

# MÉTODO DE KARMARKAR

JUAN FÉLIX AVILA HERRERA<sup>1</sup>

---

## Resumen

Este es el primero de una serie de dos artículos en los que se estudia el algoritmo de Karmarkar. En el presente se da un enfoque ciertamente novedoso sobre lo que se llamará el método de Karmarkar. Se trata aquí de dar una versión de este algoritmo que permita una fácil implementación.

## Abstract

This is the first of a series of two articles in which we study the Karmarkar's Algorithm. In this article we present a new approach that will be called the Karmarkar's method. We intent to present a version of this algorithm that could be implemented easily.

**Palabras clave:** Algoritmo de Karmarkar, Método de Karmarkar, proceso de Karmarkar, transformaciones proyectivas, matrices ralas, esquema de purificación.

## 1 Descripción del método de Karmarkar

El algoritmo de Karmarkar al igual que el símplex no resuelve los programas lineales en su forma inicial. Es necesario transformar el problema en otro equivalente, de suerte que este último se ajuste a los requerimientos que exige el algoritmo de Karmarkar. Llamaremos *método de Karmarkar* al proceso que toma un problema en forma canónica:

$$\begin{aligned} &\text{Maximizar : } z = \sum_{j=1}^n c_j x_j, \\ &\text{sujeta a: } \begin{cases} \sum_{j=1}^n a_{ij} x_j \geq b_i & i = 1, \dots, m, \\ x_j \geq 0, & j = 1, \dots, n, \end{cases} \end{aligned} \quad (1)$$

lo convierte y halla la solución del problema original, o bien concluye sobre la vacuidad o no unicidad de la solución.

**Definición 1** Un programa lineal se dice que está en la *forma estándar de Karmarkar* (FEK) si es de la forma:

$$\text{Minimizar : } z = cx, \quad \text{sujeta a: } Ax = 0, \quad U_n x = 1, \quad x \geq 0, \quad (2)$$

---

<sup>1</sup>ESCUELA DE INFORMÁTICA, UNIVERSIDAD NACIONAL

donde  $A$  es  $m \times n$  de rango  $m$ ,  $n \geq 2$ ,  $c$  y  $A$  están compuestos por números enteros,  $\mathcal{U}_n$  es un vector con  $n$  unos, y cumple las siguientes condiciones **(S1)** el punto  $x_0 = \left(\frac{1}{n}, \dots, \frac{1}{n}\right)$ , es factible para (2), y **(S2)** el valor objetivo óptimo del problema (2) es cero.

El método de Karmarkar puede dividirse en tres fases claramente delimitadas:

1. **(Fase I)** Convertir el problema (1), a la FEK (2).
2. **(Fase II)** Hallar una solución factible y “satisfactoria” para el problema de programación lineal dado en la forma estándar de Karmarkar.
3. **(Fase III)** En la fase II se obtiene una solución  $x$  factible pero no necesariamente es un punto extremo del poliedro respectivo. Se debe emplear aquí algún mecanismo (comúnmente llamado esquema de purificación) para encontrar la solución de punto extremo.

## 2 Fase I: Convirtiendo un programa lineal a la forma de Karmarkar

Considérese el siguiente problema de programación lineal<sup>2</sup>:

$$\text{Maximizar } \{ cx : Ax \geq b, x \geq 0 \}, \quad (3)$$

donde  $A$  es  $m \times n$  de rango  $m$  y los datos son todos enteros. Desglosamos el proceso de conversión en los siguientes pasos.

1. Por dualidad de programación lineal, las condiciones de optimalidad para el problema (3) requieren una solución, si existe alguna, para el sistema:

$$\begin{array}{lll} Ax \geq b, & A^T u \leq c, & \text{y} \quad cx = bu, \\ x \geq 0, & u \geq 0 & \end{array}$$

en donde  $x = (x_1, \dots, x_n)$ , y  $u = (x_{m+n+1}, \dots, x_{2m+n})$ .

2. Se agregan  $y = (x_{n+1}, \dots, x_{m+n})$ , y  $v = (x_{2m+n+1}, \dots, x_{2m+2n})$  (vectores de holgura) al sistema anterior, y se obtiene:

$$\begin{array}{lll} Ax - y = b, & A^T u + v = c, & \text{y} \quad cx - bu = 0. \\ x, y \geq 0, & u, v \geq 0 & \end{array} \quad (4)$$

3. Con el fin de crear un punto interior de partida  $a$ , se introduce una variable artificial  $x_{2m+2n+1}$ . Sean  $x_0, y_0, u_0$  y  $v_0$ , vectores con componentes estrictamente positivos, por ejemplo<sup>3</sup>

$$x_0 = \mathcal{U}_n, y_0 = \mathcal{U}_m, u_0 = \mathcal{U}_m, v_0 = \mathcal{U}_n. \quad (5)$$

<sup>2</sup>El proceso que se explicará, puede ser aplicado también sobre la forma estándar de un programa lineal. Sin embargo, el dual del problema estándar ofrece algunas dificultades extra, como por ejemplo, variables no restringidas en signo.

<sup>3</sup>La elección de  $x_0, y_0, u_0$  y  $v_0$ , es bastante libre, se pide solamente que sean puntos estrictamente interiores en  $P_+ := \{x : x_i > 0, \forall i\}$ . El lector puede perfectamente escogerlos de otra manera.

Para dar al lector la libertad de escoger  $x_0$ ,  $y_0$ ,  $u_0$  y  $v_0$ , de otra forma, en lo que sigue, se brindarán las fórmulas para el caso general, e inmediatamente la forma particular que adoptan éstas bajo el escogimiento que se ha hecho anteriormente.

Se construye, entonces, el punto  $a$  como  $a := (x_0, y_0, u_0, v_0, 1)$ , o bien  $a = \mathcal{U}_{2m+2n+1}$  bajo la elección (5). De esta forma, el programa lineal que se debe resolver es

$$\begin{aligned} & \text{Minimizar : } z = x_{2m+2n+1} \\ & \text{sujeta a : } \begin{cases} Ax - y + (b + y_0 - Ax_0)x_{2m+2n+1} = b, \\ A^T u + v + (c - v_0 - A^T u_0)x_{2m+2n+1} = c, \\ cx - bu + (bu_0 - cx_0)x_{2m+2n+1} = 0, \\ x_i \geq 0, \quad \text{con } 1 \leq i \leq 2m + 2n + 1. \end{cases} \end{aligned}$$

Obsérvese que  $a := (x_0, y_0, u_0, v_0, 1)$ , es una solución factible para el problema anterior, y se toma como punto de partida. El valor mínimo de  $x_{2m+2n+1}$  es cero si y solamente si el problema (4) es factible.

4. Para escribir el sistema anterior en forma matricial primero se definen  $I_k$ , como la matriz identidad de tamaño  $k \times k$ ,  $\mathcal{Z}_{p \times q}$ , como la matriz nula de tamaño  $p \times q$ , los vectores  $\alpha$  y  $\beta$  como  $\alpha = b + y_0 - Ax_0$ ,  $\beta = c - v_0 - A^T u_0$  que en el caso (5) toman la forma  $\alpha = b + \mathcal{U}_m - A\mathcal{U}_n$  y  $\beta = c - \mathcal{U}_n - A^T \mathcal{U}_m$ , y el escalar  $\gamma$  como  $\gamma = bu_0 - cx_0$ , que en el caso particular (5) se convierte en  $\gamma = \sum_{i=1}^m b_i - \sum_{i=1}^n c_i$ . De esta forma el programa se puede reescribir como:

$$\text{Minimizar : } z = x_{2m+2n+1}, \quad Hx = f, \quad x \geq 0, \quad (6)$$

en donde  $f = \begin{bmatrix} b \\ c \\ 0 \end{bmatrix}$ , y  $H$  es una matriz de tamaño  $(m+n+1) \times (2m+2n+1)$ , dada por

$$H := \left[ \begin{array}{cc|cc|c} A_{m \times n} & -I_m & \mathcal{Z}_{m \times (m+n)} & & \alpha_{m \times 1} \\ & & A_{n \times m}^T & I_n & \beta_{n \times 1} \\ c_{1 \times n} & \mathcal{Z}_{1 \times m} & -b_{1 \times m} & \mathcal{Z}_{1 \times n} & \gamma \end{array} \right].$$

5. Se define ahora la transformación *proyectiva*  $T : \mathbb{R}^{2(m+n)+1} \rightarrow \mathbb{R}^{2(m+n)+1}$  dada por  $T(x) = x'$ , en donde  $x'_i := \frac{x_i/a_i}{1 + \sum_{j=1}^{2(m+n)+1} (x_j/a_j)}$ , que en el caso (5) toma la forma

$$x'_i := \frac{x_i}{1 + \sum_{j=1}^{2(m+n)+1} x_j}, \quad \text{para } i = 1, 2, \dots, 2(m+n)+1, \quad \text{y } x'_{2(m+n)+1} := 1 - \sum_{i=1}^{2(m+n)+1} x'_i.$$

Se definen también:

$$P_+ := \left\{ x \in \mathbb{R}^{2(m+n)+1} : x \geq 0 \right\}, \quad \text{y}$$

$$\Delta := \left\{ x \in \mathbb{R}^{2(m+n)+1} : x \geq 0 \quad \sum_{i=1}^{2(m+n)+1} x_i = 1 \right\}.$$

La transformación  $T$  tiene las siguientes propiedades:

- (a)  $T(P_+) \subseteq \Delta$ , i.e.  $T$  lleva  $P_+$  en  $\Delta$ .
- (b)  $T$  es uno-a-uno y la transformación inversa de  $T$ , está dada por  $T^{-1}(x') = x$ , o bien  $x_i = \frac{a_i x'_i}{x'_{2(m+n+1)}}$ , que bajo la elección (5) se convierte en  $x_i = \frac{x'_i}{x'_{2(m+n+1)}}$ , para  $i = 1, 2, \dots, 2(m+n)+1$ .
- (c) La transformación  $T$  asigna el centro del simplex  $\Delta$  al punto  $a$ , esto quiere decir en el caso (5) que  $T(a) = \frac{1}{2(m+n+1)} \mathcal{U}_{2(m+n+1)}$ .
- (d) Si  $H_i$  es la  $i$ -ésima columna de la matriz  $H$ , entonces la ecuación  $Hx = f$ , se puede escribir como  $\sum_{i=1}^{2(m+n)+1} H_i x_i = f$ . La ecuación anterior se convierte

(usando  $T$ ) en  $\sum_{i=1}^{2(m+n)+1} H_i a_i x'_i - f x'_{2(m+n+1)} = 0$ , que en el caso (5) toma la

forma  $\sum_{i=1}^{2(m+n)+1} H_i x'_i - f x'_{2(m+n+1)} = 0$ . De esta forma, si se define la matriz

$$H' = [a_1 H_1 \mid a_2 H_2 \mid \cdots \mid a_{2(m+n)+1} H_{2(m+n)+1} \mid -f],$$

que en el caso (5) se reduce a  $H' := [H \mid -f]$ , la ecuación  $Hx = f$  es convertida

en  $\sum_{i=1}^{2(m+n)+1} H'_i x'_i = 0$ , o bien en  $H' x' = 0$ .

En resumen, para resolver el problema 3 utilizando la elección particular (5), se debe hallar la solución<sup>4</sup> de:

$$\begin{aligned} & \text{Minimizar : } z = x'_{2(m+n)+1} \\ & \text{sujeta a: } \begin{cases} H' x' = 0, \\ \sum_{i=1}^{2(m+n)+1} x'_i = 1, \\ x'_i \geq 0, \quad \text{para } 1 \leq i \leq 2(m+n)+1, \end{cases} \end{aligned} \quad (7)$$

donde:

$$H' := \left[ \begin{array}{cc|cc|c|c} A_{m \times n} & -I_m & \mathcal{Z}_{m \times (m+n)} & \alpha_{m \times 1} & -b_{m \times 1} \\ \mathcal{Z}_{n \times (m+n)} & & A_{n \times m}^T & I_n & \beta_{n \times 1} & -c_{n \times 1} \\ c_{1 \times n} & \mathcal{Z}_{1 \times m} & -b_{1 \times m} & \mathcal{Z}_{1 \times n} & \gamma & 0 \end{array} \right],$$

con:

$$\begin{aligned} \alpha &= b + \mathcal{U}_m - A \mathcal{U}_n, \\ \beta &= c - \mathcal{U}_n - A^T \mathcal{U}_m, \\ \gamma &= \sum_{i=1}^m b_i - \sum_{i=1}^n c_i. \end{aligned}$$

Una vez que se resuelva el problema (7), la solución de (3), está dada por

$$x_i = \frac{x'_i}{x'_{2(m+n+1)}}, \quad \text{para } i = 1, 2, \dots, n.$$

<sup>4</sup>Observe que en virtud de esta transformación podemos redefinir la FEK, pidiendo sencillamente que se minimice la  $(n-1)$ -ésima variable de decisión.

### 3 Fase II: El algoritmo proyectivo de Karmarkar

En esta sección se explicará el algoritmo de Karmarkar. Esta es la fase álgida del método.

Defínase  $\Delta$  como:

$$\Delta := \{ x : x \geq 0, \sum_{i=1}^n x_i = 1 \}. \quad (8)$$

Nótese que bajo las suposiciones (S1) y (S2), el problema (2) es factible ( $x_0$  es una solución) y acotado (la solución óptima se halla en el simplex  $\Delta$ ) y por tanto, tiene un valor objetivo óptimo. Si la restricción  $x \in \Delta$ , se pudiera reemplazar por  $x \in S$ , en donde  $S$  es una esfera, el problema (como se verá) se simplifica, pues el valor óptimo puede ser encontrado resolviendo un sistema lineal de ecuaciones. Para lograr esta simplificación se emplearán lo que se conoce como transformaciones proyectivas. Tales transformaciones convierten el problema dado en la FEK, en otro más simple. Una vez ahí, se considera una restricción de éste, utilizando una esfera con radio apropiado. Se halla el valor óptimo para este último problema, y dado que la transformación proyectiva en cuestión posee inversa, se regresa al problema original (el dado en la FEK) con un nuevo punto factible  $x_1$  (el primero era  $x_0$ ) con quizás mejor valor objetivo. El proceso se repite hasta alcanzar un valor óptimo apropiado (recuérdese que, en virtud de la suposición (S2), el valor objetivo óptimo del problema es 0). De acuerdo con [3] la sucesión ( $x_k$ ) que se genera tiende a cero conforme  $k \rightarrow +\infty$ , aunque no necesariamente en forma monótona. Por tanto, en la práctica, el algoritmo se puede detener cuando el valor objetivo esté “bastante cerca” de cero.

#### 3.1 Resumen del Algoritmo de Karmarkar

Antes de explicar la deducción del algoritmo de Karmarkar, se expondrá en forma algorítmica su funcionamiento<sup>5</sup>.

1. INICIACIÓN: ( $k = 0$ )

(a) Calcule el radio  $r = 1/\sqrt{n(n-1)}$ .

(b) Calcule  $L = \lceil 1 + \log(1 + |c_{j_{\max}}|) + \log(|\det_{\max}|) \rceil$ . Aquí  $\lceil \cdot \rceil$ , significa la parte entera superior y  $\log(\cdot)$  denota logaritmo en base 2, además,  $|c_{j_{\max}}| = \max_{1 \leq j \leq n} \{ |c_j| \}$ , y

la cantidad  $|\det_{\max}|$  es:

$$|\det_{\max}| = \max \{ |\det(B)| : B \text{ es una base para el problema (2)} \}.$$

La constante  $L$  se llama la *longitud de entrada* del problema (2). El cálculo directo de  $L$  es difícil y en la práctica se puede evitar usando otros criterios. Una forma de estimar  $L$  es utilizar  $\tilde{L}$  en donde:

$$\tilde{L} := \lceil 1 + \log(1 + |c_{j_{\max}}|) + \log(1 + m) + \sum_{i=1}^m \sum_{j=1}^n \log(1 + |a_{ij}|) \rceil, \quad (9)$$

y se satisface que  $L \leq \tilde{L}$ .

---

<sup>5</sup>La matriz de restricciones tecnológicas obtenida en la Fase I es de tamaño  $(m+n+1) \times 2(m+n+1)$ , no obstante por comodidad en la notación, se supondrá en lo que sigue, que su tamaño es  $m \times n$ .

- (c) Calcule  $\alpha = (n - 1)/3n$ .
- (d) Coloque  $x_0 = (1/n, \dots, 1/n)$ .
2. ELEGIR SOLUCIÓN: Si  $cx_k < 2^{-L}$ , la solución actual  $x_k$  es factible y satisfactoria, entonces parar el algoritmo. Dado que, en la práctica  $2^{-L}$  es muy pequeño, el algoritmo se puede detener si  $cx_k < \epsilon$ , en donde  $\epsilon > 0$  es alguna tolerancia predefinida.
3. PASO PRINCIPAL: Se definen las siguientes cantidades matriciales:
- (a) La matriz  $D_k = \text{diag} \{ x_{k1}, \dots, x_{kn} \}$ , en donde  $x_k = (x_{k1}, \dots, x_{kn})$ .
- (b) La matriz  $P = \begin{bmatrix} AD_k \\ \mathcal{U}_n \end{bmatrix}$ .
- (c) El vector  $\bar{c} = cD_k$ .

Se calcula entonces  $x^*$  como  $x^* = x_0 - \alpha r \frac{c_p}{\|c_p\|}$ , con  $c_p = [I - P^T(PP^T)^{-1}P] \bar{c}^T$ .

De esta forma se obtiene  $x_{k+1} = (D_k x^*) / (\mathcal{U}_n D_k x^*)$ . Incremente  $k$  en uno y regrese al paso ELEGIR SOLUCIÓN.

Para el cálculo de  $c_p$  se puede proceder resolviendo el sistema:

$$(PP^T)x = P\bar{c}^T, \quad (10)$$

mediante<sup>6</sup> una descomposición L-U, y calcular entonces  $c_p$  por  $c_p = \bar{c}^T - P^T x$ .

### 3.2 Deducción del algoritmo de Karmarkar

Se empieza con  $x_0 = (1/n, \dots, 1/n)$  y  $k = 0$ , el algoritmo ejecuta el siguiente paso iterativo, toda vez que se tenga una solución factible  $x_k > 0$ .

#### 1. (Primera simplificación)

Defina la transformación proyectiva<sup>7</sup>  $T : \Delta \rightarrow \Delta$ , en donde:

$$y = T(x) = \frac{D_k^{-1}x}{\mathcal{U}_n D_k^{-1}x}, \quad (11)$$

y  $D_k$  es la matriz diagonal  $D_k = \text{diag} \{ x_{k1}, \dots, x_{kn} \}$ , donde  $x_k = (x_{k1}, \dots, x_{kn})$ , ( $\Delta$  es el simplex descrito en (8)). Es claro que las entradas de  $T$  están dadas por:

$$y_i = \frac{x_i/x_{ki}}{\sum_{j=1}^n x_j/x_{kj}}, \quad \text{para } i = 1, \dots, n. \quad (12)$$

<sup>6</sup>Nótese que esta es la *ecuación normal* para el problema de “mínimos cuadrados”: hallar  $x$  que minimiza  $\|P^T x - \bar{c}^T\|$ , consúltese [5].

<sup>7</sup>Las transformaciones proyectivas de  $\mathbb{R}^n$  son descritas por la fórmula  $T(x) = \frac{Cx + d}{f^T x + g}$ , en donde  $C$  es una matriz  $n \times n$  con valores reales,  $d, f \in \mathbb{R}^n$ ,  $g \in \mathbb{R}$  y la matriz  $\begin{bmatrix} C & d \\ f^T & g \end{bmatrix}$ , es no-singular.

Nótese también que  $\sum y_i = 1$ , y que  $y \geq 0$  cuando  $x \geq 0$ , por tanto el codominio de  $T$  es efectivamente  $\Delta$ .

¿Qué pasa con el problema (2) bajo la transformación (11)? Veamos:

(a)  $T(x_k) = x_0$ . La demostración de ésto es trivial.

(b)  $T$  es invertible, de hecho:

$$T^{-1}(x) = \frac{D_k x}{\mathcal{U}_n D_k x}. \quad (13)$$

En efecto, si se pone  $Z(x) = D_k x / \mathcal{U}_n D_k x$  y se calcula  $(T \circ Z)(x) = \frac{x}{\mathcal{U}_n x} = x$ , ya que  $\mathcal{U}_n x = 1$ . De forma análoga  $(Z \circ T)(x) = x$ , y por tanto  $Z = T^{-1}$ .

(c) El problema (2) se transforma en:

$$\text{Minimizar } \left\{ \frac{cD_k x}{\mathcal{U}_n D_k x} : AD_k x = 0, \mathcal{U}_n x = 1, x \geq 0 \right\}. \quad (14)$$

## 2. (Segunda simplificación)

Se nota de (14), que aunque las restricciones permanecen lineales ya que  $Ax = 0$  es homogénea, la función objetivo se ha convertido en un cociente de funciones lineales. No obstante debido al supuesto (S2), el valor objetivo óptimo en la ecuación (14) es cero. Por tanto, podemos minimizar equivalentemente el numerador en el problema, ya que el denominador es positivo y acotado lejos de cero para todo  $x \in \Delta$ . Se obtiene así el siguiente problema:

$$\text{Minimizar } \{ \bar{c}x : Px = P_0, x \geq 0 \}, \quad (15)$$

$$\text{donde } \bar{c} = cD_k, \quad P = \begin{bmatrix} AD_k \\ \mathcal{U}_n \end{bmatrix} \quad \text{y} \quad P_0 = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}.$$

## 3. (Tercera simplificación)

En lugar de resolver el problema (15), que es equivalente a resolver el problema original, se tratará de optimizar una *restricción* más simple de este último. Aunque al resolver esta restricción no se resuelve (indirectamente) el problema original (pues los problemas no son equivalentes), sí se varía (tal vez mejora) la solución actual  $x_k$  vía  $T^{-1}$ . Para describir la restricción, considere primero la bola  $S$  inscrita en el simplex  $\Delta$ , es decir,  $B(x_0, r)$  en donde el radio  $r = 1/\sqrt{n(n-1)}$ . Ahora, en lugar de restringir el problema a  $S$ , se utilizará una bola más pequeña, a saber,  $B(x_0, \alpha r)$ , en donde  $0 < \alpha < 1$ , con  $\alpha = (n-1)/3n$ . La elección de  $\alpha$  no es trivial, y el lector<sup>8</sup> puede consultar los detalles en [3].

La restricción que se resolverá es entonces:

$$\text{Minimizar } \left\{ \bar{c}x : Px = P_0, (x - x_0)^T (x - x_0) \leq \alpha^2 r^2 \right\}. \quad (16)$$

---

<sup>8</sup>Al final de esta sección se discute un poco sobre la elección particular de  $\alpha$ .

Se nota, entonces, que la solución para el problema (16) es obtenida proyectando el gradiente negativo (pues se está minimizando, consulte [2]) de la función objetivo  $-\bar{c}^T$ , centrado en  $y_0$ , sobre el espacio nulo o la superficie determinada por la restricción  $Px = P_0$  y moviéndose desde  $y_0$  a lo largo de esta dirección proyectada hasta la frontera de la bola  $B(y_0, \alpha r)$ . Observe que la simplicidad de este proceso de solución es una consecuencia del hecho de que estamos optimizando una función lineal sobre una bola, y que estamos actualmente ubicados en el centro de ésta, de manera que, la frontera está siempre a la misma distancia; sin importar la dirección que se adopte. Si denotamos la proyección del vector gradiente<sup>9</sup>  $\bar{c}^T$  por  $c_p$  y la solución óptima del problema (16) como  $x^*$ , obtenemos:

$$x^* = x_0 - \alpha r \frac{c_p}{\|c_p\|}. \quad (17)$$

[Note que si  $c_p = 0$ , entonces cualquier solución factible es optimal,<sup>10</sup> y se debe terminar con  $x_k$  como solución óptima del problema (2).]

**Proposición 1** Se verifica  $c_p = [I - P^T(PP^T)^{-1}P]\bar{c}^T$ .

DEMOSTRACIÓN: Consulte [3].

□

La solución  $x^*$  obtenida en (17), debe ser ahora transformada al problema original (2). Para lograr ésto se usa la inversa de la transformación proyectiva, i.e.  $T^{-1}$ . De esta forma el correspondiente vector revisado  $x_{k+1}$  está dado por

$$x_{k+1} = \frac{D_k x^*}{\mathcal{U}_n D_k x^*}. \quad (18)$$

Observe que  $x^* > 0$ , ya que pertenece al interior de la esfera  $(n-1)$  dimensional inscrita en  $\Delta$ . Por tanto, el  $x_{k+1}$  dado en (18) es positivo. Esto completa una iteración. Este paso puede ser repetido después de incrementar  $k$  por uno. El proceso puede ser terminado cuando el valor objetivo está suficientemente cerca de cero.

Ya se ha dicho algo sobre la forma de detener el algoritmo de Karmarkar, no obstante es pertinente hacer algunas observaciones adicionales. Cuando se indicó (en la pág. 49) la forma de detener el proceso, se introdujo la constante  $L$ , que proporciona un criterio en este sentido. El resultado es como sigue.

**Teorema 1** El algoritmo de Karmarkar produce una solución  $x_k$  para el problema (2) que satisface  $cx_k < 2^{-L}$ , en  $10nL$  iteraciones, toda vez que  $\alpha = (n-1)/3n$ .

DEMOSTRACIÓN: Consulte [3].

□

---

<sup>9</sup>De esta forma,  $c_p$  pertenece a la superficie  $Px = P_0$ .

<sup>10</sup>Si  $c_p = 0$ , entonces  $\bar{c}$  es perpendicular a la región  $P_y = P_0$  del problema proyectado, entonces  $\bar{c}^T y = 0$  para todo  $y$  en tal región (i.e. se alcanza el costo óptimo).



Sobre el cálculo de la constante  $L$  se insta al lector a consultar [1, 3, 6], en donde se apunta que en la práctica es innecesario hacerlo. De hecho de 30 a 60 iteraciones son suficientes, independientemente del tamaño del problema. Por otro lado el mismo Karmarkar asegura que en la práctica  $L$  es mucho más pequeño que  $n$ .

Sería imperdonable terminar esta sección sin mencionar uno de los aspectos cruciales en las ideas de Karmarkar: la *función potencial*, que tiene injerencia directa en la elección del  $\alpha$  mencionado en el teorema 1. Como se sabe, el algoritmo construye una sucesión de puntos  $x_0, x_1, \dots, x_k$ , de suerte que éstos van generando valores decrecientes en la función objetivo, pero como se ha dicho antes, no en forma monótona. El problema es entonces: ¿Cómo asegurar que la sucesión así obtenida va convergiendo efectivamente hacia la solución óptima? Afortunadamente, existe una función que preserva monotonía estricta (dado que la función objetivo original no necesariamente lo hace) y por tanto asegura la convergencia hacia la solución óptima. Esta función es conocida como función potencial, y está dada por:

$$f(x) = \sum_{j=1}^n \ln \left[ \frac{cx}{x_j} \right] = n \ln(cx) - \sum_{j=1}^n \ln(x_j).$$

Al aplicar esta función potencial a la función objetivo del problema (14), se concluye (por supuesto no de forma trivial) que la sucesión generada por el algoritmo de Karmarkar converge a la solución óptima toda vez que el valor de  $\alpha$  se escoja en la forma que establece el teorema 1, consulte [3]. Es importante también mencionar que la elección particular de  $\alpha$  permite una velocidad de convergencia apropiada, de suerte que se aproxime la solución óptima en pocas iteraciones.

## 4 Fase III: Rutina de redondeo optimal

En la Fase II se mostró cómo determinar una solución  $x_k$  apropiada, satisfaciendo  $cx_k < 2^{-L}$ . Sin embargo,  $x_k$  no necesariamente es un punto extremo. Es por esto que se debe “redondear”  $x_k$ , de manera que se obtenga una solución óptima con esta cualidad, y tal que tenga tan buen valor objetivo como  $x_k$ .

Antes es conveniente hacer una definición.

**Definición 2** Sean  $c, x \in \mathbb{R}^n$  y  $b \in \mathbb{R}$ . Se dice que  $u \in \mathbb{R}^n$  satisface en forma *activa* la desigualdad  $c \cdot x \leq b$ , si  $c \cdot u = b$ . Análogamente  $u$  satisface en forma *activa*  $c \cdot x \geq b$  ó  $c \cdot x = b$  si  $c \cdot u = b$ .

A continuación se describe la rutina de redondeo expuesta en [3].

1. ELEGIR SOLUCIÓN: Si al sustituir  $x_k$  en el programa lineal (2),  $n$  restricciones linealmente independientes están siendo satisfechas en forma activa, se concluye entonces que  $x_k$  es una solución básica óptima. Parar.

Es importante aclarar que si  $m + 1 < n$ , no basta con que  $Ax_k = b$  y  $\mathcal{U}_n x = 1$ , en el problema (2). Si  $m + 1 < n$ , además de satisfacerse estas igualdades, se debe cumplir también, que algunas entradas de  $x_k$  (a saber  $n - m - 1$ ), deben ser nulas.

2. **CALCULAR DIRECCIÓN:** Si  $x_k$  no es una solución de punto extremo, existen varias restricciones (digamos  $l < n$ ) del problema (2) que  $x_k$  satisface en forma activa. Denotemos con  $\tilde{A}$  la matriz formada a partir de esas restricciones, de la siguiente forma:
- (a) Si una restricción de  $Ax = b$  se satisface en forma activa con  $x_k$ , se incluye la fila correspondiente de  $A$  en  $\tilde{A}$ .
  - (b) Si  $\mathcal{U}_n x_k = 1$ , el vector  $\mathcal{U}_n = (1, 1, \dots, 1)$  se incluye en  $\tilde{A}$ .
  - (c) Si la entrada  $i$ -ésima de  $x_k$  es nula, se agrega  $e_i = (0, 0, \dots, 1, \dots, 0)$  (con un único 1 en la  $i$ -ésima posición) a  $\tilde{A}$ .

La matriz  $\tilde{A}$  resultante es, entonces, de tamaño  $l \times n$  con  $l < n$ . Considere ahora el sistema  $\tilde{A}x = 0$  y sea  $d$  una solución no trivial de este sistema.

3. **ACTUALIZAR SOLUCIÓN:** Se obtiene una nueva solución  $x_k^*$ , moviéndose a lo largo de la dirección  $d$  si  $cd < 0$  y a lo largo de  $-d$  en otro caso<sup>11</sup>, hasta que algunas restricciones bloqueen cualquier movimiento adicional por consideraciones de factibilidad. Así:
- (a) si  $cd < 0$ , y todas las entradas de  $d$  son no negativas, se concluye que la solución es no acotada. En el caso contrario:  $x_k^* = x_k + \lambda \cdot d$ , en donde  $\lambda = \min_{1 \leq i \leq n} \{-x_{ki}/d_i : d_i < 0\}$ ,  $x_k = (x_{k1}, x_{k2}, \dots, x_{kn})$  y  $d = (d_1, d_2, \dots, d_n)$ .
  - (b) si  $cd \geq 0$ , y todas las entradas de  $d$  son no positivas, se concluye que la solución es no acotada. En el caso contrario:

$$x_k^* = x_k - \lambda \cdot d,$$

en donde:

$$\lambda = \min_{1 \leq i \leq n} \{x_{ki}/d_i : d_i > 0\}.$$

Se regresa entonces al paso ELEGIR SOLUCIÓN.

De acuerdo con [3], siguiendo este procedimiento se obtiene una solución básica factible  $\bar{x}$  para el problema (2), con valor objetivo estrictamente menor que  $2^{-L}$ . Esto concluye la fase III del método de Karmarkar.

El esquema de purificación tiene sus orígenes en 1965 en el trabajo de Charnes, Kortanek y Raíke, consulte [4]. Posteriormente, en 1988, Kortanek y Jishan (consulte [7]), mostraron cómo usar este esquema para lograr una solución básica factible óptima. El lector interesado en otros enfoques de este problema, puede consultar el trabajo de Ye, en el cual se explica cómo obtener una base óptima vía el algoritmo de Karmarkar, ver [8].

## 5 Conclusiones

Las tres fases que componen el método de Karmarkar no son, de ninguna forma triviales. En este artículo se ha explicado en qué consiste cada una de ellas. El enfoque elegido ha

---

<sup>11</sup>En virtud de la fase I, basta con considerar el signo de  $d[n-1]$ .

sido en su mayor parte algorítmico, esto permitirá al lector interesado, una implementación sin mayores contratiempos. La fase I del método de Karmarkar abre las puertas del próximo artículo. En efecto, después de convertir un programa lineal (1) en (2), la matriz resultante tiene una estructura rala que dificulta una implementación eficiente del método de Karmarkar tal cual. En el artículo siguiente se estudiarán algunos métodos para manejar tales matrices y de esta forma obtener una *implementación eficiente del algoritmo de Karmarkar*.

## Referencias

- [1] Anstreicher, K.M. (1990) “Progress in interior point algorithms since 1984”, *SIAM News* 3
- [2] Apostol, T.M. (1977) *Calculus*. Reverté, Barcelona, vol. II.
- [3] Bazaraa, M.S.; Jarvis, J.J.; Sherali, H. (1990) *Linear Programming and Network Flows, 2nd Edition*. John Wiley & Sons, New York.
- [4] Charnes, A.; Kortanek, K.O.; Raik, W. (1965) “Extreme point solution in mathematical programming: an opposite sign algorithm”, *System Research Memorandum* No. 129, Northwestern University, Evanston
- [5] Golub, G.H.; Van Loan, C.F. (1989) *Matrix Computations, 2nd Edition*. The Johns Hopkins University Press, Baltimore.
- [6] Karmarkar, N. (1984) “A new polynomial-time algorithm for linear programming”, *Combinatorica* 4
- [7] Kortanek, K.O.; Jishan, Z. (1988) “New Purification algorithms for linear programming”, *Naval Research Logistics Quarterly* 35
- [8] Ye, Y. (1990) “Recovering optimal basic variables in Karmarkar polynomial algorithm for linear programming”, *Mathematics of Operation Research* 3