PARAMETRIZACION DEL VECTOR COSTO EN MODELOS LINEALES DE OPTIMIZACION ESCALONADO

Alibeit Kakes, Departamento de Matemáticas Aplicadas, Facultad de Matemática y Computación, Universidad de La Habana.

RESUMEN
El objetivo de este artículo es diseñar un algoritmo de parametrización del vector costo para una clase particular de modelos lineales de optimización con matrices escalonadas. Se dan fórmulas de los intervalos de estabilidad de la solución. Dichos intervalos se construyen trabajando con inversas llamadas "locales", y no con la inversa de la matriz asociada al problema que se resuelve.

ABSTRACT
An algorithm for the parametrization of the cost vector of a particular class of linear programming problems with staircase matrices is designed. Formulas for the calculation of the intervals of stability are given, which are constructed by working only with the such called "local" inverses, instead of the inverse of the matrix A of the original problem.

1. INTRODUCCION

Luego de contar con una solución óptima de un problema de Programación Lineal, es común, en los problemas reales, conocer cuánto puede cambiar el vector costo, c, antes de que la solución hallada deje de ser óptima. Para ello se asume que se tiene una solución del problema P: min{c_0x : Ax = b, $x \geq 0$}; y se reemplaza c_o por $c_o + \alpha c_1$, con c_1 un vector dado fijo y α un parámetro no negativo. Se trata entonces de saber para qué valores de α se sigue cumpliendo el criterio de optimalidad. Se escribe entonces [2],

$$(c_{0B} + \alpha c_{1B})y_j - (c_{0j} + \alpha c_{1j}) = c_{0B}y_j - c_{0j} + \alpha(c_{1B}y_j - c_{1j}) = z_{0j} - c_{0j} + \alpha(c_{1B}y_j - c_{1j})$$

donde c_{1B} es un vector que contiene las componentes de c_1 que ocupan la misma posición de c_{0B} en c_0 y $z_{0j} = c_{0B}y_j$. Por analogía, se puede escribir $z_{0j} - c_{0j} + \alpha(z_{1j} - c_{1j})$. Se debe buscar un intervalo para α que mantenga no positiva la expresión anterior.

Se supone ahora que se tiene el siguiente modelo lineal escalonado [6], [8]; al que se desea parametrizar el vector costo.

$$\min J(\mu) = a(T)x(T) \tag{1}$$

sa:

$$G(t)x(t) + D(t)\mu(t) = f(t) \tag{2}$$

$$x(t+1) = A(t)x(t) + B(t)\mu(t) \tag{3}$$

$$x(0) \tag{4}$$

dado

$$\mu(t) \geq 0 \tag{5}$$

$$t = 0,...,T-1 \tag{6}$$

Las dimensiones de las matrices $G(t), D(t), A(t), B(t)$ son m×n , m×r, n×n, n×r respectivamente.

Los vectores a(T) y f(t) tienen dimensión n y m respectivamente.

El modelo (1) – (6) posee una matriz de orden $(m + n)T \times (r + n)T$.

En [6] y [8] se dan resultados teóricos y se diseña un algoritmo tipo Simplex que resuelve modelos lineales escalonados del tipo anterior. Tal algoritmo se caracteriza por dar expresiones para el cálculo del criterio de optimalidad, criterio de salida y entrada de la base, y pivoteo, que no depende explícitamente de las inversas de las bases del sistema dado por (2), (3), (5). Tales inversas son de orden $(m+n)T\times(m+n)T$, y sin embargo el algoritmo sólo utiliza T matrices de orden m x m , una para cada momento t. A esas matrices se les llama "bases locales" [8].

Observar que las variables x(t), una vez calculadas las variables u(t); pueden ser halladas a través de (3), teniendo en cuenta (4).

A continuación se expone el algoritmo:

Se supone que se cuenta con una solución factible básica $\hat{\mu}_{0B}(t)$; y variables x(t+1); t =0,...,T-1.

Algoritmo

Paso 1.

Calcular:

$$z_j(t) - c_j(t) = \lambda(t) d_j(t) - p(t+1) b_j(t);\ t = T\text{-}1,...,0;$$

donde

$$p(T) = a(T)$$

$$\lambda(t) = p(t+1)\hat{B}_{OB}(t)\hat{D}_{OB}^{-1}(t)$$

$$p(t) = p(t+1)A(t) - \lambda(t)G(t)$$

$$t = T\text{-}1,..,0$$

Los vectores $d_j(t)$ y $b_j(t)$ son las columnas de las matrices D(t) y B(t) respectivamente.
Notar que todos los cálculos se realizan paso a paso, desde T-1 a 0.

Paso 2.

Si $z_j(t) - c_j(t) \le 0$ para todo j y todo t; PARAR. El par $(\hat{\mu}_{OB}(t); x(t+1))$, t = 0,...,T-1 es la solución óptima del problema dado.

En caso contrario ir al siguiente paso.

Paso 3.

Calcular: (k, t_2)

$$z_k(t_1) - c_k(t_1) = \max\{z_j(t) - c_j(t) : z_j(t) - c_j(t) > 0\}$$

Paso 4.

Calcular los vectores $\hat{V}_{OB}(t) = \hat{V}^*_{OB}(t) - \hat{V}^1_{OB}(t);\ t = 0,...,T-1$

$$\hat{V}^*_{OB}(\tau) = 0;\ 0 \le \tau \le t_1 - 1$$

$$\hat{V}^*_{OB}(t_1) = \hat{D}^{-1}_{OB}(t_1) d_k(t_1)$$

$$\hat{V}^*_{OB}(\tau) = -\hat{D}^{-1}_{OB}(\tau) G(\tau) y^*(\tau);\ t_1 + 1 \le \tau \le T - 1$$

$$y^*(\tau) = 0;\ 0 \le \tau \le t_1$$

$$y^*(t_1+1) = \hat{B}_{OB}(t_1)\hat{V}^*_{OB}(t_1) - b_k(t_1)$$

$$y^*(\tau+1) = A(\tau)y^*(\tau) + \hat{B}_{OB}(\tau)\hat{V}^*_{OB}(\tau);\ t_1+1 \le \tau \le T-1$$

$$\hat{V}_{OB}(T-1) := \hat{V}^*_{OB}(T-1)$$

$$\hat{V}^1_{OB}(t) = \sum_{j=t+1}^{T-1}\left[\phi^j_B(t) : 0\right]\hat{V}_{OB}(j);\ t = T-1,\ldots,0$$

Los vectores $\hat{V}_{OB}(t)$, yuxtapuestos, formarían el clásico vector y_j [1], [2], asociado a la variable que "entra" en la base. Cada $\hat{V}_{OB}(t)$ es de m componentes. Notar que el cálculo se hace "localmente", hasta tener mT componentes.

Los vectores $y^*(\tau)$ tienen en las restricciones de tipo (3) el mismo significado que los vectores $\hat{V}_{OB}(t)$ en las restricciones de tipo (2).

Las matrices $\phi_B(t)$ están asociadas a columnas básicas del momento t que no forman parte de la base "local" elegida en ese momento [6], [8].

Paso 5.

Si $\hat{V}_{OB}(t) \le 0;$ para todo t; PARAR. El problema tiene solución no acotada.

Si existe $(i,t) : \hat{V}_i(t) > 0;$ ir al siguiente paso.

Paso 6.

Calcular el par de índices (l, t_2):

$$\bar{\theta} = \hat{\mu}_l(t_2)\big/\hat{V}_l(t_2) = \min\left\{\hat{\mu}_i(t)\big/\hat{V}_i(t) : \hat{V}_i(t) > 0\right\}$$

Paso 7.

Proceder con el cambio de base, pivotear, e ir al Paso1.

2. PARAMETRIZACION DE LA FUNCION OBJETIVO EN MODELOS LINEALES ESCALONADOS

Se desea, dada una solución $(\hat{\mu}, x)$ del problema (1) - (6), hallar un intervalo de estabilidad de dicha solución respecto a variaciones en la función objetivo. La búsqueda de ese intervalo debe obtenerse a partir de cálculos locales; esto es, operando de momento a momento, con t = 0, ...T - 1. Se dan también resultados que precisan lo que ocurre en los extremos del intervalo, y fuera de ellos. El aporte radica en que nunca se trabaja con la matriz original del problema. Trabajar con esa matriz es un problema clásico de optimización ([1] , [5]). El modelo parametrizado viene dado por:

$$\min J(\mu,\alpha) = \left[a_0(T) + \alpha a_1(T)\right]x(T)$$

sa:

$$G(t)x(t) + D(t)\mu(t) = f(t)$$

$$x(t+1) = A(t)x(t) + B(t)\mu(t)$$

$$x(0)$$

dado

$$\mu(t) \geq 0$$

$$t = 0,...,T-1$$

Se suponen las siguientes hipótesis:

1. $\text{rang}\begin{pmatrix} a_0(T) \\ a_1(T) \end{pmatrix} = 2$

2. rang D(t) = m; t = 0,...,T-1

3. $\alpha \in [\alpha_{inf}, \alpha_{sup}]$; $\alpha_{inf} < \alpha_{sup}$; fijos.

Las abreviaturas inf y sup significan inferior, y superior respectivamente.

4. Para $\alpha = \hat{\alpha}$ el problema tiene solución; sea esta $(\hat{\mu}, x)$.

Las hipótesis 1- 4 no son restrictivas. Sirven para simplificar la presentación y el desarrollo del algoritmo. Si consideramos el dual del problema parametrizado anterior, se tiene

$$\max \sum_{t=0}^{T-1} \lambda(t)f(t) + p(0)x(0)$$

s.a.

$$p(t+1)B(t) - \lambda(t)D(t) \leq 0$$

$$p(t) = p(t+1)A(t) - \lambda(t)G(t)$$

$$t = T-1,...,0$$

$$p(T) = a_0(T) + \alpha a_1(T) \tag{7}$$

Las variables $\lambda(t)$ y $p(t)$ están asociadas a las restricciones (2) y (3) respectivamente.

El proceso que sigue se hace necesario para, como en el caso de un modelo general de Programación Lineal, trabajar separadamente los $z_{0j} - c_{0j}$ asociados a $a_0(T)$ y los $z_{1j} - c_{1j}$ asociados a $a_1(T)$.

Se define:

$$p_0(T) = a_0(T); p_1(T) = a_1(T); \tag{8}$$

$$p(T,\alpha) = p_0(T) + \alpha p_1(T)$$

Del paso 1 del algoritmo se tiene

$$\lambda(t) = p(t+1)\hat{B}_{OB}(t)\hat{D}_{OB}^{-1}(t) \tag{9}$$

$$p(t) = p(t+1)A(t) - \lambda(t)G(t) \tag{10}$$

Sustituyendo (7) en (9) y teniendo en cuenta (8)

$$\lambda(T-1) = (p_0(T) + \alpha p_1(T))\hat{B}_{OB}(T-1)\hat{D}_{OB}^{-1}(T-1)$$

$$= p_0(T)\hat{B}_{OB}(T-1)\hat{D}_{OB}^{-1}(T-1) + \alpha p_1(T)\hat{B}_{OB}(T-1)\hat{D}_{OB}^{-1}(T-1)$$

Si se define

$$\lambda_0(T-1) = p_0(T)\hat{B}_{OB}(T-1)\hat{D}_{OB}^{-1}(T-1)$$

$$\lambda_1(T-1) = p_1(T)\hat{B}_{OB}(T-1)\hat{D}_{OB}^{-1}(T-1);$$

se obtiene

$$\lambda(T-1) = \lambda_0(T-1) + \alpha\lambda_1(T-1)$$

Como esa expresión depende de α, se define

$$\lambda(T-1,\alpha) = \lambda_0(T-1) + \alpha\lambda_1(T-1)$$

Si ahora repetimos la operaciones realizadas utilizando (10) y (9) para t =T-1 y t = T-2; se obtiene

$$p(T-1) = \left(p_0(T) + \alpha p_1(T)\right)A(T-1) - \left(\lambda_0(T-1) + \alpha\lambda_1(T-1)\right)G(T-1)$$

Efectuando las operaciones indicadas y agrupando, se define

$$p_0(T-1) = p_0(T)A(T-1) - \lambda_0(T-1)G(T-1)$$

$$p_1(T-1) = p_1(T)A(T-1) - \lambda_1(T-1)G(T-1)$$

finalmente se define

$$p(T-1,\alpha) = p_0(T-1) + \alpha p_1(T-1)$$

Realizando el mismo procedimiento hasta t = 0 se tiene la siguiente

Proposición 2.1

Sean $p_0(T) = a_0(T);\ p_1(T) = a_1(T);$ entonces

$$p(t,\alpha) = p_0(t) + \alpha;\ p_1(t);$$

$$\lambda(t,\alpha) = \lambda_0(t) + \alpha;\ \lambda_1(t);$$

donde

$$p_0(t) = p_0(t+1)A(t) - \lambda_0(t)G(t)$$

$$\lambda_0(t) = p_0(t+1)\hat{B}_{OB}(t)\hat{D}_{OB}^{-1}(t)$$

$$p_1(t) = p_1(t+1)A(t) - \lambda_1(t)G(t)$$

$$\lambda_1(t) = p_1(t+1)\hat{B}_{OB}(t)\hat{D}_{OB}^{-1}(t)$$

Corolario 2.1

Para t = T - 1,...,0 se tiene $z_j(t,\alpha) - c_j(t,\alpha) = z_{0j}(t) - c_{0j}(t) + \alpha\left[z_{1j}(t) - c_{1j}(t)\right]$

Demostración

Del Paso 1 del algoritmo se tiene

$$z_j(t) - c_j(t) = \lambda(t)d_j(t) - p(t+1)b_j(t) \;;$$

entonces

$$z_j(t,\alpha) - c_j(t,\alpha) = \lambda(t,\alpha)d_j(t) - p(t+1,\alpha)b_j(t) = \left(\lambda_0(t) + \alpha\lambda_1(t)\right) d_j(t) - (p_0(t+1) + \alpha p_1(t+1))b_j(t)$$

Agrupando:

$$z_j(t,\alpha) - c_j(t,\alpha) = z_{0j}(t) - c_{0j}(t) + \alpha\left[z_{1j}(t) - c_{1j}(t)\right] \;\blacksquare$$

A continuación se enuncia y demuestra el teorema que permite construir el intervalo de estabilidad asociado a la solución óptima.

Teorema 2.1

La solución $(\hat{\mu}, x)$ es óptima para todo α en $[\underline{\alpha}, \overline{\alpha}]$, donde; $\underline{\alpha} = \max\left\{\alpha_{\inf}, \max\left\{\underline{\alpha}(t)\right\} \;;\; t = T-1,\ldots,0\right\}$ $\overline{\alpha} = \min\left\{\alpha_{i\sup}, \min\left\{\overline{\alpha}(t)\right\}; t = T-1,\ldots,0\right\}$, con

$$\underline{\alpha}(t) = \max\left\{-\frac{z_{0j}(t) - c_{j0}(t)}{z_{1j}(t) - c_{1t}(t)}; \quad j : z_{1j}(t) - c_{1j}(t) > 0\right\}$$

$$\overline{\alpha}(t) = \min\left\{-\frac{z_{0j}(t) - c_{0j}(t)}{z_{1j}(t) - c_{1j}(t)}; \quad j : z_{1j}(t) - c_{1j}(t) < 0\right\}$$

Las expresiones $z_{0j} - c_{0j}$ y $z_{1j} - c_{1j}$ son calculadas en el paso 1 del algoritmo utilizando $a_0(T)$ y $a_1(T)$ respectivamente.

Demostración

Por hipótesis; $z_j(t,\hat{\alpha}) - c_j(t,\hat{\alpha}) \geq 0$

Sea

$$\begin{aligned} &z_j(t,\alpha) - c_j(t,\alpha) = z_{0j}(t) - c_{0j}(t) + \alpha\left[z_{1j}(t) - c_{1j}(t)\right] = \\ &z_{0j}(t) - c_{0j}(t) + \hat{\alpha}\left[z_{1j}(t) - c_{1j}(t)\right] + \left((\alpha - \overline{\alpha}\right)\left[z_{1j}(t) - c_{1j}(t)\right] = z_j(t,\hat{\alpha}) - c_j(t,\hat{\alpha}) + \left(\alpha - \hat{\alpha}\right)\left(z_{1j}(t) - c_{1j}(t)\right) \end{aligned}$$

Se desea que $z_j(t,\alpha) - c_j(t,\alpha) \geq 0$.

Si $z_{1j}(t) - c_{1j}(t) > 0$, entonces $\alpha - \hat{\alpha} \geq -\dfrac{z_j(t,\hat{\alpha}) - c_j(t,\hat{\alpha})}{z_{1j}(t) - c_{1j}(t)}$

$$\alpha \geq \max\left\{-\frac{z_j(t,\hat{\alpha}) - c_j(t,\hat{\alpha})}{z_{1j}(t) - c_{1j}(t)} + \hat{\alpha}\right\} = \max\left\{\frac{-\left(z_j(t,\hat{\alpha}) - c_j(t,\hat{\alpha})\right) + \hat{\alpha}\left(z_{1j}(t) - c_{1j}(t)\right)}{z_{1j}(t) - c_{1j}(t)}\right\} =$$

$$= \max\left\{\frac{-\left[\left(z_{0j}(t) - c_{0j}(t)\right) + \hat{\alpha}\left(z_{1j}(t) - c_{1j}(t)\right)\right] + \hat{\alpha}\left(z_{1j}(t) - c_{1j}(t)\right)}{z_{1j}(t) - c_{1j}(t)}\right\} = \max\left\{-\frac{z_{0j}(t) - c_{0j}(t)}{z_{1j}(t) - c_{1j}(t)}\right\}$$

Definimos entonces; para t = T - 1,...,0.

$$\underline{\alpha}(t) = \max\left\{-\frac{z_{0j}(t) - c_{0j}(t)}{z_{1j}(t) - c_{1j}(t)}; \; j; \; z_{1j}(t) - c_{1j}(t) > 0 \quad\right\}; \; t = T-1,\ldots,0$$

Finalmente tomamos

$$\underline{\alpha} = \max\left\{\alpha_{inf}, \max\left\{\underline{\alpha}(t)\right\} ; t = T-1,...,0\right\}$$

De manera similar, se obtienen $\overline{\alpha}(t)$ y $\overline{\alpha}$ ■

Teorema 2.2

i) Si $\alpha = \overline{\alpha}$ y $\hat{v}_{0B}(t) \le 0,\ t = 0,...,T-1$, entonces $\hat{\mu}^{\theta}_{0B}(t) = \hat{\mu}_{0B}(t) - \theta\hat{v}_{0B}(t)$ es óptima para $\theta \in [0,+\infty[$.

ii) Si $\alpha > \overline{\alpha}$ y $\hat{v}_{0B}(t) \le 0,\ t = 0,...,T-1$, entonces el problema no tiene solución óptima para $\alpha > \overline{\alpha}$.

iii) Si $\hat{v}_{0i}(\tau) > 0$ para al menos un $i \in \{1,...,m\}$ y $\tau \in \{0,...,T-1\}$, con $\alpha = \overline{\alpha}$, entonces puede obtenerse una nueva solución factible básica $\hat{\mu}'_{0B}$, y nuevos valores $z_{0j}(t) - c_{0j}(t)$ y $z_{1j}(t) - c_{1j}(t)$ pueden ser calculados. Por lo tanto, nuevos valores $\underline{\alpha}$ y $\overline{\alpha}$ existen.

Demostración.

i) Por definición, para $\alpha = \overline{\alpha}$, existe un par de índices (k, t_1) para los cuales, $z_k(t_1,\overline{\alpha}) - c_k(t_1,\overline{\alpha}) = 0$. Entonces, se "introduce" en la base la columna $d_k(t_1)$.

A partir del Paso 5 del algoritmo se calculan los vectores $\hat{v}_{0B}(t)$; para todo

i = 0,...,T-1. Los vectores $\hat{v}_{0B}(t)$ nos llevan a la expresión $\hat{\mu}^{\theta}_{0B}(t) = \hat{\mu}^{\theta}_{0B}(t) - \theta\hat{v}^{\theta}_{0B}(t)$ (7)

donde θ es valor asignado a la variable entrante $\mu_k(t_1)$.

Como $\hat{v}_{0B}(t) \le 0$ y $\theta \ge 0$; entonces (7) es no negativa.

ii) Sea $\underline{\alpha} = \overline{\alpha} + \varepsilon$. Se tiene

$$z_k(t_1,\alpha) - c_k(t_1,\alpha) = z_{0j}(t_1) - c_{0j}(t_1) + (\overline{\alpha} + \varepsilon)(z_{1k}(t_1) - c_{1k}(t_1)) =$$

$$z_{0j}(t_1,\overline{\alpha}) - c_{0j}(t_1,\overline{\alpha}) + \varepsilon(z_{1k}(t_1) - c_{1k}(t_1))$$

Del Teorema (2.1) se sabe que $z_{1k}(t_1) - c_{1k}(t_1) < 0$; entonces, $z_k(t_1,\alpha) - c_k(t_1,\alpha) < 0$. Eso implica que $J(\mu,\alpha)$ puede ser mejorada sin perder factibilidad (parte i); así la solución es no acotada.

iii) Como existe $\hat{v}_{0i}(\tau) > 0$; se elige un par de vértices (l,t_2) tales que $\theta_0 = \min\left\{\frac{\hat{\mu}_{0i}(t)}{\hat{v}_{0i}(t)} : \hat{v}_{0i}(t) > 0\right\} = \frac{\hat{\mu}_{0l}(t_2)}{\hat{v}_{0l}(t_2)}$.

Se toma $\hat{v}_{lk}(t_1)$ como pivote y se ejecuta una nueva iteración del algoritmo.

Teorema 2.3

Si $\alpha = \underline{\alpha}$ y $\hat{v}_{0B}(t) \le 0,\ t = 0,...,T-1$, entonces la solución factible básica $\hat{\mu}^{\theta}_{0B}(t) = \hat{\mu}_{0B}(t) - \theta\hat{v}_{0B}(t)$ es óptima para $\theta \in [0,+\infty[$.

Si $\alpha = \underline{\alpha}$ y $\hat{v}_{0i}(\tau) > 0$, para al menos un $i \in \{1,...,m\}$ y un $\tau \in \{0,...,T-1\}$, se puede obtener una nueva solución factible básica, nuevos valores $z_j(t,\alpha) - c_j(t,\alpha)$ y nuevos valores $\underline{\alpha}$ y $\overline{\alpha}$.

Si $\alpha \le \underline{\alpha}$ y $\hat{v}_{0B}(t) \le 0,\ t = 0,...,T-1$, entonces no existe solución óptima finita.

Demostración.

Similar al teorema anterior.
Los resultados anteriores nos llevan al siguiente algoritmo:

3. ALGORITMO

Paso 1. Poner k = 1 y $\alpha_0 = \alpha_{inf}$.

Aplicar el Método Simplex para modelos lineales escalonados (algoritmo dado en la introducción) tomando $a_0(T) + \alpha_0 a_1(T)$ como costo en la función objetivo.

Paso 2. Sea $I = \{(j,t) : z_{1j}(t) - c_{1j}(t) < 0\}$

Si $I = \theta$, poner $\alpha_k = \alpha^{sup}$. Parar.

Paso 3. Calcular

$$\overline{\alpha} = \min \left\{ \alpha^{sup}, \min\left\{\underline{\alpha}(t)\right\}\right\};\ t = T-1,...,0$$

Si $\overline{\alpha} = \alpha_{k-1}$, ir al Paso 5, en otro caso poner $\alpha_k = \overline{\alpha}$ e imprimir $[\alpha_{k-1}, \alpha_k]$.

Si $\alpha_k = \alpha^{sup}$, Parar

Si no, poner k = k + 1.

Paso 4. Sea $I(\overline{\alpha}) = \left\{(j,t) : z_j(t,\overline{\alpha}) - c_j(t,\overline{\alpha}) = 0\right\}$

Sea $\hat{v}_{0B}(t)$ los vectores calculados a partir de los índices dados en el conjunto $I(\overline{\alpha})$.

Si existe un elemento de $I(\overline{\alpha})$ tal que $\hat{v}_{0i}(\tau) \leq 0$; para i = 1,...,m; $\tau = 0,...,T-1$. Parar.

De lo contrario, realizar una iteración por el simplex para matrices escalonadas e ir al paso 2.

Paso 5. Poner $\alpha = \overline{\alpha} + \varepsilon$ y aplicar el algoritmo. Si existe (j, t) con $z_j(t,\alpha) - c_j(t,\alpha) < 0$ y $\hat{v}_{0B}(\tau) < 0$, entonces hacer $\alpha_k = \alpha^{sup}$ y Parar.

CONCLUSIONES

El algoritmo y resultados anteriores muestran que es posible parametrizar el vector costo en un modelo lineal escalonado. El Teorema 2.1 muestra como los valores α(t) que determinarán los extremos del intervalo de estabilidad de la solución son hallados a partir de las expresiones $z_{0j}(t) - c_{0j}(t)$ y $z_{1j}(t) - c_{1j}(t)$; expresiones que se calculan utilizando inversas locales de dimensión m x m y no la inversa del problema que se resuelve.

Los resultados clásicos de parametrización del vector costo pueden ser extendidos de manera natural trabajando con bases locales.

REFERENCIAS

BAZARAA, M.S.; H.D. SHERALI and J. JARVIS (1989): **Linear Programming and Network Flows**. John Wiley & Sons, Inc.

DANTZIG, G.B. (1963): **Linear Programming and Extensions**. Princeton University Press, New Jersey.

FOURER, R. (1982): "Solving staircase linear programs by the simplex method, 1: inversión. 2: Pricing", **Mathematical Programming** 23, 274- 313.

GONZAGA, C.C., B. FEIJOO and A. SÁNCHEZ (1993): **Interior post-optimality techniques for lineal programming: methods of centers**. COPPE- Federal University of Río de Janeiro.

GUDDAT J.; F. GUERRA VASQUEZ; K. TAMMER and K. WENDLER (1985): **Multiobjetive and Stochastic Optimization based on Parametric Optimization**. Akademie-Verlag. Berlin.

KAKES, A. (2001): **Modelos Lineales de Optimización con una Estructura Particular**. Departamento de Matemática Aplicada. Universidad de La Habana.

MONTEIRO, R.D.C. and S. MERHOTRA S. (1996): "A general parametric analysis approach and its implication to sensitivity analysis in interior point methods", **Mathematical Programming** 72.

PROPOI, A. (1988): "Problems of dynamic linear programming". Luxenburgo (Austria) IIASA. **RM** 76-78.