo y de rango de A respectivamente tenemos que

$$Au = 0 \quad A^T y = v,$$

donde $y \in \mathbb{R}^m$. Podemos sustituir el valor anterior de v en $d = u + v$, obteniendo

$$u = d - A^T y.$$

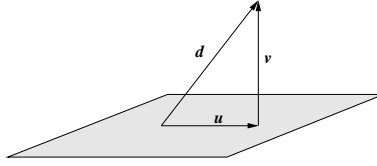


Figura 14. Descomposición del vector d como suma de los vectores ortogonales u y v .

Multiplicando por A en ambos lados de la expresión anterior, considerando que A es de rango completo, y utilizando que $Au = 0$, obtenemos y :

$$0 = Au = Ad - AA^T y \Rightarrow y = (AA^T)^{-1}d.$$

Este valor de y lo sustituimos ahora en la expresión $u = d - A^T y$:

$$u = d - A^T y = d - A^T (AA^T)^{-1}d = (I_n - A^T (AA^T)^{-1}A)d = Pd.$$

Por tanto P nos proporciona la proyección ortogonal u de d sobre el subespacio nulo de A .

Puede realizarse una deducción equivalente, observando la Fig. 14 y planteando el problema de obtención del vector v de norma mínima de forma que u pertenezca al espacio nulo de A , esto es:

$$\begin{aligned} \min \quad & \frac{1}{2} \|v\|^2 = \frac{1}{2} \|d - u\|^2 \\ \text{sujeto a} \quad & Au = 0. \end{aligned}$$

Planteando la función Lagrangiana de este problema

$$L(u, y) = \frac{1}{2}(d - u)^T(d - u) - y^T Au,$$

y derivándola respecto u e y se obtienen las condiciones

$$\begin{aligned} A^T y &= d - u \\ Au &= 0, \end{aligned}$$

que coinciden con las obtenidas por el primer método.

- 2.** Comprobad que la matriz de proyección ortogonal $P = I_n - A^T(AA^T)^{-1}A$ definida en (2) verifica las tres propiedades de (3): $AP = \mathbf{0}$, es simétrica, y es idempotente.

SOLUCIÓN

Para ver que $AP = \mathbf{0}$ basta sustituir la definición de P :

$$AP = A(I_n - A^T(AA^T)^{-1}A) = A - (AA^T)(AA^T)^{-1}A = A - A = \mathbf{0}.$$

La simetría se comprueba aplicando propiedades básicas de cálculo matricial:

$$\begin{aligned} P^T &= (I_n - A^T(AA^T)^{-1}A)^T = I_n^T - (A^T(AA^T)^{-1}A)^T \\ &= I_n - (A^T((AA^T)^{-1})^T(A^T)^T) = I_n - (A^T((AA^T)^T)^{-1}A) \\ &= I_n - A^T(AA^T)^{-1}A = P. \end{aligned}$$

La idempotencia se observa simplemente elevando al cuadrado P :

$$\begin{aligned} P^2 &= (I_n - A^T(AA^T)^{-1}A) \cdot (I_n - A^T(AA^T)^{-1}A) \\ &= I_n - A^T(AA^T)^{-1}A - A^T(AA^T)^{-1}A + A^T(AA^T)^{-1}(AA^T)(AA^T)^{-1}A \\ &= I_n - A^T(AA^T)^{-1}A = P. \end{aligned}$$

3. Supongamos que la dirección de movimiento $\Delta x = -Pc$ introducida en (4) verifica que $\|Pc\|^2 = 0$. Comprobad que en estas circunstancias cualquier punto factible es óptimo.

SOLUCIÓN

Que Pc sea igual a 0 significa que la proyección de c sobre el espacio nulo de A es 0. O lo que es lo mismo, que c pertenece al espacio de rango o imagen de A , por lo que existe un vector $y \in \mathbb{R}^m$ (considerando que $A \in \mathbb{R}^{m \times n}$) tal que

$$c^T = y^T A$$

(esto es, el vector de costes c es combinación lineal de las filas de la matriz de restricciones A). Consideremos ahora un punto factible cualquiera. Postmultiplicando la ecuación anterior por este punto obtenemos

$$c^T x = y^T Ax = y^T b.$$

En la expresión anterior hemos utilizado la factibilidad de x , $Ax = b$. Por tanto, la función objetivo es constante en la región factible, concluyéndose que cualquier punto factible es óptimo.

Puede darse una justificación alternativa observando que y es un punto factible para el problema dual (D) $\max y^T b$ sujeto a $A^T y + z = c^T$, $z \geq 0$, con $z = 0$. Dado un x factible primal cualquiera, hemos visto que $y^T b = c^T x$. Por el Teorema 2 de la Dualidad Fuerte podemos concluir que y y x son óptimos de sus respectivos problemas dual y primal. Como que no hemos impuesto ninguna condición especial sobre x (únicamente que sea factible), se tiene que cualquier x factible es óptimo.

Podemos observar esta situación en el problema

$$\begin{aligned} \min \quad & 3x_1 + 4x_2 \\ \text{sujeto a} \quad & x_1 + x_2 + x_3 = 1 \\ & 2x_1 + 3x_2 - x_3 = 1 \\ & x_1 \geq 0 \quad x_2 \geq 0 \quad x_3 \geq 0, \end{aligned}$$

donde $y = (1 \ 1)^T$. En este caso se puede comprobar que cualquier punto factible verifica que $3x_1 + 4x_2 = 2$.

4. Comprobad que la dirección de movimiento $\Delta x = -D(I_n - A^T(ADA^T)^{-1}AD)c$ definida en (11), donde $D = (X^k)^2$, preserva la factibilidad del nuevo punto y es de descenso, esto es

- 1) $A\Delta x = 0$.
- 2) $c^T \Delta x \leq 0$.

SOLUCIÓN

- 1) Directamente multiplicando comprobamos la factibilidad:

$$\begin{aligned} A\Delta x &= -AD(I_n - A^T(ADA^T)^{-1}AD)c \\ &= -(ADI_n - (ADA^T)(ADA^T)^{-1}AD)c \\ &= -(AD - AD)c = 0. \end{aligned}$$

- 2) Sabiendo que $\Delta x = -X^k \tilde{P} \tilde{c}$, donde \tilde{P} es la matriz de proyección ortogonal sobre \tilde{A} , que \tilde{c} y \tilde{A} se definen como en (9), y que \tilde{P} es idempotente y simétrica (ver el

ejercicio 2), entonces:

$$\begin{aligned}
 c^T \Delta x &= -c^T X^k \tilde{P} \tilde{c} \\
 &= -c^T X^k \tilde{P} X^k c \\
 &= -c^T X^k \tilde{P}^T \tilde{P} X^k c \\
 &= -\|\tilde{P} X^k c\|^2 \leq 0.
 \end{aligned}$$

5. Mostrad que si en una iteración del método del escalado afín se tiene una dirección $\Delta x \geq 0$ donde $\Delta x \neq 0$, entonces el problema es ilimitado (esto es, podemos disminuir el valor de la función objetivo tanto como queramos).

SOLUCIÓN

Sabemos que, si $\Delta x \neq 0$, la dirección Δx calculada es siempre de descenso, verificándose que $c^T \Delta x < 0$ (ver ejercicio 4). El nuevo punto se obtiene como $x^{k+1} = x^k + \alpha \Delta x$, y la mejora en nuestra función objetivo respecto el punto x^k viene dada por $\alpha c^T \Delta x$. Ahora bien, como $\Delta x \geq 0$ no podemos llevar ninguna componente de $x^k > 0$ a 0, por lo que el valor de α (que es la longitud de paso que podemos realizar manteniendo la no-negatividad de x^k) sería infinito, y la reducción de coste que obtendríamos $\alpha c^T \Delta x$ también, concluyendo que el problema es ilimitado.

6. Considerad una partición de las columnas de $A = [B \ N]$ en una parte básica $B \in \mathbb{R}^{m \times m}$ y otra no básica $N \in \mathbb{R}^{m \times (n-m)}$ de forma análoga a como se realiza en el método del simplex. Utilizando una partición equivalente para $x^k = [x_B^T \ x_N^T]^T$ y $c = [c_B^T \ c_N^T]^T$, y suponiendo que no existe degeneración (esto es, que $x_B > 0$), mostrad que el vector de estimadores duales $y = (A(X^k)^2 A^T)^{-1} A(X^k)^2 c$ definido en (21) verifica

$$\lim_{x_N \rightarrow 0} y = (B^T)^{-1} c_B,$$

siendo entonces equivalente con el vector de las variables duales obtenido en el método del simplex.

SOLUCIÓN

En primer lugar expresemos y según el particionamiento de matrices y vectores del enunciado:

$$\begin{aligned}
 y &= (A(X^k)^2 A^T)^{-1} A(X^k)^2 c \\
 &= \left((B \ N) \begin{pmatrix} X_B^2 & \\ & X_N^2 \end{pmatrix} \begin{pmatrix} B^T \\ N^T \end{pmatrix} \right)^{-1} (B \ N) \begin{pmatrix} X_B^2 & \\ & X_N^2 \end{pmatrix} \begin{pmatrix} c_B \\ c_N \end{pmatrix} \\
 &= (B X_B^2 B^T + N X_N^2 N^T)^{-1} (B X_B^2 c_B + N X_N^2 c_N).
 \end{aligned}$$

A partir de la expresión anterior, y utilizando que $B X_B^2 B^T$ es no singular por ser $x_B > 0$, se obtiene el resultado deseado:

$$\lim_{x_N \rightarrow 0} y = (B X_B^2 B^T)^{-1} (B X_B^2 c_B) = (B^T)^{-1} (X_B^2)^{-1} B^{-1} B X_B^2 c_B = (B^T)^{-1} c_B.$$

7. Programad el algoritmo del escalado afín tal y como se presenta en el Algoritmo 1 de la subsección 5.8. Utilizad algún lenguaje de alto nivel que permita operaciones de cálculo matricial, como Matlab.

SOLUCIÓN

La codificación del algoritmo en el lenguaje de Matlab es como sigue:

```
[1] function xopt= eap(A,b,c,info)
[2]
[3] % uso: eap(A,b,c,info)
[4] %     A es una matriz (m x n)
[5] %     b es un vector columna de dimension m
[6] %     c es un vector columna de dimension n
[7] %     info indica nivel de informacion:
[8] %         si info>0 se muestra informacion a cada iteracion
[9] %         si info=0 se muestra la solucion obtenida unicamente
[10] % Halla la solucion del problema de programacion lineal
[11] %     min    c'x
[12] %     suj.a  Ax=b
[13] %         x>=0
[14] % aplicando el algoritmo del escalado afin (primal)
[15]
[16] % comprobamos parametro de informacion
[17] if (nargin==3) info=0;
[18] end%if
[19]
[20] % comprobamos que las dimensiones son correctas
[21]
[22] [m,n]= size(A);
[23] [mc,nc]= size(c);
[24] [mb,nb]= size(b);
[25]
[26] % comprobamos si c y b son vectores columna
[27] if ((nc ~= 1) | (nb ~= 1))
[28]     error('Error: c y b deben ser vectores columna');
[29] end%if
[30]
[31] % comprobamos si las dimensiones de b y c concuerdan con la matriz A
[32] if ((mc ~= n) | (mb ~= m))
[33]     error('Error: dimensiones incompatibles para c, b y A');
[34] end%if
[35]
[36] % comprobamos rango de A
[37] if (rank(A) ~= m)
[38]     error('Error: la matriz A no tiene rango completo');
[39] end%if
[40]
[41] % parametros
[42] optgap= 1.0e-6;
[43] myeps= 1.0e-12;
[44] rho= 0.95;
[45] M= 100*(max(abs(c)));
[46]
[47] % calculamos punto factible inicial, ampliamos el problema e iteramos
[48] x= ones(n,1);
[49] A= [A b-A*x];
[50] x= [x;1];
[51] c= [c;M];
[52] D= diag(x)^2;
[53] y= inv(A*D*A')*A*D*c;
[54] i=0;
[55] dualgap= abs(c'*x-b'*y)/(1.0+abs(c'*x));
[56] while ( dualgap > optgap )
```

```

[57]         z= c-A'*y;
[58]         dx= -D*z;
[59]         if all(dx>myeps)
[60]             x= inf*ones(n,1);
[61]             fprintf('Problema ilimitado\n');
[62]             break;
[63]         end%if
[64]         alph= min(-x(dx<myeps) ./ dx(dx<myeps));
[65]         if (info > 0)
[66]             % escribimos informacion
[67]             fprintf('\n\t\t\t Iteracion %d\n',i);
[68]             fobj= c'*x;
[69]             x,y,z,dx,alph,dualgap,fobj
[70]         end%if
[71]         x= x+rho*alph*dx;
[72]         D= diag(x)^2;
[73]         y= inv(A*D*A')*A*D*c;
[74]         dualgap= abs(c'*x-b'*y)/(1.0+abs(c'*x));
[75]         i=i+1;
[76]     end%while
[77]     if (x(n+1)>1.0e-4)
[78]         fprintf('Problema infactible\n');
[79]         xopt=nan*ones(1,n);
[80]     else
[81]         xopt=x(1:n);
[82]     end%if
[83] %endfunction
[84]

```

Puede comprobarse que se ha utilizado un valor de $\varepsilon = 10^{-6}$ para detectar la finalización del algoritmo (línea 44), de $\rho = 0.95$ (línea 46), y una constante $M = 100 \cdot \max\{|c_i| \mid i = 1, \dots, n\}$ como coste de la columna de infactibilidades del método Big-M (línea 47).

8. Solucionad el problema lineal

$$\begin{aligned}
 & \min \quad 3x_1 + x_2 \\
 & \text{sujeto a} \\
 & \quad 2x_1 + x_2 \geq 2 \\
 & \quad 3x_1 + 4x_2 \leq 12 \\
 & \quad x_1 \geq 0 \quad x_2 \geq 0.
 \end{aligned}$$

aplicando el algoritmo del escalado afín.

SOLUCIÓN

El punto solución es $x^* = (0 \ 2)^T$. Se dejan los cálculos al lector.

7. Trabajo práctico: resolución de un problema de ingeniería mediante el escalado afín

En esta sección sugerimos un ejercicio de carácter práctico. Consiste en solucionar un problema de ingeniería mediante una implementación del Algoritmo 1 similar a la presentada en el ejercicio 7. El problema consiste en la optimización de un sistema turbo-generator, y constituye una simplificación del presentado en Edgar y Himmelblau (1988). Las subsecciones 7.1 y 7.2 presentan el enunciado del ejercicio. La subsección 7.3 muestra la solución del mismo.

(Nota: la notación utilizada en la presentación del ejercicio para designar las variables y constantes del problema difiere de la del resto del documento. Así, por ejemplo, las variables se denotarán en minúscula y las constantes en mayúscula; los términos con subíndice no corresponderán a componentes de vectores).

7.1. Presentación del problema: optimización de un sistema turbo-generador

Las técnicas de Optimización son a menudo usadas para el diseño y determinación del régimen de funcionamiento de sistemas de vapor y turbo-generación en la industria química. Este tipo de sistemas funcionan con vapor de agua, el cual es utilizado para generar electricidad mediante el uso de turbinas. En este caso concreto intentaremos determinar los valores de las corrientes de vapor y flujos de potencia que minimizan los costes de un sistema de estas características.

La Fig. 15 muestra el esquema del sistema de vapor y turbo-generación eléctrica que será considerado. El agua se calienta en la caldera, usando un determinado combustible. El vapor generado se incorpora a una corriente de vapor de alta presión de v_a Kg/h. Con este vapor de alta presión se alimentan dos turbinas (denotadas por turb. 1 y turb. 2 en el esquema). Los flujos de entrada a cada turbina son de i_1 e i_2 Kg/h, y con éstos se genera una potencia eléctrica de p_1 y p_2 Kw, respectivamente. En el caso que no pueda satisfacerse una determinada demanda de potencia eléctrica mediante las dos turbinas será necesario adquirir p_e Kw de un proveedor externo. El vapor de salida de las turbinas, o_1 y o_2 , se incorpora a una corriente de vapor de media presión de v_m Kg/h. También es posible traspasar directamente v_{am} Kg/h de vapor de la corriente de alta a la de media presión a través de la válvula v_{am} .

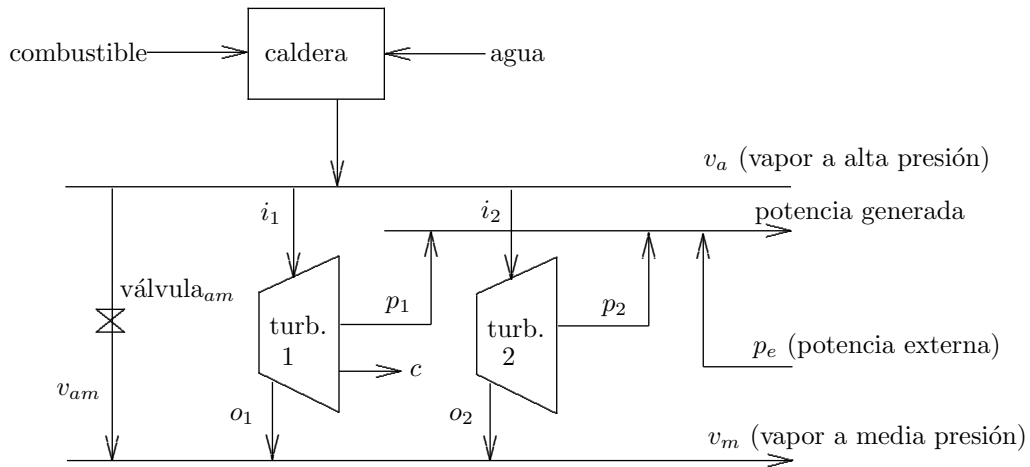


Figura 15. Esquema del sistema turbo-generador.

La turbina 1 es de doble extracción. Esto significa que realiza una doble transformación de la energía mecánica en energía eléctrica, primero pasando el vapor de alta a media presión, y después condensado este vapor de media presión, obteniéndose la corriente de c Kg/h de agua mostrada en la Fig. 15. La turbina 2, por su parte, es de extracción simple, y únicamente tiene una etapa de generación de energía eléctrica (pasando de alta a media presión).

Nuestro objetivo es satisfacer, con el coste mínimo, una determinada demanda de potencia eléctrica generada y de vapor a media presión. En este problema el coste se debe únicamente a la obtención de la corriente de alta presión en la caldera y a la adquisición de potencia eléctrica externa.

7.2. Modelización del problema

Una vez detallado el funcionamiento del proceso, pasamos a realizar la modelización de la función objetivo y de las diferentes restricciones que intervienen. Éstas se han agrupado 6 categorías: restricciones asociadas con la turbina 1 y con la turbina 2, balance de flujos del problema, balances de potencia del sistema, y demanda de energía y vapor a media presión.

Función objetivo

La función objetivo a minimizar es en este caso lineal, y únicamente debe contemplar la suma de los costes de producción de la corriente de vapor a alta presión y de adquisición de potencia externa. Supondremos que el coste de obtención de un Kg de vapor a alta presión ya ha sido calculado previamente, teniendo en cuenta el precio del combustible y la eficiencia de la caldera. El coste, en pesetas por hora de funcionamiento del sistema, viene dado entonces por

$$\text{función objetivo} = C_{v_a} v_a + C_{p_e} p_e \text{ pts/h,}$$

donde el término C_{v_a} (pts/Kg) es el coste de generación de un Kg de vapor a alta presión, y C_{p_e} (pts/Kwh) es el coste de adquisición de un Kwh de energía.

Restricciones asociadas con la turbina 1

Las restricciones que modelizan el funcionamiento de la turbina 1 son:

- Límite sobre el flujo máximo de entrada i_1 :

$$i_1 \leq \bar{I}_1.$$

- Límite sobre el condensado producido c :

$$c \leq \bar{C}.$$

- Límite sobre la potencia p_1 generada por la turbina:

$$p_1 \leq \bar{P}_1.$$

Restricciones asociadas con la turbina 2

Las restricciones que determinan el comportamiento de la segunda turbina son:

- Límite sobre el flujo máximo de entrada i_2 :

$$i_2 \leq \bar{I}_2.$$

- Límite sobre la potencia p_2 generada por la turbina:

$$p_2 \leq \bar{P}_2.$$

Restricciones de balance de flujos

Las ecuaciones de balance de flujos nos permiten modelizar las relaciones que hay entre los diferentes flujos y corrientes de vapor del sistema, según el esquema de la Fig. 15. Las unidades de todos los flujos son Kg/h. Estas restricciones son:

- Balance de flujos general sobre el vapor de entrada a alta presión, el de salida a media presión, y el condensado obtenido:

$$v_a = v_m + c.$$

- Balance de flujos en la turbina 1. El vapor de entrada debe ser igual a todo el vapor y condensado de salida:

$$i_1 = o_1 + c.$$

- Balance de flujos en la turbina 2. De nuevo, el vapor de entrada debe ser igual a todo el vapor de salida:

$$i_2 = o_2.$$

A la vista de esta ecuación podemos eliminar una de las dos variables. Eliminaremos o_2 y mantendremos i_2 .

- Balance de flujos en la corriente de alta presión. Lo que entra en esta corriente (v_a) debe ser igual a lo que sale:

$$v_a = i_1 + i_2 + v_{am}.$$

- Balance de flujos en la corriente de media presión. Como antes, todo lo que entra en esta corriente debe ser igual a todo lo que sale.

$$v_m = v_{am} + o_1 + o_2.$$

Puede observarse que esta ecuación es combinación lineal de las anteriores, por lo que será eliminada (garantizando que la matriz de restricciones tenga rango completo).

Restricciones de balance de potencias

Las ecuaciones de balance de potencias nos permitirán ligar las potencias p_1 y p_2 generadas con los diferentes flujos de vapor de entrada y salida de cada turbina. Para obtener la potencia de una determinada corriente de vapor (expresada en Kg/h), sólo debemos multiplicar su valor por una constante que nos proporciona la energía por cada Kg de vapor. Estas constantes se denotarán por E_a , E_m y E_c (en Kwh/Kg) para la corriente de alta presión, de media presión, y el condensado, respectivamente. Las ecuaciones de balance de potencias para cada turbina serán:

- Balance de potencias para la turbina 1:

$$E_a i_1 = E_m o_1 + E_c c + p_1.$$

- Balance de potencias para la turbina 2:

$$E_a i_2 = E_m o_2 + p_2.$$

Teniendo en cuenta que $i_2 = o_2$, tal y como hemos visto antes, podemos formular esta restricción como

$$(E_a - E_m) i_2 = p_2.$$

Restricciones de demanda del sistema

Finalmente sólo nos queda indicar cuanto vapor y potencia debe producir el sistema que estamos considerando.

- Demanda de vapor generado a media presión. Este valor v_m no debe ser inferior a su demanda D_{v_m} :

$$v_m \geq D_{v_m}.$$

- Demanda de potencia. La suma de la potencia generada y la externa adquirida no debe ser inferior a la potencia demandada D_p :

$$p_1 + p_2 + p_e \geq D_p.$$

En la modelización anterior hemos obtenido un total de 10 variables y 12 restricciones. De éstas, 5 eran de igualdad, 5 de \leq y 2 de \geq . Todas las variables están acotadas inferiormente por 0, dado que representan potencias o flujos de vapor. Añadiendo variables de holgura h_j $j = 1, \dots, 5$ para las restricciones de \leq y de exceso f_j $j = 1, 2$ para las de \geq , el problema finalmente obtenido consta de 17 variables y 12 restricciones, y puede ser formulado en forma estándar como sigue:

$$\begin{aligned}
& \min \quad C_{v_a} v_a + C_{p_e} p_e \\
& \text{sujeto a} \\
& \left. \begin{aligned} i_1 + h_1 &= \bar{I}_1 \\ c + h_2 &= \bar{C} \\ p_1 + h_3 &= \bar{P}_1 \end{aligned} \right\} \text{Turbina 1} \\
& \left. \begin{aligned} i_2 + h_4 &= \bar{I}_2 \\ p_2 + h_5 &= \bar{P}_2 \end{aligned} \right\} \text{Turbina 2} \\
& \left. \begin{aligned} v_a - v_m - c &= 0 \\ i_1 - o_1 - c &= 0 \end{aligned} \right\} \text{Balance de flujos} \\
& \left. \begin{aligned} v_a - i_1 - i_2 - v_{am} &= 0 \\ E_a i_1 - E_m o_1 - E_c c - p_1 &= 0 \end{aligned} \right\} \text{Balance de potencias} \\
& \left. \begin{aligned} (E_a - E_m) i_2 - p_2 &= 0 \\ v_m - f_1 &= D_{v_m} \end{aligned} \right\} \text{Demanda del sistema} \\
& p_1 + p_2 + p_e - f_2 = D_p \\
& v_a, v_m, v_{am}, i_1, i_2, o_1, c, p_1, p_2, p_e \geq 0 \\
& h_j \geq 0 \quad j = 1, \dots, 5 \quad f_1, f_2 \geq 0
\end{aligned} \tag{35}$$

El ejercicio consiste en solucionar el problema (35) mediante una implementación del algoritmo del escalado afín similar a la del ejercicio 7. Para ello se utilizarán los datos particulares presentados en la Tabla 3.

Tabla 3. Datos del problema

Coeficiente	Valor	Unidad
C_{v_a}	1	pts/Kg
C_{p_e}	5	pts/Kwh
\bar{I}_1	192000	Kg/h
\bar{I}_2	150000	Kg/h
\bar{C}	40000	Kg/h
\bar{P}_1	12000	Kw
\bar{P}_2	6000	Kw
D_{v_m}	15000	Kg/h
D_p	2000	Kw
E_a	0.8779	Kwh/Kg
E_m	0.8185	Kwh/Kg
E_c	0.1246	Kwh/Kg

7.3. Resolución del problema

El problema (35) se ha solucionado utilizando la implementación del Algoritmo 1 realizada en el ejercicio 7. Para mejorar el condicionamiento numérico de los cálculos (especialmente a la hora de factorizar la matriz ADA^T) se han escalado los términos independientes $\bar{I}_1, \bar{C}, \bar{P}_1, \bar{I}_2, \bar{I}_2, D_{v_m}$ y D_p , dividiéndolos por 1000 (por tanto los flujos de vapor son de 1000·Kg/h, las potencias 1000·Kw, y la función objetivo está en 1000·pts/h durante el proceso de optimización). Hay que hacer notar que sin este escalado previo de los datos el programa abortaba por problemas de inestabilidad numérica. Una implementación más robusta, y realizada en un lenguaje de más bajo nivel, evitaría este inconveniente.

El algoritmo del escalado afín ha realizado 14 iteraciones para encontrar el punto óptimo. La solución proporcionada se muestra en la Tabla 4. Los datos se presentan ya en las unidades originales (Kw, Kg/h) tras haber multiplicado por 1000 el óptimo obtenido. Se observa como hay 12 valores significativamente diferentes de 0, mientras que 5 de ellos son prácticamente nulos (si hubiéramos utilizado el algoritmo del símplex serían 0 y corresponderían a las variables no básicas). El coste óptimo es de 66475.99 pts/h. En la Fig. 16 se representa gráficamente la solución obtenida. Pueden observarse algunas discrepancias en los balances de flujo, motivadas por el hecho que ninguna variable, por ser interior, llega a tener el valor de 0.

Tabla 4. Solución óptima proporcionada por el método del escalado afín primal

Variable	Valor óptimo	Holgura	Valor óptimo
v_a	30929.92	h_1	176070.07
v_m	15000.01	h_2	0.00
v_{am}	0.01	h_3	24070.09
i_1	15929.92	h_4	135000.01
i_2	14999.99	h_5	5109.21
o_1	0.01		
c	15929.91	Exceso	Valor óptimo
p_1	12000.00	f_1	0.00
p_2	890.79	f_2	0.01
p_e	7109.21		

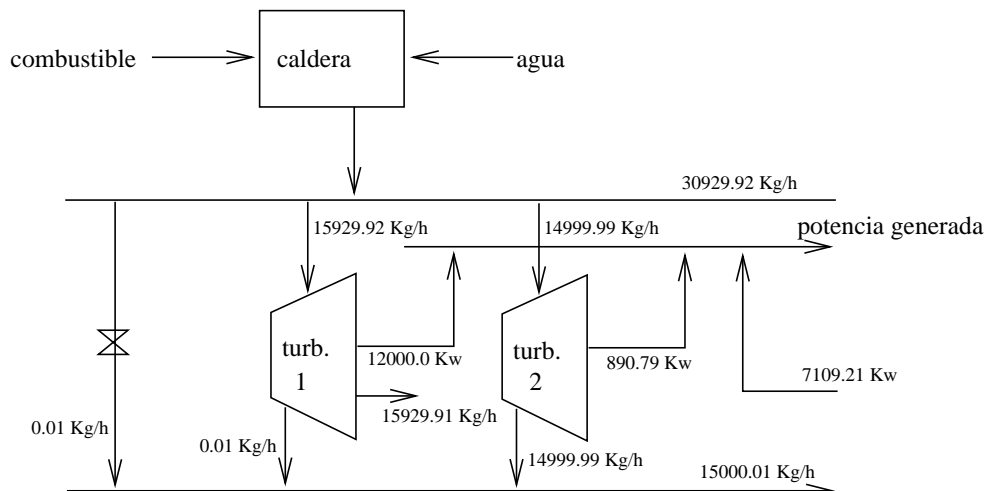


Figura 16. Representación de la solución óptima obtenida.

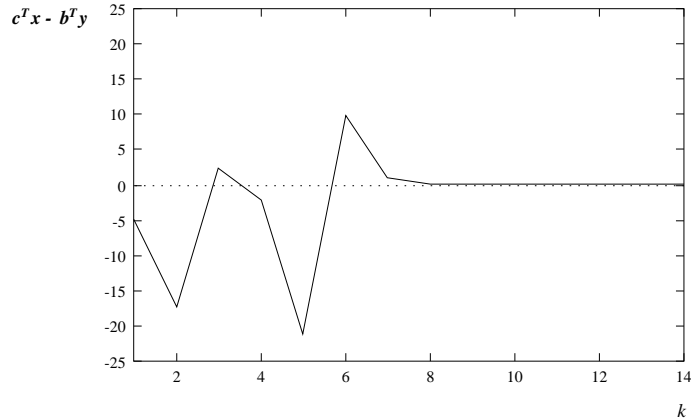


Figura 17. Evolución del gap dual durante la optimización del sistema turbo-generador por el método del escalado afín.

Finalmente, en la Fig. 17 se muestra la evolución del gap dual $c^T x - b^T y$ durante las 14 iteraciones realizadas por el algoritmo. Se observa como tiende hacia 0 a medida que nos acercamos al punto óptimo. También puede comprobarse como el punto (vértice) solución ha sido identificado ya en la iteración número 8. En el resto de iteraciones el algoritmo únicamente realiza movimientos de aproximación a este punto.

8. Aspectos del algoritmo no tratados

Existen diversos aspectos del método que no han sido tratados y que podrían incluirse en una presentación más extensa. Entre estos destacamos los siguientes:

- La versión del método del escalado afín presentado soluciona el problema (P) en forma primal. Por este motivo se denomina normalmente método del *escalado afín primal*. Existe una versión, denominada *escalado afín dual*, que soluciona el problema dual. En este caso se itera con las variables duales y y z , y se realizan estimaciones de las variables primales x . La principal ventaja del método dual es que puede hallarse un punto inicial de forma directa para aquellos problemas donde las variables primales estén acotadas superior e inferiormente, sin utilizar la técnica Big-M. El principal inconveniente es que, al iterar en el espacio dual, proporciona únicamente estimaciones de las variables x originales del problema. Para una descripción de este método puede consultarse el capítulo 6 de Arbel (1993).
- A la hora de realizar una implementación eficiente existen diversos detalles a tener en cuenta. A cada iteración debe factorizarse el sistema $A(X^k)^2 A^T$, que es simétrico, definido positivo y tiene el mismo patrón de esparsidad para toda X^k . Esto permite que sea posible aplicar técnicas específicas para sistemas simétricos y definidos positivos de gran dimensión. Para una presentación de este tipo de técnicas se podría utilizar el texto clásico de George y Liu (1981). El hecho de contar con una columna densa en A también debe ser tenido en cuenta, puesto que puede degradar completamente la esparsidad de $A(X^k)^2 A^T$. Esto ocurre, por ejemplo, al añadir la nueva columna de infactibilidades en el método Big-M. Entonces pueden aplicarse tanto técnicas de descomposición del sistema en bloques (descomposición de Schur), como métodos iterativos de solución de sistemas simétricos y definidos positivos (como el del gradiente conjugado). En el trabajo de Andersen *et al.* (1996) pueden obtenerse más detalles sobre este tema.
- El algoritmo del escalado afín (dual) ha sido utilizado con éxito en la optimización de problemas de flujos en redes, solucionando el sistema ADA^T de forma iterativa (mediante

un gradiente conjugado). Para una descripción de esta aplicación concreta del escalado afín puede consultarse Resende y Veiga (1993). Métodos alternativos de punto interior también han sido aplicados con éxito a problemas de flujos en redes denominados multiartículo —diferentes artículos comparten la misma red—, solucionando el sistema de ecuaciones ADA^T mediante un gradiente conjugado precondicionado. Pueden hallarse más detalles en Castro (1999).

- Los resultados de convergencia del algoritmo presentados en la subsección 5.9 podrían ampliarse. En particular, podría realizarse un análisis más profundo de las implicaciones de la degeneración del problema y como ésta afecta a la convergencia del método. El trabajo de Tsuchiya (1996) podría servir de base para una presentación de estas características.

9. Referencias

- [1] E.D. Andersen, J. Gondzio, C. Mészáros y X. Xu (1996), “Implementation of interior point methods for large scale linear programming”, *Interior Point Methods of Mathematical Programming*, ed. T. Terlaky, Kluwer Academic Publishers, The Netherlands.
- [2] A. Arbel (1993), *Exploring Interior-Point Linear Programming: Algorithms and Software*, The MIT Press, Cambridge, USA.
- [3] E.R. Barnes (1986), “A variation on Karmarkar’s algorithm for solving linear programming problems”, *Mathematical Programming*, 36, pp. 174–182.
- [4] D. Bertsimas y J.N. Tsitsiklis (1997), *Introduction to Linear Optimization*, Athena Scientific, Belmont MA.
- [5] D.P. Bertsekas (1995), *Nonlinear Programming*, Athena Scientific, Belmont, USA.
- [6] J. Castro (1999), “A specialized interior point algorithm for multicommodity flows”, *SIAM Journal on Optimization* (pendiente de publicación).
- [7] I.I. Dikin (1967), “Iterative solution of problems of linear and quadratic programming”, *Soviet Mathematics Doklady* 8(3), pp. 674–675.
- [8] T.F. Edgar y D.M. Himmelblau (1988), *Optimization of Chemical Processes*, McGraw-Hill, New York.
- [9] D.M. Gay (1985), “Electronic mail distribution of linear programming test problems”. *Mathematical Programming Society COAL Newsletter*, 13 10–12.
- [10] J.A. George y J.W.H. Liu (1981), *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Englewood Cliffs, USA
- [11] B.A. Murtagh y M.A. Saunders. (1983), “MINOS 5.0. User’s guide”, Dept. of Operations Research, Stanford University, USA.
- [12] S.G. Nash y A. Sofer (1996), *Linear and Nonlinear Programming*, McGraw-Hill, Singapore.
- [13] S. Park *et al.* (1997), “Lpabo 5.2 User’s Manual”, Department of Industrial Engineering, Seoul National University.
- [14] M. Resende y G. Veiga (1993), “An implementation of the dual affine scaling algorithm for minimum cost flow on bipartite uncapacitated networks”, *SIAM Journal of Optimization*, 3, pp. 516–537.
- [15] T. Tsuchiya (1996), “Affine Scaling Algorithm” en *Interior Point Methods of Mathematical Programming*, ed. T. Terlaky, Kluwer Academic Publishers, The Netherlands.

- [16] R.J. Vanderbei (1996), *Linear Programming: Foundations and Extensions*, Kluwer Academic Publishers, Boston, USA.
- [17] R.J. Vanderbei, M.S. Meketon y B.A. Freedman (1986), “A modification of Karmarkar’s linear programming algorithm”, *Algorithmica*, 1, pp. 395–407.