REVISTA DE ECONOMÍA Y ESTADÍSTICA
INSTITUTO DE ECONOMÍA Y FINANZAS
FACULTAD DE CIENCIAS ECONÓMICAS
UNIVERSIDAD NACIONAL DE CÓRDOBA
ISSN 0034-8066

ARTÍCULOS

Optimización No Lineal Restringida: Algoritmos de Resolución

Darío D. Donolo

Revista de Economía y Estadística, Tercera Época, Vol. 19, No. 1-2-3 (1975): 1º, 2º, 3º y 4º Trimestre, pp. 71-153.

http://revistas.unc.edu.ar/index.php/REyE/article/view/3708

OPTIMIZACION NO LINEAL RESTRINGIDA: ALGORITMOS DE RESOLUCION *

DARÍO D. DONOLO

La optimización restringida debe mucho de su avance en la resolución de los modelos a los métodos de cálculo desarrollados en base a la reiteración de pruebas. Es decir el avance teórico en la determinación de las condiciones necesarias y suficientes en Programación No Lineal (Kuhn-Tucker, convexidad, etc.) han transformado, en casos empíricos un problema de optimización en la resolución de un sistema de ecuaciones e inecuaciones bastante difícil de abordar, especialmente cuando las funciones y restricciones son de grados superiores a 1 y relacionadas en forma no aditiva.

Este trabajo, en ese sentido, describe en la Primera Parte el algoritmo de búsqueda de Paviani y Himmelblau para programación no lineal, y provee el programa de cómputo adoptado para un equipo I.B.M. 1130. En forma suscinta entonces, este informe resume los aspectos empíricos más relevantes de la P.N.L., más que los desarrollos teóricos.

En la Segunda Parte se plantea mediante la introducción de ecuaciones convenientes la resolución de problemas de programación entera, discreta, mixta o total y casos en que las variables tienen un dominio limitado. Entendemos que aquí es donde el método de búsqueda prueba su eficacia y permite abordar con éxito problemas que mediante otros algoritmos resultaba dificultoso encarar.

El orden seguido en los capítulos es:

PRIMERA PARTE

1. Concepto de Algoritmos de Búsqueda
2. Algoritmos de la Optimización no Restringida
   2.1. Método de Nelder y Mead o del Poliedro Flexible
   2.2. Método de las secciones Proporcionales

* La Primera Parte de este trabajo se realizó durante el Seminario que sobre Programación no lineal dictó el Prof. Rolando F. Orban.

PRIMERA PARTE

1. CONCEPTO DE ALGORITMOS DE BUSQUEDA

Por algoritmo de "búsqueda" se entiende aquel método de optimización basado en reiteradas pruebas del valor de la función objetivo con valores que satisfaciendo las restricciones, son generados a partir de una solución inicial conveniente según la naturaleza del problema.

Para estos métodos la convergencia absoluta no está demostrada, aun cuando por los resultados obtenidos se asegura eficiencia relativa, medida ésta en tiempo de duración del programa y exactitud de los valores hallados.

Seguidamente se desarrollarán dos algoritmos de "búsqueda" para el caso de optimización no restringida, dado que luego van a ser utilizados en el caso de programación con restricciones.

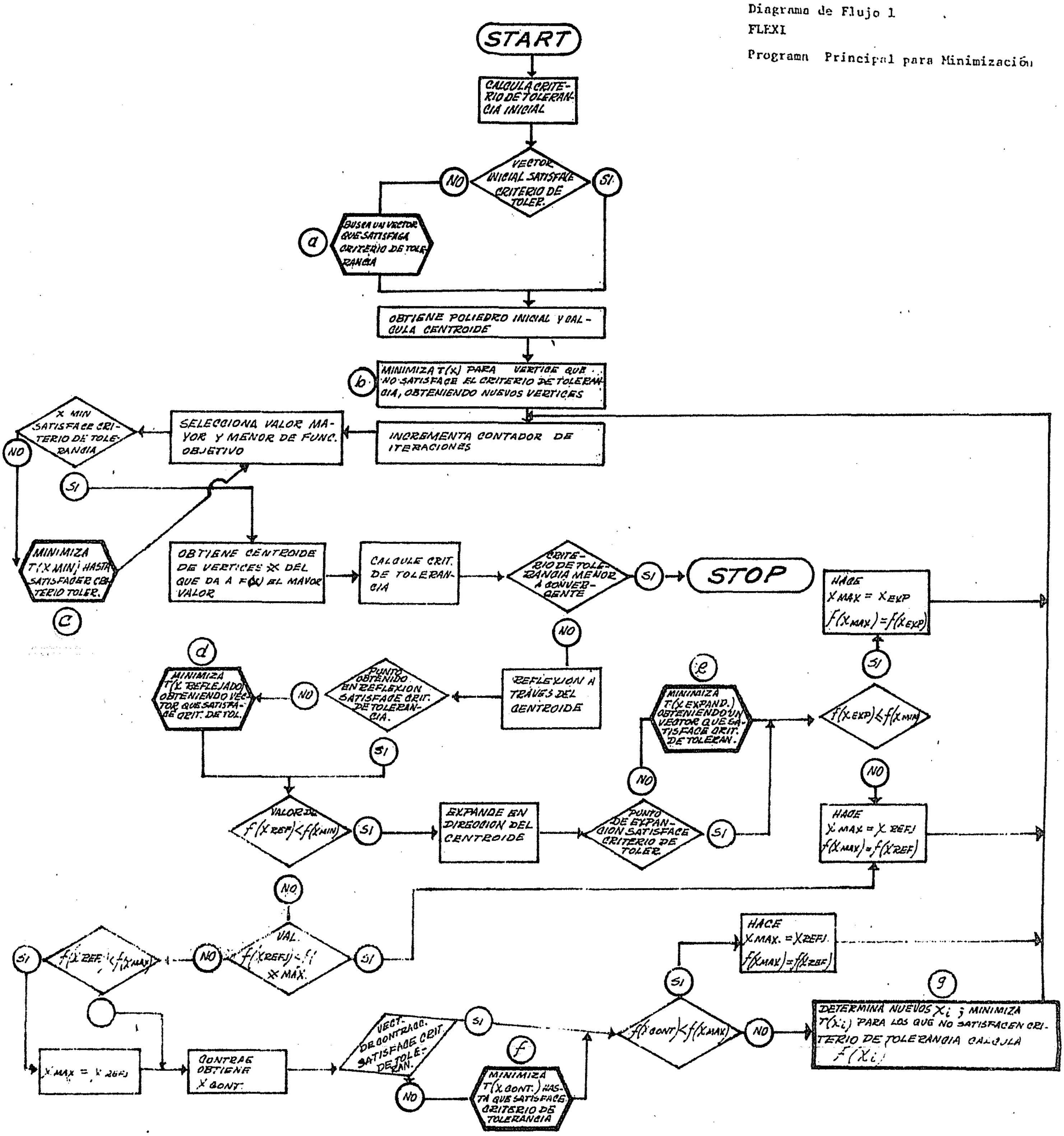
2. ALGORITMOS DE OPTIMIZACION NO RESTRINGIDA

2.2. *Método de Nelder y Mead o del Poliedro Flexible*

Este método minimiza una función de n variables independientes mediante el uso de $(n + 1)$ vértices de un poliedro flexible definido en E^n.

Diagrama de Flujo 1
FLEXI

Programa Principal para Minimización



Escribamos el poliedro en forma matricial, donde las columnas indican las coordenadas y las filas los diferentes puntos.

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2j} & \dots & x_{2n} \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ x_{i1} & x_{i2} & \dots & x_{ij} & \dots & x_{in} \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ x_{n+1,1} & x_{n+1,2} & \dots & x_{n+1,j} & \dots & x_{n+1,n} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_i \\ \cdot \\ \cdot \\ \cdot \\ x_{n+1} \end{bmatrix}$$

Nelder y Mead dieron a este método el nombre de "poliedro flexible" dado que las filas del conjunto X van cambiando de forma de mejorar a la función objetivo. Los extremos (puntos) del poliedro X son datos iniciales que se deben proveer al problema considerando la topología de la función a minimizar.

Usualmente la forma de obtener el poliedro X es por la suma de un vector inicial *I* dado y un poliedro regular *D* (lados iguales) que se calcula a partir de un tamaño lateral *t* dado.

Así:

$$X = I + D$$

donde:

$$I = \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ i_{n+1} \end{bmatrix} = \text{puntos iniciales de cálculo.}$$

$$D = \begin{bmatrix} 0 & d_1 & d_2 & \dots & d_2 \\ 0 & d_2 & d_1 & \dots & d_2 \\ 0 & d_2 & d_2 & \dots & d_2 \\ \cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot\cdot\cdot & \cdot\cdot \\ 0 & d_2 & d_2 & \dots & d_2 \\ \cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot\cdot\cdot & \cdot\cdot \\ 0 & d_2 & d_2 & \dots & d_1 \end{bmatrix} = \text{poliedro regular de } n+1 \text{ puntos, en el espacio n-dimensional.}$$

$$d_1 = \frac{t}{n\sqrt{2}} (\sqrt{n+1} + n - 1)$$

$$d_2 = \frac{t}{n\sqrt{2}} (\sqrt{n+1} - 1)$$

$t =$ distancia entre 2 vértices seguidos cualquiera.

PROCESO DE MINIMIZACIÓN:

Se evalúa la función para cada punto:

$$\begin{bmatrix} f(X_1) \\ f(X_2) \\ \cdot \\ \cdot \\ \cdot \\ f(X_i) \\ \cdot \\ \cdot \\ \cdot \\ f(X_{n+1}) \end{bmatrix}$$

El vértice que arroje máximo valor de $f(X_i)$, $i = 1, 2, \ldots n+1$, lo llamamos X_h y el que arroje el valor mínimo X_e.

Calculamos ahora el centroide (vértice que llamaremos X_{n+2}) de todos los puntos excluído el que da a la función objetivo el valor máximo y que señalamos como X_h.

Así las coordenadas del centroide serán las medias de las columnas de X, excluyendo a X_h.

$$x_{n+2,j} = \frac{1}{n} \underbrace{\left[\left(\sum_{i=1}^{n+1} x_{ij} \right) - x_{hj} \right]} \quad ; \; j = 1, 2, \ldots, n$$

Suma de una columna de la matriz X, incluyendo x_{hj}

Con $(n+2)$ vértices $(X_1\ X_2 \ldots X_h \ldots X_e \ldots X_{n+1},\ X_{n+2})$ vamos a mejorar (minimizar) el valor de la función objetivo generando otro punto que puede surgir de cuatro operaciones:

1) *Reflexión*:

Reflejar X_h a través del centroide X_{n+2}, es obtener otro vértice X_{n+3}, tal que:

$$X_{n+3} = X_{n+2} + \alpha (X_{n+2} - X_h)$$

donde $\alpha > 0$ es el coeficiente de reflexión.

2) *Expansión*

Si $f(X_{n+3}) < f(X_e)$ expandir $(X_{n+3} - X_{n+2})$

en la misma línea del centroide, haciendo

$$X_{n+4} = X_{n+2} + \gamma(X_{n+3} - X_{n+2})$$

para $\gamma > 1$ (coeficiente de expansión).

Si la expansión es buena es decir que $f(X_{n+4}) < f(X_e)$ se reemplaza a X_h por X_{n+4} en el poliedro X original y se comienza nuevamente el proceso de minimización en una iteración siguiente. Si $f(X_{n+4}) > f(X_e)$, reemplazar a X_h por X_{n+3} en el poliedro y reiniciar las etapas nuevamente.

Una visualización de estas dos operaciones (Reflexión y Expansión) las podemos ver en el siguiente gráfico.

Función: $f(X) = f(x_1, x_2)$

Espacio de dos dimensiones; $n = 2$

Número de vértices de trabajo $= n+1 = 3$

$$\text{Vector de vértices} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} X_h \\ X_e \\ X_3 \end{bmatrix}$$

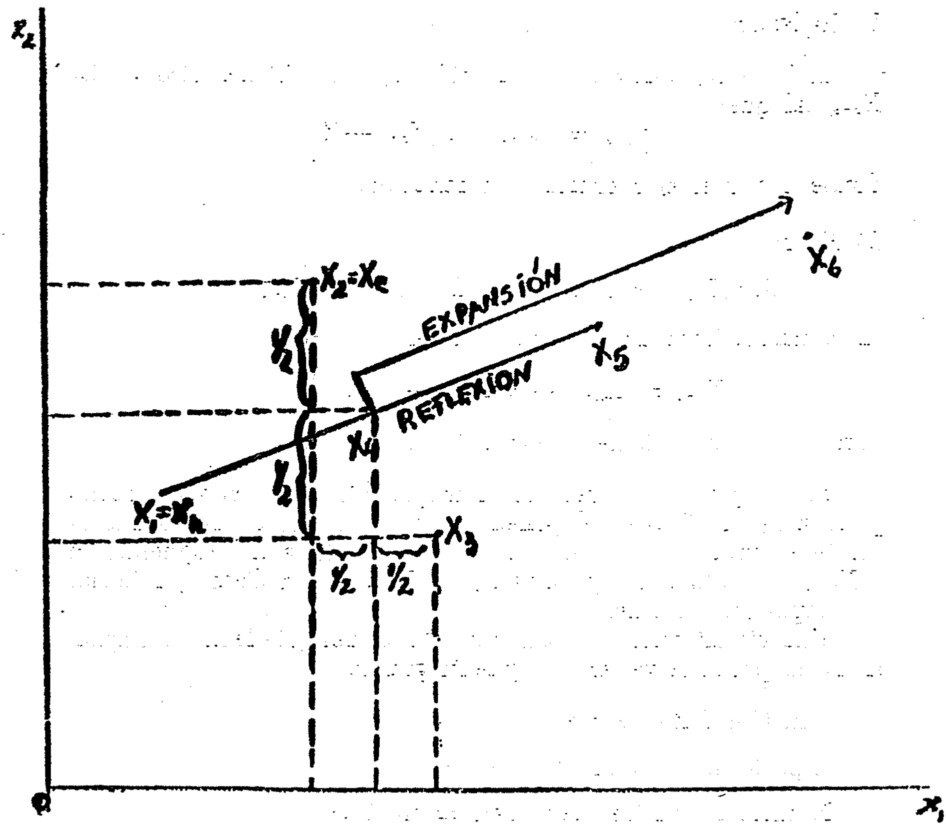
Centroide $= X_{n+2} = X_4$; $\alpha = 1$; $\gamma = 2$

Reflexión: $X_5 = X_4 + 1(X_4 - X_1)$

Expansión: $X_6 = X_4 + 2(X_5 - X_4)$

Si $f(X_6) < f(x_7)$, el poliedro queda:

$$\begin{bmatrix} X_6 \\ X_e \\ X_3 \end{bmatrix}$$



Si $f(X_6) > f(X_2)$, el poliedro queda:

$$\begin{bmatrix} X_5 \\ X_6 \\ X_3 \end{bmatrix}$$

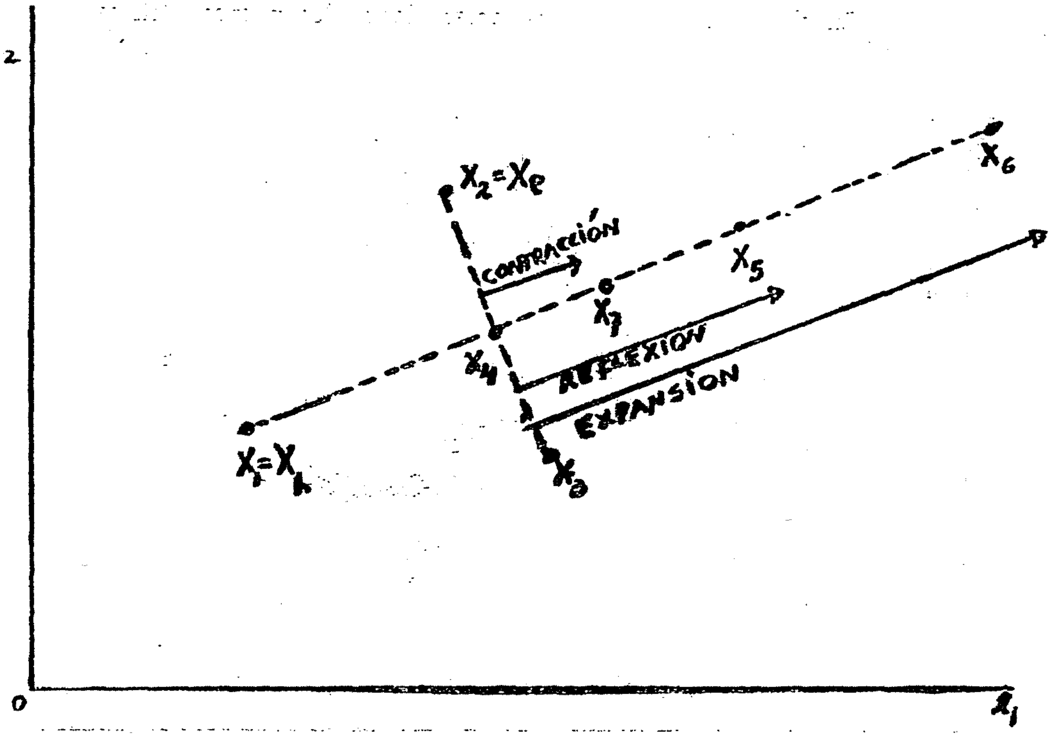
3) *Contracción*

Si $f(X_{n+3})$; $f(X_i)$ $V^{i \neq h}$, se contrae el vector $(X_h - X_{n+2})$ haciendo

$$X_{n+5} = X_{n+2} + \beta(X_h - X_{n+2})$$

$$0 < \beta < 1$$

donde β es el coeficiente de contracción. Se reemplaza a X_h por X_{n+5}, en la Reflexión y se sigue con el proceso desde el principio.



En nuestro ejemplo, si $\beta = 0{,}5$,

$$X_7 = X_4 + 0{,}5\ (X_h - X_4)$$

y el poliedro queda

$$\begin{bmatrix} X_7 \\ X_e \\ X_3 \end{bmatrix}$$

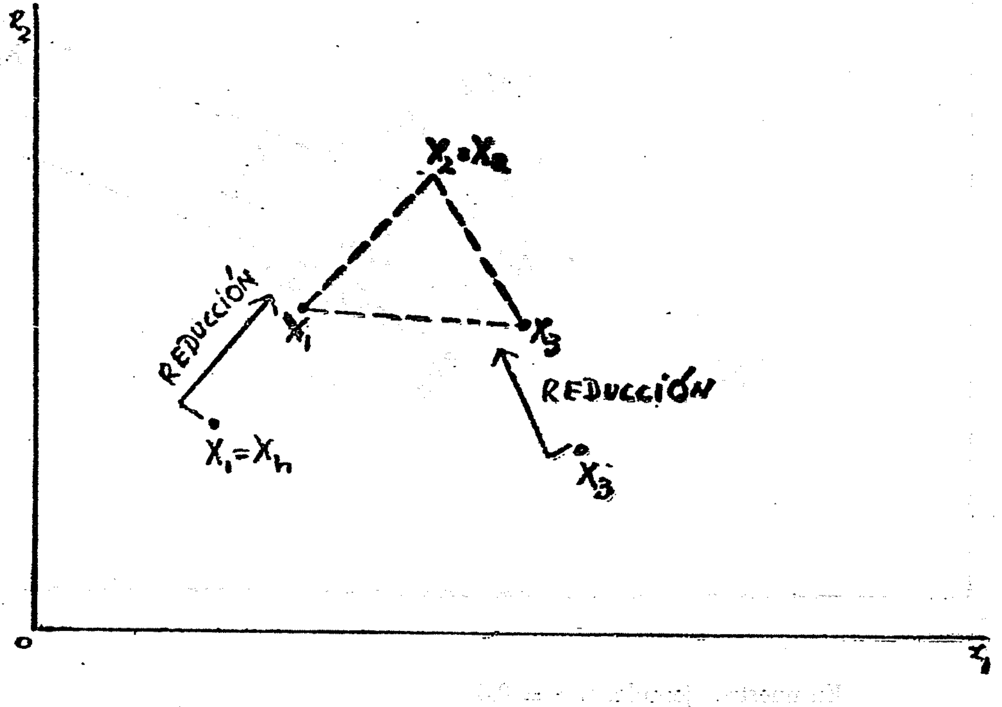
4) *Reducción*

Si $f(X_{n+3}) > f(X_h)$ se reducen todos los vectores $(X_i - X_e)$ para $i = 1, 2, \ldots, n + 1$, por mitad a partir de X_e, haciendo

$$X_i = X_e + 0{,}5\ (X_i - X_e)$$

Luego se reinicia el proceso de optimización desde el principio.

En el ejemplo que se está desarrollando los nuevos vértices serían:



Para terminar el proceso con resultados de aproximación al óptimo local se calcula la media de las distancias entre el valor de la función para cada vértice y el correspondiente al centroide.

El criterio de finalización es:

$$\left\{ \frac{1}{n+1} \sum_{i=1}^{n+1} [f(X_i) - f(X_{n+2})]^2 \right\}^{1/2} \leqslant \epsilon$$

Según se observa en la expresión anterior cuando más pequeño sea el poliedro, luego de un proceso de minimización, se entiende que estamos más cercanos al óptimo dado que ϵ es un número arbitrariamente pequeño.

Finalmente señalemos que el valor de los coeficientes utilizados depende de la naturaleza del problema aunque los más utilizados y recomendados por los autores en base a la experiencia son:

$$\alpha = 1$$
$$\beta = 0{,}5$$
$$\gamma = 2$$

La *convergencia* de este algoritmo se asegura dado que en varias etapas $f(X_{n+3}) > f(X_h)$ como consecuencia de que la reflexión propondrá un punto que excede al mayor de los vectores del poliedro, lo que implica una reducción del poliedro. Al reiterarse este proceso, el poliedro tenderá a convertirse en un punto y entonces todas las reflexiones lo excederán, lo que implicará una nueva reducción, y así sucesivamente, hasta que el criterio de finalización se verifique.

2.2. *Método de las "Secciones Proporcionales"* [1]

Este es un método de búsqueda unidimensional, cuando el proceso de optimización ya esté avanzado. Supongamos que se han ido probando diversos puntos x que han dado $f(x)$ decreciente, hasta que en un punto que llamamos x_3, $f(x)$ vuelve creciente o estacionario. Es decir

$$f(x_3) \geqslant f(x_2)$$

A este nivel se inicia el proceso de minimización a partir de x_3, x_2 y el punto anterior de prueba x_1. El objetivo es reducir el intervalo desde x_1 a x_3 a los efectos de captar el punto mínimo.

Definimos:

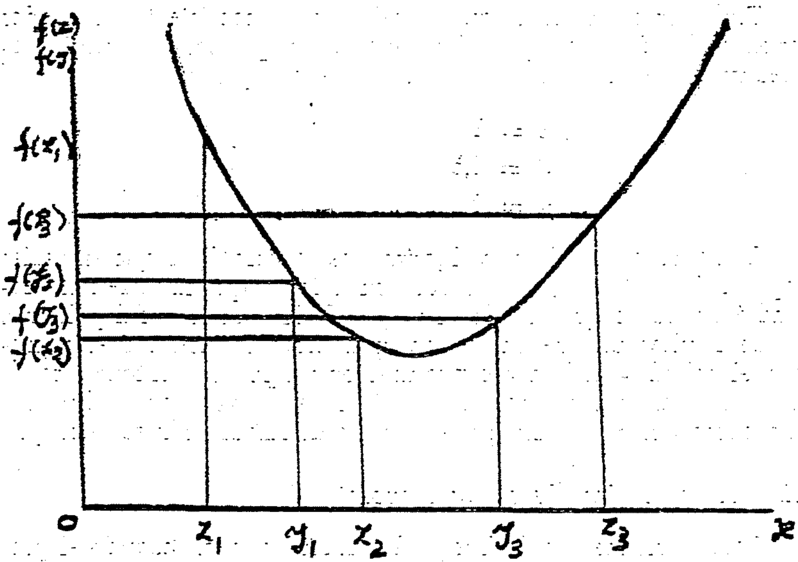
$$\Delta = x_3 - x_1$$

Luego calculamos

$$y_1 = x_1 + F_1 \Delta^{2} = x_1 + 0{,}38\,\Delta$$
$$y_2 = x_1 + F_2 \Delta = x_3 - F_1 \Delta = x_3 - 0{,}38\,\Delta$$

[1] También llamada de "Secciones Medias", como traducción de "Golden Sections".

[2] F_1 y F_2 son las denominadas fracciones de Fibonacci, que indican que todo un intervalo de puntos es el segmento mayor lo mismo que éste es el segmento menor. Es decir $\Delta/F_2 = F_2/F_1 \rightarrow \Delta F_1 = (F_2)^2$ y si $\Delta = 1 \rightarrow F_1 + F_2 = 1$; Fibonacci detectó que estas fracciones podrían ser $F_1 = 0{,}38$ y $F_2 = 0{,}62$.



Ahora se evalúa la función para los puntos y_1, y_2.

Si $f(y_1) < f(y_2)$ reemplazamos a x_3 por y_2 y volvemos la interacción desde el cálculo de .

Si $f(y_1) < f(y_2)$ reemplazamos x_1 por y_1 y volviendo a Δ nuevamente.

Si $f(y_1) = f(y_2)$ podemos reemplazar o bien x_3 por y_2 o bien x_1 por y_1 y retornar al cálculo de Δ y seguir las interacciones.

El proceso se termina cuando $|y_2 - y_1| \leqslant \epsilon$, número arbitrariamente pequeño.[3]

3. TECNICA DE INTERPOLACION CUADRATICA[4]

Sean dos puntos X_1 y X_2 entre los que se desea obtener otro X^*, tal que determinado funcional $\Theta\ [f(X)]$ esté cercano a la frontera de $f(X)$, de forma que $\Theta\ [f(X)]$ sea cercana a cero.

Cualquier punto entre X_1 y X_2 se puede obtener como

$$X = X_2 + \lambda^{(k)} S \qquad \text{para } 0 \leqslant \lambda^{(k)} \leqslant \lambda^*$$

[3] Para ampliar este punto ver Ref. (5) y Ref. (6).
[4] También llamada de "Lagrange Modificada".

donde

k = iteraciones.

$$\lambda^* = \left[\sum_{j=1}^{n} ({}^1x_j - {}^2x_j)^2 \right]^{1/2} = \text{distancia entre}$$

${}^1x_j$ pertenece a X_1
${}^2x_j$ pertenece a X_2

$$S = (X_1 - X_2)/\lambda^*$$

y de aquí

$$X_1 = X_2 + \lambda^* S$$

Calculamos

$$Z_1 = \theta\ [f(X_2)]$$
$$Z_2 = 0\ [f(X_2 + 0{,}5\,\lambda^* S)]$$
$$Z_3 = \theta\ [f(X_1)]$$

de donde Z_1, Z_2 y Z_3 son valores de $\theta[f(X)]$ igualmente espaciados entre los puntos X_2 y X_1. El vector X^* para el que $\theta[f(X)]$ sea aproximadamente nulo es:

$$X^* = X_2 + \frac{B + \sqrt{B^2 - 8AZ}}{4A}\lambda^* S$$

donde

$$A = Z_1 - 2Z_2 + Z_3$$
$$B = 3Z_1 - 4Z_2 + Z_3$$

La expresión anterior se obtiene de una aproximación de segmento de $\theta\ [f(X)]$ en el intervalo definido por λ^*.

4. OPTIMIZACION RESTRINGIDA: METODO DE LA TOLERANCIA FLEXIBLE (M.T.F.)

Dado el problema general de programación no lineal:

$$\text{PROBLEMA (1)} \left\{ \begin{array}{lll} & \text{minimizar } f(x) & \text{para } X \varepsilon E^n \\ & & \\ h_i(x) = 0 & & i = 1, \ldots, m \\ g_i(x) \geqslant 0 & & i = m+1, \ldots, p \\ m \leqslant n & & \end{array} \right. \qquad (1)$$

el M.T.F. combina el Método de Nelder and Mead, el de las "Secciones Proporcionales" y el de "Interpolación" en un nuevo problema no lineal del que se obtendrá una solución similar al señalado en (1).

Este es:

$$\left.\begin{array}{l} \text{minimizar } f(x) \quad \text{para } X \varepsilon E^n \\ \text{con la siguiente restricción:} \\ T(X) \leqslant \varnothing^{(k)} \end{array}\right\} \text{PROBLEMA (2)}$$

donde T(X) es un funcional compuesto por todas las igualdades y desigualdades no satisfechas por el vector que hasta ese momento se haya generado. Así:

$$T(X) = + \left[\sum_{i=1}^{m} h^2_i(X) + \sum_{i=m+1}^{p} U_i g^2_i(X) \right]^{1/2}$$

Cuando una $h_i(X)$ es satisfecha automáticamente se hace igual a cero por definición del problema (1). Si $g_i(X) < 0$, es decir X cumple con la desigualdad, el U_i correspondiente es nulo, y cuando $g_i(X) < 0$, $U_i = 1$.

Es decir, U_i (operador Heaviside) trabaja así:

$$g_i(X) < 0 ===> U_i = 1$$
$$g_i(X) \geqslant 0 ===> U_i = 0$$

En resumen, cuando $T(X) = 0$, vale decir que todas las restricciones se verifican, se ha generado un vector X que es *factible* para recién evaluar a f(X) en ese punto. Es así que gran parte de los algoritmos de búsqueda invierten gran número de iteraciones en generar puntos factibles, lo que va en desmedro de la celeridad en la obtención del resultado.

Ante este problema D. Paviani y D. M. Himmelblau [5] diseñan el M. de la Tolerancia Flexible que prefiere relegar algo de la *factibilidad absoluta* de la solución por otra *cuasi-factible*, que permita incrementar el número de pruebas de la función objetivo y así acercarnos más rápidamente al *óptimo o sub-óptimo* (en nuestro caso mínimo).

[5] Ver Ref. (1).

¿Cuál es el *criterio* de la *cuasi-factibilidad*?

$\varnothing^{(k)}$, *criterio de tolerancia flexible,* es el que determina en el programa (2) si una solución es cuasi factible o *no* factible; de ahí el nombre de criterio de tolerancia. Lo de *flexibilidad* se debe a que el número $\varnothing^{(k)}$ va disminuyendo con cada iteración k de forma que a medida que el algoritmo va avanzando $\varnothing^{(k)}$ tiende a cero, o lo que es lo mismo que cuando:

$$X \longrightarrow X^* \text{ (óptimo)} \Longleftrightarrow \varnothing^{(k)} \longrightarrow 0$$

De otra forma

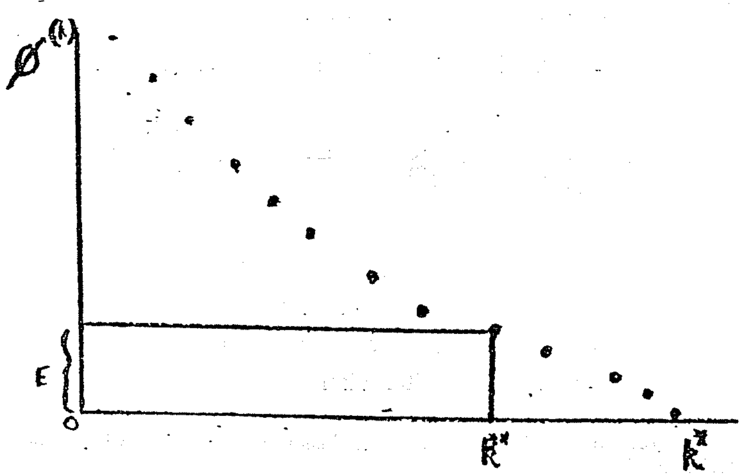
$$\lim_{X \to X^*} \varnothing^{(k)} = 0$$

Así es que el problema (2) se convertirá en el problema (1) a medida que nos acercamos al óptimo.

$$\text{Problema (2)} \begin{cases} f(X^*) \\ T(X^*) = 0 \end{cases} \qquad \text{Problema (1)} \left.\begin{matrix} f(X) \\ h_i(X^*) = 0 \\ g_i(X^*) \geqslant 0 \end{matrix}\right\} = \begin{cases} f(X^*) \\ T(X^*) = 0 \end{cases}$$

El problema de cuán cerca se esté del *óptimo factible* depende del número de iteraciones k realizadas y del *criterio de convergencia* elegido (ε).

Gráficamente:



donde k^{**} corresponde a X^{**} y k^{*} a X^{*}.

ε es un número arbitrario tan pequeño como se quiera que delimite un *sub-óptimo* local X^{**}, del *óptimo local* X^{*}.

Por lo tanto en la finalización del programa se verificará que

$$f(x^{**}) \leqslant f(x^{*} \pm)$$

$$\left[\sum_{i=1}^{m} h_i^2 (X^{**}) + \sum_{j=m+1}^{p} U_i\, g_i^2 (X^{**}) \right]^{1/2} \leqslant \varepsilon$$

Esta última inecuación significa que la suma de las restricciones no satisfechas no supera a ε, lo que implica que cada restricción no puede tampoco ser excedida por más de ε.

Planteado ya el funcionamiento general del M.T.F. queda por determinar:

4.1. Los diferentes valores de $\varnothing^{(k)}$ (criterio de tolerancia).

4.2. Cómo se generan los diversos vectores soluciones X.

4.3. Cómo se convierten los vectores X en soluciones *factibles* o *cuasi factibles* para luego probar f(X)

4.1. *Criterio de Tolerancia*

El criterio de tolerancia $\varnothing^{(k)}$ ya vimos que era una función positiva no creciente; su forma de cálculo es:

$$\varnothing^{(k)} = \min \left\{ \varnothing^{(k-1)}, \theta^{(k)} \right\} \; ; \quad \varnothing^{(0)} = 2\,(m+1)t^{*}$$

$$\theta^{(k)} = \frac{m+1}{m+1} \left\{ \sum_{i=1}^{n+1} \sum_{j=1}^{n} \left(x_{ij}^{(k)} - x_{m+2,j}^{(k)}\right)^2 \right\}^{1/2}$$

donde

t = tamaño del poliedro inicial
m = cantidad de ecuaciones
n = número de coordenadas
k = número de iteración

* El algoritmo también permite la inclusión de un $\theta^{(k)}$ exógeno cuando la función no decrezca.

$x^{(k)}_{n+2,j}$ = coordenada j correspondiente al centroide calculado con $n+1$ vértices

Expliquemos el significado de $\theta^{(k)}$. Supongamos que en una iteración k cualquiera el poliedro con que trabajamos es:

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2j} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{i1} & x_{i2} & \dots & x_{ij} & \dots & x_{in} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{r+1,1} & x_{r+1,2} & \dots & x_{r+1,j} & \dots & x_{r+1,n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{n+1,1} & x_{n+1,2} & \dots & x_{n+1,j} & \dots & x_{n+1,n} \end{bmatrix} = \begin{bmatrix} X_1 \\ X_2 \\ \cdot \\ X_i \\ \cdot \\ X_{r+1} \\ \cdot \\ X_{n+1} \end{bmatrix}$$

De todos los n+1 vértices se calcula el centroide X_{n+2}, de siguientes coordenadas:

$$[x_{n+2,1}\ x_{n+2,2}\ \dots\ x_{n+2,j}\ \dots\ x_{n+2,n}]$$

Luego de hacer:

$$\theta^{(k)} = \frac{m+1}{n+1} \sum_{i=1}^{r+1} \left\| X^{(k)}_i - X^{(k)}_{n+2} \right\|$$

lo que se está implícitamente calculando es una distancia (desviación) media de todos los vértices X_i, al centroide.

Veamos un ejemplo:

Sea una función definida en E^n donde:

n = 4
n+1 = 5, número de vértices con que trabaja el algoritmo.
m = 4, número de ecuaciones

	x_{i1}	x_{i2}	x_{i3}	x_{i4}
i = 1	2	3	4	2
i = 2	3	8	6	1
i = 3	5	3	5	3
i = 4	1	3	3	3
i = 5	1	2	7	1

Cálculo del centroide

$$x_{n+2,j} = \frac{n}{1}\left[\left(\sum_{i=1}^{n+1} x_{i,j}\right) - x_{hj}\right] \quad ; \; j = 1, 2, \ldots, n.$$

x_{hj}	5	8	7	3
$\sum_{i=1}^{n+1} x_{ij}$	12	19	25	10
$\sum_{i=1}^{n+1} x_{ij} - x_{hj}$	7	11	18	7
$x_{n+2,j}$	1,4	2,2	3,6	1,4

Cálculo de θ

$$\theta = \frac{4+1}{4+1}\left\{\sum_{i=1}^{5}\sum_{j=1}^{4}(x_{ij} - x_{n+2,j})^2\right\}^{1/2} = 8.789$$

$\theta^{(k)}$ (positiva) puede aumentar o disminuir según k (iteración), pero $\varnothing^{(k)}$ será siempre una función no creciente dado que siempre es el menor valor de los calculados anteriormente.

Finalmente indiquemos que según avance el algoritmo y dado que los vectores X se generan por el método de Nelder y Mead ya estudiado, la diferencia $(x_{ij} - x_{n+2,j})$ tenderá a cero, de donde $\varnothing \to 0$ lo cual asegura la convergencia hacia una solución mínima local, cualquiera sea el vector inicial elegido [6].

En el trabajo inicial Paviani - Himmelblau [7] propusieron como criterios de tolerancia a un componente:

$$\theta^{(k)} = \frac{1}{(r+1)}\left\{\sum_{i=1}^{r+1}\left(x_{ij}^{(k)} - x_{r+2,j}^{(k)}\right)^2\right\}^{1/2}$$

$$\theta^{(o)} = 2t$$

[6] Ver discusión del tema en sección 5.

[7] Ver referencia (1).

donde:

$r = n - m :=$ grados de libertad de $f(x)$.

En una segunda publicación[8] se hace:

$$\theta^{(k)} = \frac{m+1}{r+1} \left\{ \sum_{i=1}^{r+1} \sum_{j=1}^{n} \left(x_{ij}^{(k)} - x_{r+2,j}^{(k)} \right)^2 \right\}^{1/2}$$

Nuestra observación a ambas expresiones anteriores es que se pierden vértices del poliedro para ser evaluado al reducir su cantidad de $(n+1)$ a $(r+1)$ y que justamente cuando más ecuaciones tiene el problema más puntos son necesarios para asentar el poliedro sobre la topología de la función y del funcional $T(X)$.

4.2. *Generación de diversos vectores soluciones X*

Para la generación de vértices con qué evaluar la función objetivo se utiliza el Método de Nelder y Mead. (Para una rápida comprensión ver el diagrama de flujo del Programa Principal).

Supongamos que todos los vértices que nos provee el algoritmo del poliedro flexible son factibles o cuasi-factibles, es decir que

$$T(X) \leqslant \theta^{(k)} \text{ para todo } k.$$

En ese caso el M.T.F. opera así:

1. Se induce un vector inicial I
2. Se calcula el poliedro regular (la regularidad no es limitativa)
3. $X = I + D$
4. Se calcula el centroide X_{n+2} si $r = n$, o si no X_{r+2}, cuando el problema tiene ecuaciones solamente
5. Se calculan X_h y X_e
6. Se realiza la reflexión
7. Se realiza la expansión o la contracción o la reducción
8. Con el nuevo vector vuelve a 2.

4.3. *Cómo se convierten los vectores X en soluciones factibles o cuasifactibles.*

¿Qué sucede si $T(X) \geqslant \varnothing^{(k)}$? En este caso el vector X no provee una solución factible o cuasi-factible y por lo tanto debemos buscar otra solución para seguir operando.

[8] Ver referencia (2).

El M.T.F. propone minimizar el T(X) no factible hasta

$$T(X) \leqslant \emptyset^{(k)}$$

luego continuar con la operatoria indicada en la sección anterior.

En el diagrama de flujo número 1 en trazo grueso, se denotan las etapas en las que se minimizará T(X); ellas son:

a) después de introducir el vector inicial I
b) cuando hemos obtenido el poliedro
c) después de haber elegido X_e
d) cuando se halla el vector reflejado
e) cuando se halla el vector expandido
f) cuando se halla el vector contraído
g) cuando se reduce el tamaño del poliedro.

¿En qué consiste este proceso de minimizar T(X)? Recordemos primero que:

$$T(X) = \left[\sum_{i=1}^{m} h_i^2(X) + \sum_{i=m+1}^{p} U_i g_i^2(X) \right]^{1/2}$$

$$\text{para} \quad \begin{array}{l} U_i = 0 \text{ si } g_i(X) \geqslant 0 \\ U_i = 1 \text{ si } g_i(X) < 0 \end{array}$$

Supongamos que en la interacción (k) el vector X no satisface a $\emptyset^{(k)}$; es decir:

$$T(X^k) \geqslant \emptyset^{(k)}$$

Para obtener otro vector factible o cuasifactible aplicamos alternativamente tres algoritmos: (ver Diagrama de Flujo 2).

a) el de Nelder y Mead
b) el de Secciones proporcionales
c) el de interpolación.

a) En el primer caso consideramos que T(X) es una función que se debe minimizar, como el planteo general no restringido visto en la sección 3.2.

De este proceso se obtendrá una serie de vectores de los cuales se seleccionará aquel que haga $T(X) \geqslant \varnothing^{(k)}$. Para iniciar el algoritmo de las secciones medias cuando $T(X) < \varnothing^{(k)}$ calculamos:

$$A(S) = \frac{1}{m+1} \left[\sum_{i=1}^{n+1} \{T(X) - T(X_{n+2})\}^2 \right]^{1/2}$$

Si $A(S) > 10^{-6}$ seguimos con el Nelder y Mead, de lo contrario vamos al punto b).

b) Este procedimiento visto en el punto 3.3. realiza la minimización en todas las coordenadas del conjunto de puntos X (Vértices del poliedro), tratando que durante el proceso se determine algún punto factible o cuasi-factible.

Cuando se ha realizado un número conveniente de iteraciones y ningún vector resulta apto para continuar el algoritmo, corresponderá iniciar nuevamente el proceso, cambiando los parámetros del algoritmo de Nelder y Mead (α, γ, β) como así también el vector inicial [9].

c) En los casos en que el programa no lineal se restrinja únicamente por desigualdades, el algoritmo de Nelder y Mead se prueba ineficaz en algunos casos dados que no puede llegar al óptimo, según de la experiencia en casos empíricos. Supongamos como en el gráfico anterior que el óptimo corresponde al punto *0*. Si se da el punto *A* como vértice inicial, luego de la reflexión y contracción estamos en B. El proceso contrario nos llevará a otro punto *C*, factible, pero cercano al vértice inicial A.

Reiterando el proceso se generan nuevos puntos siempre lejanos a 0; la experiencia aconseja entonces realizar una interpolación entre C y *W*, último punto factible de forma de captar un valor próximo a *0*.

Este procedimiento se ha visto en el punto 3, donde aquí

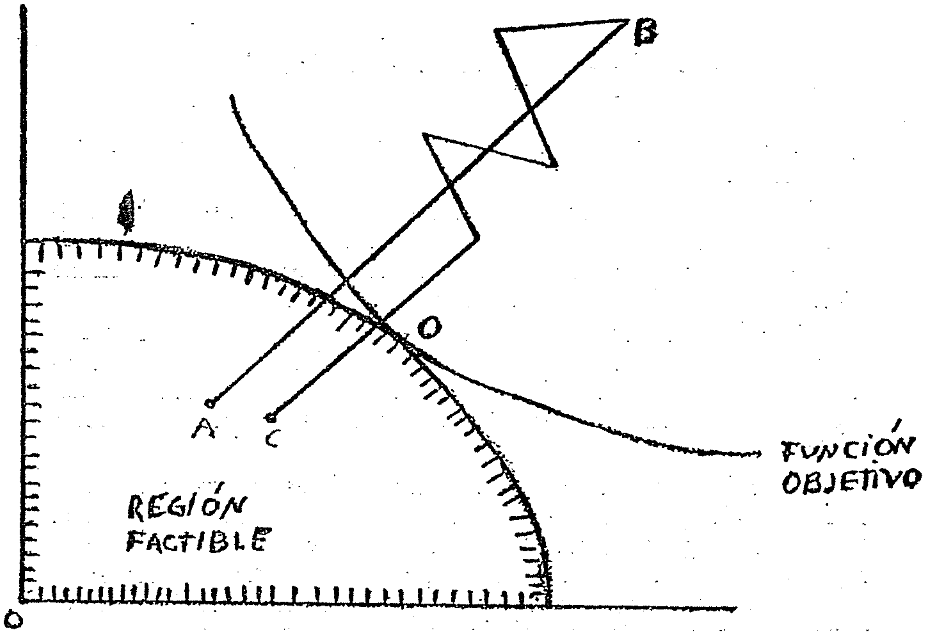
$$\theta[f(x)] = \sum_{i=1}^{p} g_i^2 (X)\, U_i \qquad \text{en el punto } X_1$$

$$W = X_1$$

$$C = X_2$$

[9] La salida indicará que los resultados obtenidos son provisorios.

Como resumen señalemos que la interpolación surge cuando $m = 0$ y $T(X) = 0$.



5. CONVERGENCIA Y CONFIABILIDAD DE LOS RESULTADOS

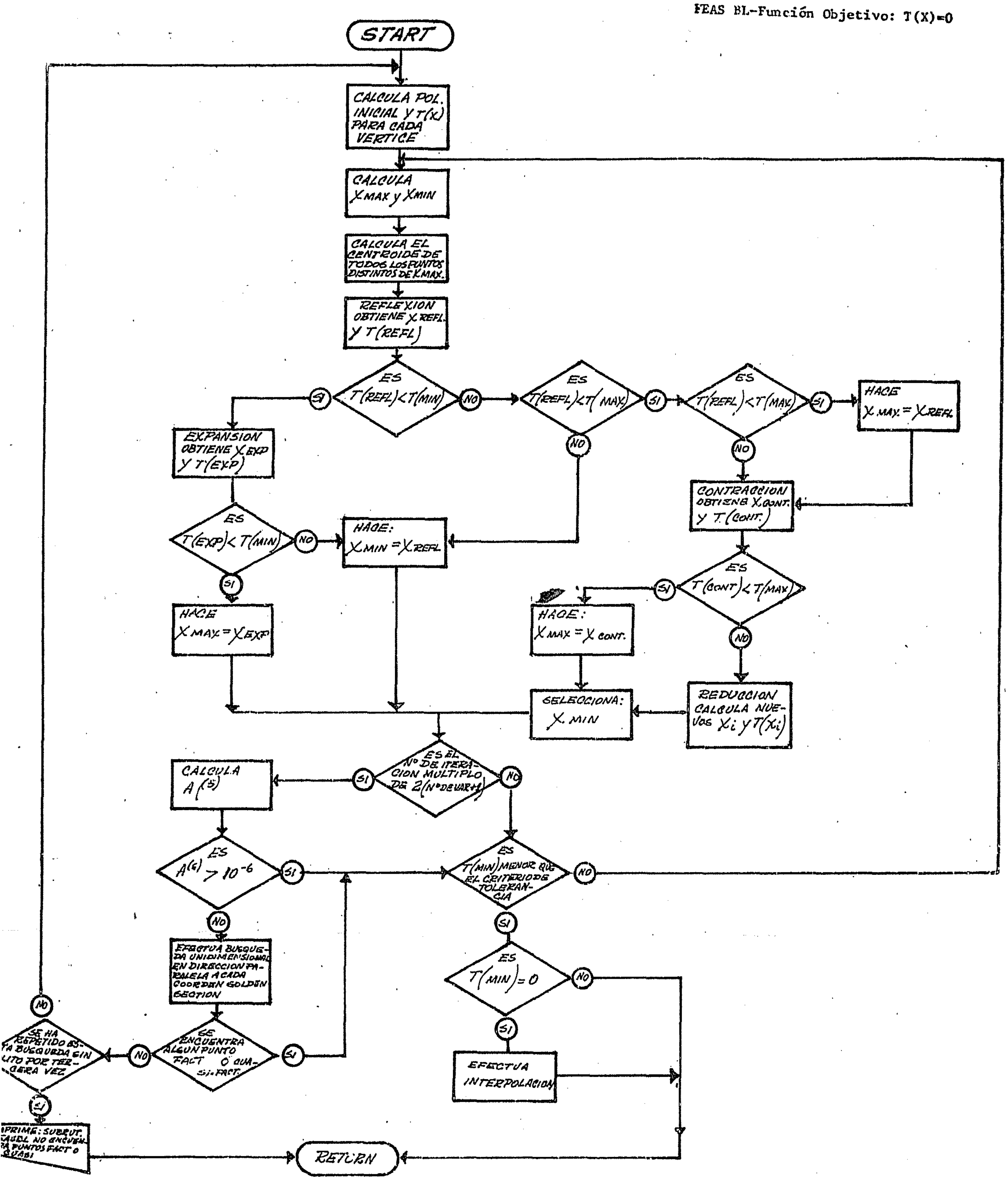
En varias partes del trabajo se ha hecho referencia a la convergencia del Método de la Tolerancia Flexible y a la confiabilidad de los resultados. Abordamos nuevamente aquí el tema a los efectos de clarificar este aspecto.

5.1. Los algoritmos de búsqueda no garantizan convergencia monótona hacia el óptimo. El Método de la Tolerancia Flexible dirige la investigación del vector solución hacia una región que mejora el valor de la función objetivo, aun cuando aquélla no implique que ésta disminuya (en minimización) con cada iteración.

5.2. Si el problema admite varios óptimos locales, con el M.T.F. se las podrá detectar siempre dependiendo del vector inicial elegido. Esto no comporta una falla de los algoritmos de búsqueda dado que

Diagrama de Flujo 2

FEAS BL-Función Objetivo: T(X)=0



la teoría no provee, tampoco, mecanismo para la detección de todos los óptimos locales, excepto en el caso de funciones continuas y diferenciables no restringidas.

En este sentido conviene señalar que la condición suficiente de un óptimo estacionario[10] permite analizar en qué entorno del conjunto factible el vector solución es mínimo o máximo local. Si las condiciones de concavidad y convexidad se mantienen para toda la región de factibilidad, el vector solución determina un óptimo global.

El procedimiento analítico señalado si bien es preciso presenta dificultad en la determinación de un conjunto de puntos factibles para evaluar concavidad o convexidad de la función objetivo y de las restricciones con miras a especificar una solución local o global.

5.3. Con el objeto de establecer un método que dé al investigador de un problema no lineal la seguridad relativa que en el entorno del punto hallado no existe otro óptimo local, es que hemos diseñado un procedimiento heurístico para establecer las cotas máximas y mínimas para el conjunto de puntos generados por el M.T.F., y para cada variable.

Es decir que en el punto *L* (ver en el gráfico) sus coordenadas verifican (en caso de E^2)

$$x_{2m} \leqslant x_{2L} \leqslant x_{2M}$$
$$x_{1m} \leqslant x_{1L} \leqslant x_{1M}$$

donde:

x_{2M} y x_{2m} son respectivamente las cotas máximas y mínimas de variación de x_2 en el algoritmo, del mismo modo, las cotas x_1 son x_{1M} y x_{1m}. Los puntos que determinan las cotas son todos los empleados por el M.T.F., en el programa principal y en las subrutinas. Conociendo la densidad de puntos que provee el algoritmo es muy improbable que en el poliedro 1,2,3,4 (en nuestro ejemplo) pueda existir otro óptimo local de mejor valor para la función objetivo que L. Dicho de otra manera, cualquier otro vector inicial dentro de $\overline{1,2,3,4}$ muy posiblemente determinará L en un número finito de iteraciones.

Otro aspecto que pareciera ser importante dentro de este desarrollo es que si partiendo de puntos externos el poliedro $\overline{1,2,3,4}$, la solución sigue siendo L, se puede ir ampliando el poliedro para indicar, con las reservas apuntadas, que esa es la solución óptima local en ese poliedro.

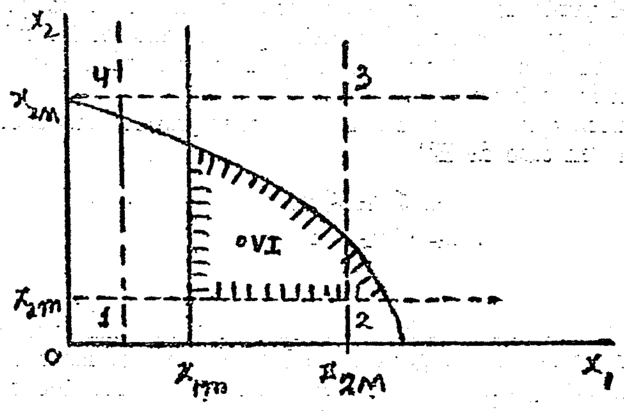
[10] En el capítulo 8.4 se enuncia la condición de suficiencia.

Así el vector $(x_1, x_2, \ldots, x_n)$ es el óptimo local en el poliedro.

$$
\begin{array}{c}
x_{1m} \leqslant x_1 \leqslant x_{1M} \\
x_{2m} < x_2 < x_{2M} \\
\cdot \\
\cdot \\
\cdot \\
x_{nm} \leqslant x_n \leqslant x_{nM}
\end{array}
$$

Este último desarrollo es una propuesta para poder determinar, en principio, un criterio relativo para fijar la validez de la solución hallada.

En casos concretos, el *poliedro de cotas* podría compararse con los campos de variación probables de las variables en cuestión.



5.4. Si el algoritmo falla después de [2x(número de variables + 1)] iteraciones en la sub-rutina del Nelder y Mead y tres veces con el Método de las Secciones Medias, entonces, se determina el vector final no factible hallado.

La falla del algoritmo puede deberse a algunos de los siguientes motivos:

a) Porque el vector inicial esté muy distante de la solución y las iteraciones límites señaladas no permiten avanzar suficientemente en el logro del óptimo. Aquí se recomienda recomenzar el proceso tomando como nuevo vector inicial al último obtenido.

b) Porque el criterio de convergencia (ε) es muy ajustado y la cantidad de iteraciones necesarias para llegar a la solución x que satisfaga dado criterio sea muy grande.

c) Porque los coeficientes α, β, γ sean inadecuados para el caso que se trate. Aquí conviene hacer modificaciones significativas en los parámetros, tomados individualmente, y comprobar la evolución de las soluciones.

d) Porque no exista solución debido a que el conjunto factible sea incompatible o bien no acotado.

El óptimo hallado por este algoritmo no tiene por sí las características de estacionaridad. Es por ello que con el vector solución deben verificarse las condiciones necesarias y suficientes para que el óptimo sea estacionario. En el Capítulo 8 se analiza un ejemplo.

5.5. La buena elección del vector (V. I.) implica evitar muchas iteraciones en la obtención del resultado. En 5.3. se ha visto cómo con diferentes V. I. podríamos obtener varios resultados (óptimos locales) si los mismos existen en el problema propuesto.

Himmelblau propone especificar como V. I. a la media del intervalo de variación que se supone tendrá cada variable según la experiencia que provea los antecedentes del problema.

En muchos casos conviene resolver el sistema de ecuaciones (si es determinado), cuando el problema contenga también inecuaciones, e iniciar el proceso con ese V. I. [11].

Finalmente, recordemos que si el programa desde el comienzo falla en especificar un vector factible, hay que tratar de introducir las ecuaciones en la función de decisión y mantener en las restricciones sólo las inecuaciones.

6. RECOMENDACIONES PARA EL USO DEL PROGRAMA [12]

Para operar con el programa M.T.F. los pasos son:

6.1. Llamar el programa con las siguientes tarjetas (luego que la Subrutina Problem está en máquina):

[11] Debe aclararse que el método también permite resolver sistemas de ecuaciones como una aplicación de la programación no lineal. Ver sección 9.5.

[12] Agradecemos al Sr. Eduardo Aime del Centro de Cómputos su amplia colaboración para modificar y ampliar el programa en FORTRAN.

El Listado del Programa y los Diagramas de Flujo en detalle se encuentran disponibles en el Instituto de Econometría y Estadística.

```
COLD START
//XEQ FLEXI ——— 2
*LOCAL FLEXI, WRITX, INIC, FEAS 1, FEAS 2, FEAS 3,
 FEAS 4, CENTR, CENTR, MAXVL, MINVL, CENTI
*LOCAL FLEXI, SEGMA, EXPAN, CONTR, REDUC
```

6.2. Introducir el problema de la siguiente forma (luego que "Subrutina Problem" está almacenada):

6.2.1. Ficha encabezamiento de columnas 1 a 80 (Tarjeta A)

6.2.2. Parámetros: (Tarjeta B)

Columnas 1 a 5: número de variables totales independientes). (NX).

Columnas 6 a 10: Número de igualdades (NC).

Columnas 11 a 15: número de desigualdades incluyendo las restricciones sobre no negatividad o no positividad de las variables (NIC).

Columnas 16-25: Valor de la magnitud inicial del poliedro; t no debe tener más de 5 decimales.

Columnas 26-35: convergencia deseada; para este equipo no más de 10^{-5}

Columnas 36-45 - ALFA - Si se deja en blanco = 1.

Columnas 46-55 - BETA - Si se deja en blanco = .5

Columnas 56-65 - GAMMA - Si se deja en bco. = 2.

De columnas 1-15 formato 315 y de 16 a 65 formato 5F10.5.; α, β, γ pueden cambiarse sólo en el Programa principal

6.2.3. Vector inicial X (Tarjeta C)

Un valor cada 10 columnas (8F10.5)

6.3. El programa se incorpora en forma de subrutina de la siguiente forma:

```
COLD START
// FOR
ONE WORD INTEGER
SUBROUTINE PROBL (INQ)
COMMON
IF(INR-2) 1, 2, 3
```

```
1. CONTINUE
   (Igualdades en lenguaje FORTRAN)
   GO TO 5
2. CONTINUE
   (Desigualdades en lenguaje FORTRAN)
   GO TO 5
3. CONTINUE
   GO TO 5
   5 RETURN.
```

6.4. Los valores de los parámetros que están en el programa son

$$\alpha = 1$$
$$\beta = 0{,}5$$
$$\gamma = 2$$

para el programa principal y para subrutinas

El criterio de convergencia de A(5) es 10^{-6}.

6.5. El programa minimiza; para maximizar se multiplica la función objetivo por (—1).

6.6. Las variables siempre se inicializan como X(i).

6.7. Conviene iniciarse un proceso con convergencias del 1% y luego ir avanzando, en ε, a partir del valor de solución hallado.

6.8. Para uso didáctico se ha incorporado el uso de llaves que permiten pormenorizar los pasos del algoritmo:

a) Con las llaves todas bajas se imprime una iteración cada 2(NX+1) vectores soluciones.
b) Con la llave 1 levantada se imprimen todas las iteraciones con vectores factibles o cuasi-factibles.
c) Con la llave 2 levantada se imprimen las operaciones (reflexión, expansión, contracción, reducción) del Programa Principal.
d) Con la llave 3 levantada se imprimen todos los pasos de minimización de T(X).
e) Levantar llave 4 ordena leer otro vector inicial del mismo problema. (Tarjeta C)
f) Levantar llave 5 significa formar un nuevo problema. (Tarjetas A, B y C)

Cada llave es independiente de las otras.

6.9. Indiquemos que una vez obtenido el resultado se debe verificar el vector obtenido en las condiciones de Kuhn-Tucker (o de otros autores) y las correspondientes a la concavidad y convexidad de la función y restricciones.[13]

6.10. El criterio de Tolerancia responde a la siguiente expresión:

$$\varnothing^{(k)} = \min\{\,{}^{(r-1)}\varnothing\ \varnothing^{(k)}\}; \quad \varnothing^{(0)} = 2(m+1)t$$

$$\varnothing^{(k)} = \frac{m+1}{n+1}\left\{\sum_{i=1}^{n+1}\sum_{j=1}^{n}\left(x_{ij}^{(k)} - x_{m+2,j}^{(k)}\right)^2\right\}^{1/2}$$

Recordemos, tal como se vio en el punto 4.1 que $\theta^{(k)}$ responde a diferente expresión en el algoritmo original.

6.11. Actualmente el equipo recibe problemas de aproximadamente 8 variables presentadas en forma aditiva. Para cada problema hay que explicar nuevamente la DIMENSION.

6.12. Como el programa no trabaja con EXTENDED PRECISION, muchos límites de convergencia no se verifican por errores de redondeo.

7. EJEMPLO PORMENORIZANDO ITERACIONES

Para mostrar el comportamiento del algoritmo de Paviani y Himmelblau elegimos el siguiente ejemplo: [14]

$$\begin{aligned} \text{Minimizar}\quad f(x) &= 4x_1 - x_2^2 - 12 \\ h_1(x) &= 25 - x_1^2 - x_2^2 \\ g_1(x) &= 10x_1 - x_1^2 + 10x_2 - x_2^2 - 34 \geqslant 0 \\ g_2(x) &= x_1 \geqslant 0 \\ g_3(x) &= x_2 \geqslant 0 \end{aligned}$$

En la subrutina PROBL. se presentaría el caso así:

```
1. CONTINUE
   R(1) = 25—X(1)**2 — X(2)**2
   GO TO 5
```

[13] Ver referencia (3).

[14] Ver referencia (1).

```
2. CONTINUE
   R(2) = 10*X(1) — X(1)** 2 + 10*X(2) — X(2)** 2 — 34
   R(3) = X(1)
   R(4) = X(2)
   GO TO 5.
3. CONTINUE
   R(5) = 4*X(1) — X(2)**2 — 12
   GO TO 5
   5 RETURN
```

Se acompaña seguidamente la solución del ejemplo en orden decreciente de detalle de las iteraciones. (Así decimos dado que la ejecución comienza con las llaves 1, 2 y 3 levantadas y a medida que avanza el algoritmo se las baja, de forma de obtener las soluciones finales de cada iteración).

En el gráfico 1 se presenta la generación de la primera solución cuasi-factible, luego se detalla la evolución de las soluciones desde $k = 1$ hasta $k = 20$. De una visualización del gráfico se puede obtener una idea de la evolución del algoritmo en todos los puntos.

Los puntos del 1 al 50 se indican en el gráfico; desde el 51 al 100 no se señalaron dado que sería dificultosa su indicación. Recién en el punto 100 se obtiene el primer vector factible:

$$x_1 = 3{,}019927 \quad y \quad x_2 = 3{,}981700$$

Desde este vector al 115 se muestra la evolución hacia la solución

$$x_1 = 1{,}001269 \quad y \quad x_2 = 4{,}89721$$

Incluimos seguidamente los resultados obtenidos a partir de diferentes vectores de prueba inicial. Obsérvese que aun cuando éstos sean muy diferenciados arribamos a resultados similares.

PROBLEMA	VECTOR INICIAL		SOLUCION		ITERACION
	x_1	x_2	x_1	x_2	FINAL
7.1	1.	1.	1.001269	4.898721	20
7.2	3.	5.	1.012195	4.896487	19
7.3	7.	3.	1.001523	4.898686	16

PROGRAMA DE *Problema 7.1*
PROGRAMACION NO LINEAL RESTRINGIDA
(MINIMIZACION)

ALGORITMOS UTILIZADOS

POLIEDRO FLEXIBLE (NELDER Y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PRUEBA PROG. NO LINEAL

NUMERO DE VARIABLES INDEPEND. 2
NUMERO DE IGUALDADES 1
NUMERO DE DESIGUALDADES 3
MAGNITUD D/POLIEDRO INICIAL 0.10000E 01 | Llaves 1,2,3
LA CONVERGENCIA DESEADA ES 0.10000-E03 | levantadas
ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.100000E 01 0.100000E 01

EL CRITERIO DE TOLERANCIA INICIAL es 0.400000E 01
LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS ES
0.280178E 02

VALOR DE LA FUNCION OBJETIVO = —0.9000001E 01

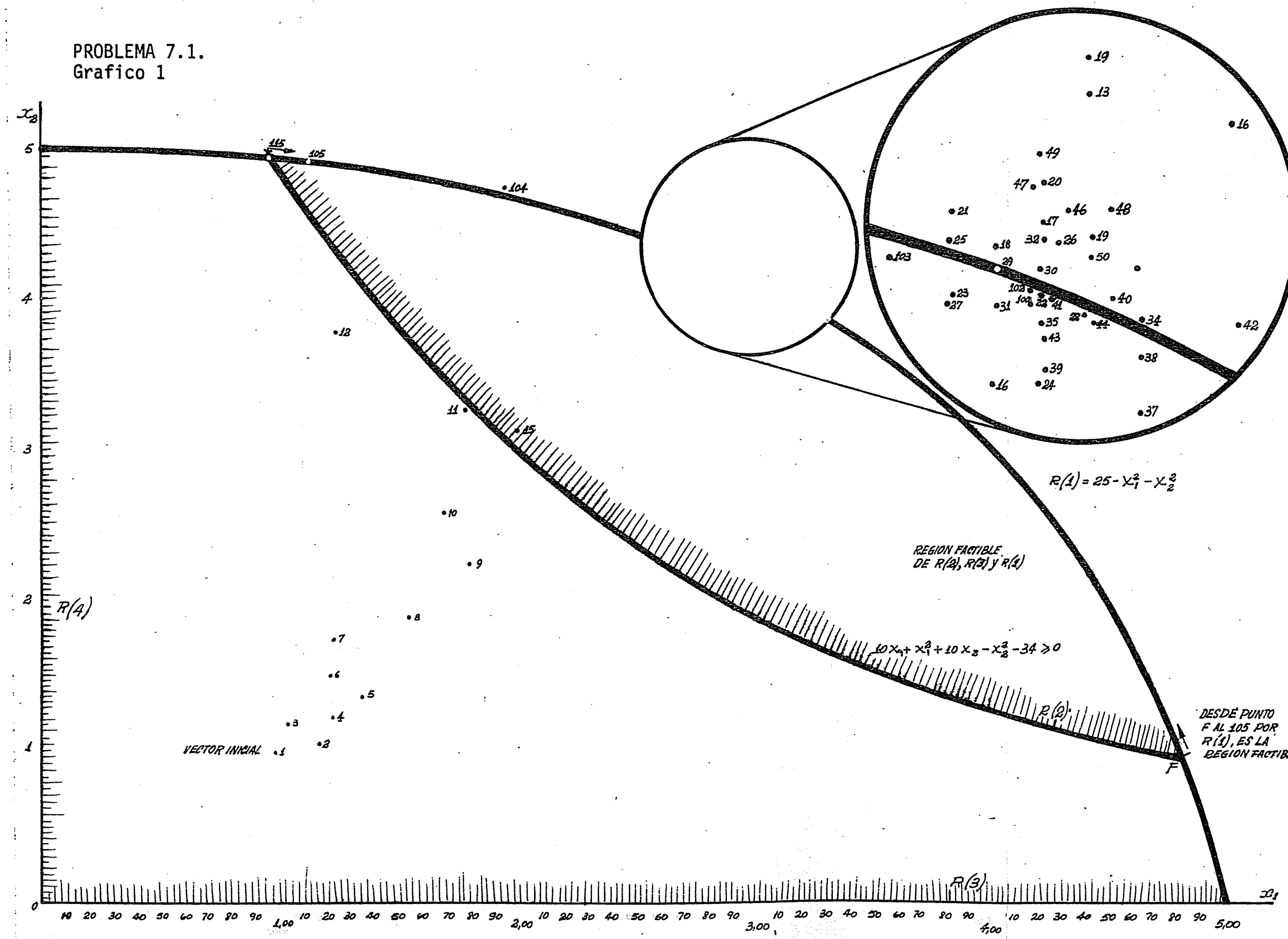
EL VECTOR ES .
0.1000000E 01 0.1000000E 01

LOS VALORES DE LAS IGUALDADES SON
0.2300000E 02

LOS VALORES DE LAS DESIGUALDADES SON
—0.1600000E 02 0.1000000E 01 0.1000000E 01

EL VECTOR INICIAL X, NO SATISFACE EL CRITERIO DE TOLERANCIA INICIAL

PROBLEMA 7.1.
Grafico 1



```
************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL
   0.1000000E 01    0.1000000E 01       (Punto Nº 1)
   0.1193185E 01    0.1051764E 01       (Punto Nº 2)
   0.1051764E 01    0.1193185E 01       (Punto Nº 3)
REFLEXION
   0.1244948E 01    0.1244948E 01       (Punto Nº 4)
EXPANSION
   0.1367422E 01    0.1367422E 01       (Punto Nº 5)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)
   0.1367422E 01    0.1367422E 01
REFLEXION
   0.1226000E 01    0.1508843E 01       (Punto Nº 6)
EXPANSION
   0.1242408E 01    0.1737383E 01       (Punto Nº 7)

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)
   0.1242408E 01    0.1737383E 01
REFLEXION
   0.1558066E 01    0.1911619E 01       (Punto Nº 8)
EXPANSION
   0.1811218E 01    0.2270836E 01       (Punto Nº 9)

VECTOR QUE EN LA ITERACION 3 MINIMIZA T(X)
   0.1811218E 01    0.2270836E 01
REFLEXION
   0.1686203E 01    0.2640796E 01       (Punto Nº 10)
EXPANSION
   0.1845593E 01    0.3277483E 01       (Punto Nº 11)

VECTOR QUE EN LA ITERACION 4 MINIMIZA T(X)
   0.1845593E 01    0.3277483E 01
```

REFLEXION
0.2414402E 01 0.3810936E 01 (Punto Nº 12)

EXPANSION
0.3000393E 01 0.4847713E 01 (Punto Nº 13)

VECTOR QUE EN LA ITERACION 5 MINIMIZA T(X)
0.3000399E 01 0.4847713E 01

REFLEXION
0.3034772E 01 0.5854356E 01 (Punto Nº 14)

CONTRACCION
0.2117106E 01 0.3166716E 01 (Punto Nº 15)

VECTOR QUE EN LA ITERACION 6 MINIMIZA T(X)
0.3000399E 01 0.4847713E 01

CALCULA A(S) E IGUALA CONTADOR DE ITERACIONES A CERO

REFLEXION
0.3271911E 01 0.4736945E 01 (Punto Nº 16)

CONTRACCION
0.2915331E 01 0.4372079E 01 (Punto Nº 17)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)
0.2915331E 01 0.4372079E 01

****************FIN SUBRUTINA FEASBL****************

EL VECTOR ENCONTRADO POR EL PROGRAMA QUE SATISFACE LA TOLERANCIA INICIAL ES
0.291533E 01 0.437207E 01

SUMA DE RESTRICCIONES NO SATISFECHAS = 0.2614230E 01

****

ITERACION NUMERO = 1
CRITERIO DE TOLERANCIA = 0.400000E 01

VALOR DE LA FUNCION OBJETIVO = —0.1945375E 02

EL VECTOR ES
0.2915331E 01 0.4372079E 01

LOS VALORES DE LAS IGUALDADES SON
—0.2614231E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.1125986E 02 0.2915331E 01 0.4372079E 01

POLIEDRO INICIAL

0.2833682E 01	0.4290430E 01	(Punto Nº 18)
0.3026866E 01	0.4342193E 01	(Punto Nº 19)
0.2885445E 01	0.4483614E 01	(Punto Nº 20)

****

ITERACION NUMERO = 2
CRITERIO DE TOLERANCIA = 0.149071E 00

VALOR DE LA FUNCION OBJETIVO = —0.2056101E 02

EL VECTOR ES
0.2885445E 01 0.4483614E 01

LOS VALORES DE LAS IGUALDADES SON
—0.3428588E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.1126199E 02 0.2885445E 01 0.4483614E 01

REFLEXION
0.2692259E 01 0.4431850E 01 (Punto Nº 21)

NO SATISFACE CRITERIO DE TOLERANCIA, PROCEDE A MINIMIZAR T(X)

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL

0.2692259E 01 0.4431850E 01
0.2885444E 01 0.4483613E 01
0.2744023E 01 0.4625035E 01

REFLEXION

0.2833680E 01 0.4290429E 01

EXPANSION

0.2878508E 01 0.4123126E 01 (Punto Nº 22)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)

0.2878508E 01 0.4123126E 01

REFLEXION

0.2685322E 01 0.4071361E 01 (Punto Nº 23)

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)

0.2685322E 01 0.4071361E 01

REFLEXION

0.2871569E 01 0.3762635E 01 (Punto Nº 24)

CONTRACCION

0.2737087E 01 0.4264546E 01 (Punto Nº 25)

VECTOR QUE EN LA ITERACION 3 MINIMIZA T(X)

0.2878508E 01 0.4123126E 01

REFLEXION

0.2930272E 01 0.4316308E 01 (Punto Nº 26)

CONTRACCION

0.2746560E 01 0.4132597E 01 (Punto Nº 27)

VECTOR QUE EN LA ITERACION 4 MINIMIZA T(X)

0.2878508E 01 0.4123126E 01

REFLEXION

0.2887980E 01 0.3901177E 01 (Punto Nº 28)

CONTRACCION

0.2774810E 01 0.4196204E 01 (Punto Nº 29)

VECTOR QUE EN LA ITERACION 5 MINIMIZA T(X)

0.2878508E 01 0.4123126E 01

REFLEXION

0.2906756E 01 0.4186732E 01 (Punto Nº 30)

CONTRACCION

0.2786609E 01 0.4146131E 01 (Punto Nº 31)

VECTOR QUE EN LA ITERACION 6 MINIMIZA T(X)

0.2786609E 01 0.4146131E 01

CALCULA A(S) E IGUALA CONTADOR DE ITERACIONES A CERO

****************FIN SUBRUTINA FEASBL****************

VECTOR ENCONTRADO POR FEASBL

0.2786609E 01 0.4146131E 01

CONTRACCION

0.2943214E 01 0.4364607E 01 (Punto Nº 32)

NO SATISFACE CRITERIO DE TOLERANCIA, PROCEDE A MINIMIZAR T(X)

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL

0.2943214E 01 0.4364607E 01
0.3136399E 01 0.4416371E 01
0.2994978E 01 0.4557792E 01

REFLEXION

0.3084635E 01 0.4223184E 01 (Punto Nº 33)

EXPANSION

0.3129463E 01 0.4055880E 01 (Punto Nº 34)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)
0.3129463E 01 0.4055880E 01

REFLEXION
0.2936277E 01 0.4004117E 01 (Punto Nº 35)

EXPANSION
0.2836216E 01 0.3797989E 01 (Punto Nº 36)

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)
0.2936277E 01 0.4004117E 01

REFLEXION
0.3122526E 01 0.3695389E 01 (Punto Nº 37)

CONTRACCION
0.3077698E 01 0.3862693E 01 (Punto Nº 38)

VECTOR QUE EN LA ITERACION 3 MINIMIZA T(X)
0.2936277E 01 0.4004117E 01

REFLEXION
0.2884512E 01 0.3810928E 01 (Punto Nº 39)

CONTRACCION
0.3068225E 01 0.3994642E 01 (Punto Nº 40)

VECTOR QUE EN LA ITERACION 4 MINIMIZA T(X)
0.2936277E 01 0.4004117E 01

REFLEXION
0.2926802E 01 0.4136063E 01 (Punto Nº 41)

CONTRACCION
0.3039974E 01 0.3931036E 01 (Punto Nº 42)

VECTOR QUE EN LA ITERACION 5 MINIMIZA T(X)
0.3039974E 01 0.3931036E 01

REFLEXION
0.2908024E 01 0.3940508E 01 (Punto Nº 43)

CONTRACCION

0.3028175E 01 0.3981108E 01 (Punto Nº 44)

VECTOR QUE EN LA ITERACION 6 MINIMIZA T(X)

0.3028175E 01 0.3981108E 01

CALCULA A(S) E IGUALA CONTADOR DE ITERACIONES A CERO

****************FIN SUBRUTINA FEASBL****************

VECTOR ENCONTRADO POR FEASBL

0.3028175E 01 0.3981108E 01

REDUCCION

0.2859563E 01 0.4387022E 01 (Punto Nº 45)
0.2956156E 01 0.4412903E 01 (Punto Nº 46)
0.2885445E 01 0.4483614E 01 (Punto Nº 47)

SE MINIMIZA T(X) PARA EL VERTICE 1 DEL POLIEDRO

*************COMIENZA SUBRUTINA FEASBL*************

POLIEDRO INICIAL

0.2859563E 01 0.4387022E 01
0.3052748E 01 0.4438785E 01 (Punto Nº 48)
0.2911327E 01 0.4580206E 01 (Punto Nº 49)

REFLEXION

0.3000983E 01 0.4245600E 01 (Punto Nº 50)

EXPANSION

0.3045811E 01 0.4078297E 01 (Punto Nº 51)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)

0.3045811E 01 0.4078297E 01 (Punto Nº 52)

REFLEXION

0.2852625E 01 0.4026533E 01 (Punto Nº 53)

EXPANSION

0.2752564E 01 0.3820406E 01 (Punto Nº 54)

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)

0.2852625E 01 0.4026533E 01

REFLEXION

0.3038872E 01 0.3717807E 01 (Punto Nº 55)

CONTRACCION

0.2994045E 01 0.3885110E 01 (Punto Nº 56)

VECTOR QUE EN LA ITERACION 3 MINIMIZA T(X)

0.2852025E 01 0.4026533E 01 (Punto Nº 57)

REFLEXION

0.2904391E 01 0.4219717E 01 (Punto Nº 58)

CONTRACCION

0.2971631E 01 0.3968762E 01 (Punto Nº 59)

VECTOR QUE EN LA ITERACION 4 MINIMIZA T(X)

0.2971631E 01 0.3968762E 01

REFLEXION

0.2778444E 01 0.3916996E 01 (Punto Nº 60)

CONTRACCION

0.2978969E 01 0.4037972E 01 (Punto Nº 61)

VECTOR QUE EN LA ITERACION 5 MINIMIZA T(X)

0.2978969E 01 0.4037972E 01 (Punto Nº 62)

REFLEXION

0.3097973E 01 0.3980199E 01 (Punto Nº 63)

VECTOR QUE EN LA ITERACION 6 MINIMIZA T(X)

0.3097973E 01 0.3980199E 01 (Punto Nº 64)

CALCULA A(S) E IGUALA CONTADOR DE ITERACIONES A CERO

REFLEXION

0.2852625E 01   0.4026533E 01   (Punto Nº 65)

CONTRACCION

0.3036636E 01   0.3991782E 01   (Punto Nº 66)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)

0.3036636E 01   0.3991782E 01   (Punto Nº 67)

REFLEXION

0.3043974E 01   0.4060989E 01   (Punto Nº 68)

CONTRACCION

0.2989717E 01   0.3991818E 01   (Punto Nº 69)

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)

0.2989717E 01   0.3991818E 01

****************FIN SUBRUTINA FEASBL****************

VECTOR ENCONTRADO POR FEASBL

0.2989717E 01   0.3991818E 01

POLIEDRO INICIAL

0.2956156E 01   0.4412993E 01   (Punto Nº 70)
0.3149341E 01   0.4464667E 01   (Punto Nº 71)
0.3007919E 01   0.4606088E 01   (Punto Nº 72)

REFLEXION

0.3097576E 01   0.4271480E 01   (Punto Nº 73)

EXPANSION

0.3142405E 01   0.4104176E 01   (Punto Nº 74)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)

0.3142405E 01   0.4104176E 01   (Punto Nº 75)

REFLEXION

0.2949219E 01   0.4052412E 01   (Punto Nº 76)

EXPANSION

0.2849158E 01   0 3846285E 01   (Punto Nº 77)

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)
0.2949219E 01 0.4052412E 01 (Punto Nº 78)

****************FIN SUBRUTINA FEASBL****************

VECTOR ENCONTRADO POR FEASBL
0.2949219E 01 0.4052412E 01

SE MINIMIZA T(X) PARA EL VERTICE 3 DEL POLIEDRO

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL

0.2885445E 01	0.4483614E 01	(Punto Nº 79)
0.3078630E 01	0.4535378E 01	(Punto Nº 90)
0.2937209E 01	0.4676799E 01	(Punto Nº 91)

REFLEXION
0.3026865E 01 0.4342191E 01 (Punto Nº 92)

EXPANSION
0.3071694E 01 0.4174887E 01 (Punto Nº 93)

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)
0.3071694E 01 0.4174887E 01 (Punto Nº 94)

REFLEXION
0.2878508E 01 0.4123124E 01 (Punto Nº 95)

EXPANSION
0.2778447E 01 0.3916996E 01 (Punto Nº 96)

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)
0.2878508E 01 0.4123124E 01 (Punto Nº 97)

REFLEXION
0.3064755E 01 0.3814396E 01 (Punto Nº 98)

CONTRACCION
0.3019927E 01 0.3981700E 01 (Punto Nº 99)

VECTOR QUE EN LA ITERACION 3 MINIMIZA T(X)
0.3019927E 01 0.3981700E 01

****************FIN SUBRUTINA FEASBL****************

VECTOR ENCONTRADO POR FEASBL
0.3019927E 01 0.3981700E 01 (Punto Nº 100)

****

ITERACION NUMERO = 3
CRITERIO DE TOLERANCIA = 0.551173E - 01

VALOR DE LA FUNCION OBJETIVO = 0.1662517E 02

EL VECTOR ES
0.2949219E 01 0.4052412E 01 (Punto Nº 101)

LOS VALORES DE LAS IGUALDADES SON
—0.1199344E 00

LOS VALORES DE LAS DESIGUALDADES SON
0.1089636E 02 0.2949219E 01 0.4052412E 01

REFLEXION
0.2919007E 01 0.4062527E 01

EXPANSION
0.2818086E 01 0.4143353E 01

NO SATISFACE CRITERIO DE TOLERANCIA, PROCEDE A MINIMIZAR T(X)

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL

0.2818086E 01 0.4143353E 01
0.2825285E 01 0.4145281E 01
0.2820015E 01 0.4150552E 01

REFLEXION

0.2823357E 01   0.4138080E 01

EXPANSION

0.2825027E 01   0.4131844E 01

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)

0.2825027E 01   0.4131844E 01

****************FIN SUBRUTINA FEASBL****************

VECTOR ENCONTRADO POR FEASBL

0.2825027E 01   0.4131844E 01

****

ITERACION NUMERO = 4

CRITERIO DE TOLERANCIA = 0.551173E -01

VALOR DE LA FUNCION OBJETIVO = —0.1777202E 02

EL VECTOR ES

0.2825027E 01   0.4131844E 01   (Punto Nº 102)

LOS VALORES DE LAS IGUALDADES SON

—0.5291271E-01

LOS VALORES DE LAS DESIGUALDADES SON

0.1051579E 02   0.2825027E 01   0.4131844E 01

REFLEXION

0.2784529E 01   0.4192436E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL

0.2784529E 01   0.4192436E 01
0.2787190E 01   0.4193148E 01
0.2785242E 01   0.4195097E 01

REFLEXION
0.2786477E 01 0.4190485E 01

EXPANSION
0.2787094E 01 0.4188179E 01

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)
0.2787094E 01 0.4188179E 01

REFLEXION
0.2784431E 01 0.4187466E 01

EXPANSION
0.2783052E 01 0.4184625E 01

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)
0.2783052E 01 0.4184625E 01

REFLEXION
0.2785616E 01 0.4180368E 01

EXPANSION
0.2786159E 01 0.4174334E 01

VECTOR QUE EN LA ITERACION 3 MINIMIZA T(X)
0.2786159E 01 0.4174334E 01

REFLEXION
0.2782116E 01 0.4170779E 01

EXPANSION
0.2779627E 01 0.4162078E 01

VECTOR QUE EN LA ITERACION 4 MINIMIZA T(X)
0.2779627E 01 0.4162078E 01

***************FIN SUBRUTINA FEASBL***************

VECTOR ENCONTRADO POR FEASBL
0.2779627E 01 0.4162078E 01

EXPANSION
0.2564635E 01 0.4301980E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL

0.2564635E 01   0.4301980E 01
0.2567297E 01   0.4302693E 01
0.2565348E 01   0.4304642E 01

REFLEXION

0.2566583E 01   0.4300030E 01

EXPANSION

0.2567201E 01   0.4297724E 01

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)
0.2567201E 01   0.4297724E 01

REFLEXION

0.2564539E 01   0.4297011E 01

EXPANSION

0.2563160E 01   0.4294170E 01

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)
0.2563160E 01   0.4294170E 01

****************FIN SUBRUTINA FEASBL****************

VECTOR ENCONTRADO POR FEASBL
0.2563160E 01   0.4294170E 01

****

ITERACION NUMERO = 5
CRITERIO DE TOLERANCIA = 0.551173E-01

VALOR DE LA FUNCION OBJETIVO = —0.2018725E 02

EL VECTOR ES
0.2563160E 01   0.4294170E 01   (Punto Nº 103)

LOS VALORES DE LAS IGUALDADES SON
—0.9680860E-02

LOS VALORES DE LAS DESIGUALDADES SON
0.9563608E 01 0.2563160E 01 0.4294170E 01

REFLEXION
0.2458909E 01 0.4373600E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL
0.2438969E 01 0.4373600E 01
0.2441630E 01 0.4374313E 01
0.2439682E 01 0.4376262E 01

REFLEXION
0.2440917E 01 0.4371648E 01

EXPANSION
0.2441534E 01 0.4369341E 01

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)
0.2441534E 01 0.4369341E 01

***************FIN SUBRUTINA FEASBL***************

VECTOR ENCONTRADO POR FEASBL
0.2441534E 01 0.4369341E 01

EXPANSION
0.1936414E 01 0.4682011E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

************COMIENZA SUBRUTINA FEASBL************

POLIEDRO INICIAL

0.1936414E 01	0.4682011E 01
0.1939076E 01	0.4682724E 01
0.1937128E 01	0.4684673E 01

REFLEXION

0.1938362E 01 0.4680059E 01

EXPANSION

0.1938979E 01 0.4677752E 01

VECTOR QUE EN LA ITERACION 1 MINIMIZA T(X)

0.1938979E 01 0.4677752E 01

REFLEXION

0.1936316E 01 0.4677038E 01

EXPANSION

0.1934937E 01 0.4674195E 01

VECTOR QUE EN LA ITERACION 2 MINIMIZA T(X)

0.1934937E 01 0.4674195E 01

REFLEXION

0.1937501E 01 0.4669936E 01

EXPANSION

0.1938044E 01 0.4663898E 01

VECTOR QUE EN LA ITERACION 3 MINIMIZA T(X)

0.1938044E 01 0.4663898 01

REFLEXION

0.1934001E 01 0.4660340E 01

EXPANSION

0.1931511E 01 0.4651634 01

VECTOR QUE EN LA ITERACION 4 MINIMIZA T(X)

0.1931511E 01 0.4651634 01

REFLEXION
0.1934618E 01 0.4641335E 01

EXPANSION
0.1934459E 01 0.4624905E 01

VECTOR QUE EN LA ITERACION 5 MINIMIZA T(X)
0.1934459E 01 0.4624905E 01

REFLEXION
0.1927926E 01 0.4612638E 01

EXPANSION
0.1922867E 01 0.4587008E 01

VECTOR QUE EN LA ITERACION 6 MINIMIZA T(X)
0.1927926E 01 0.4612638E 01

CALCULA A(S) E IGUALA CONTADOR DE ITERACIONES A CERO

***************FIN SUBRUTINA FEASBL***************

VECTOR ENCONTRADO POR FEASBL
0.1927926E 01 0.4612638E 01

****

Llaves 1 y 2 levantadas

ITERACION NUMERO = 6
CRITERIO DE TOLERANCIA = 0.551173E-01

VALOR DE LA FUNCION OBJETIVO = —0.2556472E 01

EL VECTOR ES
0.1927926E 01 0.4612638 01 (Punto Nº 104)

LOS VALORES DE LAS IGUALDADES SON
0.6675303E-02

LOS VALORES DE LAS DESIGUALDADES SON
0.6412301E 01 0.1927926E 01 0.4612638E 01

REFLEXION
0.1666058E 01 0.4774064E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.1664103E 01 0.4717858E 01

EXPANSION
0.5012224E 00 0.5246766E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.1148635E 01 0.4869454E 01

****

ITERACION NUMERO = 7
CRITERIO DE TOLERANCIA = 0.551173E-01

VALOR DE LA FUNCION OBJETIVO = —0.3111703E 02

EL VECTOR ES
0.1148635E 01 0.4869454E 01 (Punto Nº 105)

LOS VALORES DE LAS IGUALDADES SON
—0.3093928E-01

LOS VALORES DE LAS DESIGUALDADES SON
0.1149940E 01 0.1148635E 01 0.4869454E 01

REFLEXION
0.5134011E 00 0.5187921E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.1177688E 01 0.4862272E 01

****

ITERACION NUMERO = 8
CRITERIO DE TOLERANCIA = 0.551173E-01

VALOR DE LA FUNCION OBJETIVO = —0.3111703E 02

EL VECTOR ES
0.1148635E 01 0.4869454E 01 (Punto Nº 106)

LOS VALORES DE LAS IGUALDADES SON
—0.3093928E-01

LOS VALORES DE LAS DESIGUALDADES SON
0.1149940E 01 0.1148635E 01 0.4869454E 01

REFLEXION
0.3983975E 00 0.5119087E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.1324794E 01 0.4816591E 01

CONTRACCION
0.1243978E 01 0.4841226E 01

****

ITERACION NUMERO = 9
CRITERIO DE TOLERANCIA = 0.551173E-01

VALOR DE LA FUNCION OBJETIVO = 0.3111703E 02

EL VECTOR ES
0.1148635E 01 0.4869454E 01 (Punto 107)

LOS VALORES DE LAS IGUALDADES SON
—0.3093928E-01

LOS VALORES DE LAS DESIGUALDADES SON
0.1149940E 01 0.1148635E 01 0.4869454E 01

REFLEXION
0.1082345E 01 0.4890499E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.1084911E 01 0.4886242E 01

EXPANSION
0.9284099E 00 0.4927002E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.1007503E 01 0.4899045E 01

ITERACION NUMERO = 10
CRITERIO DE TOLERANCIA = 0.551173E-01

VALOR DE LA FUNCION OBJETIVO = —0.3197061E 02

EL VECTOR ES
0.1007503E 01 0.4899045E 01 (Punto Nº 108)

LOS VALORES DE LAS IGUALDADES SON
—0.1569664E -01

LOS VALORES DE LAS DESIGUALDADES SON
0.4977417E -01 0.1007503E 01 0.4899045E 01

REFLEXION
0.9784499E 00 0.4906225E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.9983010E 00 0.4897311E 01

EXPANSION
0.8387644E 00 0.4923436E 01

NO SATISFACE CRITERIO DE TOLERANCIA,
PROCEDE A MINIMIZAR T(X)

VECTOR ENCONTRADO POR FEASBL
0.1037850E 01 0.4893562E 01

****

ITERACION NUMERO = 11
CRITERIO DE TOLERANCIA = 0.551173E-01

VALOR DE LA FUNCION OBJETIVO = -0.3199044E 02

Llave 1 levantada

EL VECTOR ES
0.9983010E 00 0.4897311E 01 (Punto Nº 109)

LOS VALORES DE LAS IGUALDADES SON
0.1974743E-01

LOS VALORES DE LAS DESIGUALDADES SON
-0.2414703E-01 0.9983010E 00 0.4897311E 01

****

ITERACION NUMERO = 12
CRITERIO DE TOLERANCIA = 0.176942E-01

VALOR DE LA FUNCION OBJETIVO = -0.3199044E 02

EL VECTOR ES
0.9983010E 00 0.4897311E 01 (Punto Nº 110)

LOS VALORES DE LAS IGUALDADES SON
0.1974743E-01

LOS VALORES DE LAS DESIGUALDADES SON
-0.2414703E-01 0.9983010E 00 0.4897311E 01

****

ITERACION NUMERO = 13
CRITERIO DE TOLERANCIA = 0.423521E-02

VALOR DE LA FUNCION OBJETIVO = -0.3199986E 02

EL VECTOR ES
0.9999257E 00 0.4808936E 01 (Punto Nº 111)

LOS VALORES DE LAS IGUALDADES SON
0.5816222E-03

LOS VALORES DE LAS DESIGUALDADES SON
-0.1081848E-01 0.9999257E 00 0.4898936E 01

****

ITERACION NUMERO = 14
CRITERIO DE TOLERANCIA = 0.149946E-02

VALOR DE LA FUNCION OBJETIVO = -0.3199363E 02

EL VECTOR ES
0.1001194E 01 0.4898818E 01 (Punto Nº 112)

LOS VALORES DE LAS IGUALDADES SON
-0.8003116E-03

LOS VALORES DE LAS DESIGUALDADES SON
-0.6942750E-03 0.1001194E 01 0.4898818E 01

****

ITERACION NUMERO = 15

CRITERIO DE TOLERANCIA = 0.182181E-03

Ninguna llave levantada

VALOR DE LA FUNCION OBJETIVO = -0.3199363E 02

EL VECTOR ES
0.1001194E 01 0.4898818E 01 (Punto Nº 113)

LOS VALORES DE LAS IGUALDADES SON
-0.8003116E-03

LOS VALORES DE LAS DESIGUALDADES SON
-0.6942750E-03 0.1001194E 01 0.4898818E 01

****

ITERACION NUMERO = 16
CRITERIO DE TOLERANCIA = 0.141876E-03

VALOR DE LA FUNCION OBJETIVO = -0.3199240E 02

EL VECTOR ES
0.1001262E 01 0.4898720E 01 (Punto Nº 114)

LOS VALORES DE LAS IGUALDADES SON
0.1531839E-04

LOS VALORES DE LAS DESIGUALDADES SON
-0.1678467E-03 0.1001262E 01 0.4898720E 01

NUMERO TOTAL DE ITERACIONES = 20
LIMITE DE CONVERGENCIA = 0.225004E-05

VALOR DE LA FUNCION OBJETIVO = -0.3199238E 02

EL VECTOR ES
0.1001269E 01 0.4898721E 01 (Punto Nº 115)

LOS VALORES DE LAS IGUALDADES SON
-0.5424023E-05

LOS VALORES DE LAS DESIGUALDADES SON
-0.1068115E-03 0.1001269E 01 0.4898721E 01

ESTOS SON RESULTADOS FINALES

PROGRAMA DE
PROGRAMACION NO LINEAL *Problema 7.2*
RESTRINGIDA (MINIMIZACION)

ALGORITMOS UTILIZADOS

POLIEDRO FLEXIBLE (NELDER y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PRUEBA PROG. NO LINEAL

NUMERO DE VARIABLES INDEPENDIENTES 2
NUMERO DE IGUALDADES 1
NUMERO DE DESIGUALDADES 3
MAGNITUD DEL POLIEDRO INICIAL 0.10000E 01
LA CONVERGENCIA DESEADA ES 0.10000E-03

ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.300000E 01 0.500000E 01

EL CRITERIO DE TOLERANCIA INICIAL ES 0.400000E 01
LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS
ES 0.900000E 01

VALOR DE LA FUNCION OBJETIVO = -0.2500000E 02

EL VECTOR ES
0.3000000E 01 0.5000000E 01

LOS VALORES DE LAS IGUALDADES SON
-0.9000001E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.1200000E 02 0.3000000E 01 0.5000000E 01

EL VECTOR INICIAL X, NO SATISFACE EL CRITERIO DE TOLERANCIA INICIAL

EL VECTOR ENCONTRADO POR EL PROGRAMA QUE SATISFACE LA TOLERANCIA INICIAL ES
0.289300E 01 0.443338E 01

SUMA DE RESTRICCIONES NO SATISFECHAS = 0.3024336E 01

****

ITERACION NUMERO = 1
CRITERIO DE TOLERANCIA = 0.400000E 01

VALOR DE LA FUNCION OBJETIVO = -0.2008287E 02

EL VECTOR ES
0.2893002E 01 0.4433382E 01

LOS VALORES DE LAS IGUALDADES SON
-0.3024337E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.1123950E 02   0.2893002E 01   0.4433382E 01

****

ITERACION NUMERO = 12
CRITERIO DE TOLERANCIA = 0.598026E-02

VALOR DE LA FUNCION OBJETIVO = -0.3194561E 02

EL VECTOR ES
0.1008354E 01   0.4896839E 01

LOS VALORES DE LAS IGUALDADES SON
0.4195035E-02

LOS VALORES DE LAS DESIGUALDADES SON
0.5611420E-01   0.1008354E 01   0.4896839E

NUMERO TOTAL DE ITERACIONES = 19
LIMITE DE CONVERGENCIA = 0.974636E-05

VALOR DE LA FUNCION OBJETIVO = -0.3192679E 02

EL VECTOR ES
0.1012195E 01   0.4896487E 02

LOS VALORES DE LAS IGUALDADES SON
-0.1177787E-03

LOS VALORES DE LAS DESIGUALDADES SON
0.8669282E-01   0.1012195E 01   0.4896487E 01

ESTOS SON RESULTADOS FINALES

PROGRAMA DE
PROGRAMACION NO LINEAL
RESTRINGIDA (MINIMIZACION)

ALGORITMOS UTILIZADOS — Problema 7.3

POLIEDRO FLEXIBLE (NELDER y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PRUEBA PROG. NO LINEAL

NUMERO DE VARIABLES INDEPENDIENTES 2
NUMERO DE IGUALDADES 1
NUMERO DE DESIGUALDADES 3
MAGNITUD DEL POLIEDRO INICIAL 0.10000E 01
LA CONVERGENCIA DESEADA ES 0.10000E —03
ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.700000E 01 0.300000E 01

EL CRITERIO DE TOLERANCIA INICIAL ES 0.400000E 01
LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS
ES 0.330000E 02

VALOR DE LA FUNCION OBJETIVO = 0.7000000E 01

EL VECTOR ES
0.7000000 01 0.3000000E 01

LOS VALORES DE LAS IGUALDADES SON
-0.3300000E 02

LOS VALORES DE LAS DESIGUALDADES SON
0.8000001E 01 0.7000000E 01 0.3000000E 01

EL VECTOR INICIAL X, NO SATISFACE EL CRITERIO DE TOLERANCIA INICIAL

EL VECTOR ENCONTRADO POR EL PROGRAMA QUE SATISFACE LA TOLERANCIA INICIAL ES

0.405154E 01   0.287119E 01

SUMA DE RESTRICCIONES NO SATISFECHAS =

0.3412541E 00

****

ITERACION NUMERO = 1
CRITERIO DE TOLERANCIA = 0.400000E 01

VALOR DE LA FUNCION OBJETIVO = -0.4037550E 01

EL VECTOR ES

0.4051546E 01   0.2971190E 01

LOS VALORES DE LAS IGUALDADES SON

0.3412542E 00

LOS VALORES DE LAS DESIGUALDADES SON

0.1056859E 02   0.4051546E 01   0.2871190E 01

****

ITERACION NUMERO = 12
CRITERIO DE TOLERANCIA = 0.210256E-02

VALOR DE LA FUNCION OBJETIVO = -0.3199562E 02

EL VECTOR ES

0.1002514E 01   0.4899559E 01

LOS VALORES DE LAS IGUALDADES SON

-0.1071393E-01

LOS VALORES DE LAS DESIGUALDADES SON

0.1000976E-01   0.1002514 01   0.4899559E 01

NUMERO TOTAL DE ITERACIONES = 16
LIMITE DE CONVERGENCIA = 0.967333E-04

VALOR DE LA FUNCION OBJETIVO = -0.3190102E 02

EL VECTOR ES
0.1001523E 01 0.4898686E 01

LOS VALORES DE LAS IGUALDADES SON
0.1708865E-03

LOS VALORES DE LAS DESIGUALDADES SON
-0.1914978E-02 0.1001523E 01 0.4898686E 01

ESTOS SON RESULTADOS FINALES

8. CONDICIONES DE OPTIMIDAD ESTACIONARIA EN PROGRAMACION NO LINEAL

En esta sección enunciaremos las condiciones de optimidad siguiendo a Mangasarian,[15] incluso en la simbología. Los pasos a seguirse son:

8.1. *Obtención de un vector solución* $\bar{x}$ al problema 2.2. a través de algún método (v.g. el M.T.F.), tal que:

8.2. *Problema*

8.2.1. *Optimo global estacionario*

$$\theta(\bar{x}) = \min_{x \varepsilon X} \theta(x)$$

$$\bar{x} \varepsilon X = \{ x \mid x \varepsilon X^o \ , \ g(x) \leqslant 0, \ h(x) = 0 \}$$

donde

$\bar{x}$ = vector solución
X = conjunto factible
$X^o \subset R^n$

[15] Ver referencia (3).

8 2.2. *Optimo local estacionario*

$$\theta(\bar{x}) = \min_{x \varepsilon X} \theta(x)$$

$$\bar{x} \varepsilon X \cap B_\delta(\bar{x})$$

donde $B_\delta(\bar{x})$ es un conjunto abierto alrededor de $\bar{x}$ con radio δ.

Las condiciones siguientes se verifican si θ, h y g son diferenciables en $\bar{x}$.

8.3. *Condición necesaria para óptimo estacionario* [16]

Si g y h satisfacen la calificación de las restricciones de Kuhn Tucker en $\bar{x}$, entonces existirá un $\bar{u} \varepsilon R^m$ y $\bar{v} \varepsilon R^k$ tal que el Problema Estacionario de Kuhn Tucker tenga solución.

8.3.1. *Calificación de las Restricciones según Kuhn Tucker*

g y h satisfacen esta calificación de punto solución extremo si:

$$\left.\begin{array}{l} y \varepsilon R^n \\ \nabla g_I(\bar{x})\, y \leqslant 0 \\ \nabla h(\bar{x})\, y = 0 \end{array}\right\} \longrightarrow \left\{\begin{array}{l} \text{Existe un vector funcional n-dimensional } e \text{ en el intervalor } [0,1] \text{ tal que:} \\ \text{a.} \; e(0) = \bar{x} \\ \text{b.} \; e(\tau) \varepsilon X \text{ para } 0 \leqslant \tau \leqslant \\ \text{c.} \; e \text{ es diferenciable en } \tau = 0 \text{ y} \\ \qquad \dfrac{de(0)}{d\tau} = \lambda\, y, \text{ para } \lambda > 0 \end{array}\right.$$

donde:

$$I = \{ i \mid g_i(\bar{x}) = 0 \}$$

8.3.2. *Problema Estacionario según Kuhn Tucker*

$$\begin{aligned} \nabla\theta(\bar{x}) + \bar{u}\nabla g(\bar{x}) + \bar{v}\nabla h(\bar{x}) &= 0 \\ g(\bar{x}) &\leqslant 0 \\ h(\bar{x}) &= 0 \\ \bar{u} g(\bar{x}) &= 0 \\ u &\geqslant 0 \end{aligned}$$

[16] El óptimo estacionario se verifica siendo $L(x,u,v) = \theta(x) + ug(x) + vh(x)$, cuando $L(\bar{x}, u, v) \leqslant L(\bar{x}, \bar{u}, \bar{v}) \leqslant L(x, \bar{u}, \bar{v})$.

8.4. *Condición suficiente para un óptimo estacionario*

Si | θ es seudoconvexa en $\bar{x}$
| g_I es cuasiconvexa en $\bar{x}$
| h = es cuasiconvexa y cuasi cóncava en $\bar{x}$, se verifica

la condición suficiente para que $\bar{x}$ sea el óptimo del problema 8.2.

8.4.1. *Condición de seudoconvexidad*

$$\theta(x^2) < \theta(x') \longrightarrow \nabla\theta(x')\ (x^2 - x') < 0$$

ó

$$\theta(x^2) \geqslant \theta(x') \longleftarrow \nabla\theta(x')\ (x^2 - x') \geqslant 0$$

8.4.2. *Condición de cuasiconvexidad*

$$\theta(x^2) \leqslant \theta(x') \longrightarrow \nabla\theta(x')\ (x^2 - x') \leqslant 0$$

ó

$$\theta(x^2) \geqslant \theta(x') \longleftarrow \nabla\theta(x')\ (x^2 - x') > 0$$

8.4.3. *Condición de cuasiconcavidad*

$$\theta(\bar{x}) \geqslant \nabla\theta(x') \longrightarrow \nabla\theta(x')\ (x^2 - x') \geqslant 0$$

ó

$$\theta(x^2) < \theta(x') \longleftarrow \nabla\theta(x')\ (x^2 - x') < 0$$

Para el ejemplo de la sección 7, reiteraremos los pasos anteriormente enunciados.

8.1. y 8.2. *Resolución del problema*

$$\begin{aligned}
\text{Min } \theta(x) &= 4x_1 - x_2^2 - 12 \\
h_1(x) &= 25 - x_1^2 - {}_2^2 \\
g_1(x) &= -10x_1 + x_1^2 - 10x_2 + x_2^2 + 34 \leqslant 0 \\
g_2(x) &= -x_1 \leqslant 0 \\
g_3(x) &= -x_2 \leqslant 0
\end{aligned}$$

Según el Método de la Tolerancia Flexible

$\bar{x} = [1{,}001269 \quad 4{,}898721]$ con una aproximación de 0,001

Para simplificar los cálculos

$\bar{x} = [1 \quad 4{,}9]$, de donde

$$\theta(\bar{x}) = -32{,}01$$

$$\left.\begin{aligned}
h_1(\bar{x}) &= -0{,}01 \cong 0 \\
g_1(\bar{x}) &= 0{,}01 \cong 0
\end{aligned}\right\}\ \text{restricciones activas}$$

$$g_2(\bar{x}) = -1$$
$$g_3(\bar{x}) = -4{,}9$$
θ, h y g son diferenciables en x

8.3. *Condición Necesaria para óptimo estacionario*

8.3.1. *Calificación de las restricciones según Kuhn Tucker*

$$\begin{cases} \nabla\, g_1(x)\; y \leqslant 0 \\ \nabla\, h_1(\bar{x})\; y = 0 \end{cases}$$

$$\begin{cases} [-10 + 2\bar{x}_1 \quad -10 + 2\bar{x}_2] \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \leqslant 0 \\ \\ [-2\bar{x}_1 \quad -2\bar{x}_2 \quad] \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = 0 \end{cases}$$

$$\begin{cases} (-10 + 2\bar{x}_1)y_1 + (-10 + 2\bar{x}_2)y_2 \leqslant 0 \\ -2\bar{x}_1 y_1 \quad -2\bar{x}_2 y_2 = 0 \end{cases}$$

$$\begin{cases} -8y_1 \quad -0{,}02y_2 \qquad 0 \\ -2y_1 \quad -9{,}8y_2 = 0 \end{cases}$$

La implicancia de la calificación de la restricción determina que siendo:

$$e = \begin{bmatrix} x_1 + 1 \\ [25 - (x_1 + 1)^2]^{1/2} \end{bmatrix}$$

por ser

$$x_2 = (25 - x_1^2)^{1/2}$$

$$\star\ e(0) = \bar{x} = [1 \quad 4{,}9]$$

$$\star\ e(\tau)\varepsilon X \quad \text{para} \quad 0 \leqslant \tau \leqslant 1$$

$$\star\ \frac{de(\tau)}{d\tau} = \begin{bmatrix} 1 \\ 1/2\ [(25 - (\tau + 1)^2]^{-1/2}\, 2\,(\tau + 1)\,(-1) \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ [25-(\tau+1)^2]^{-1/2}\,[-(\tau+1)] \end{bmatrix}$$

para $\tau = 0$

$$\frac{de\,(\tau=0)}{d\,\tau} = \begin{bmatrix} 1 \\ -\dfrac{1}{\sqrt{24}} \end{bmatrix} = \begin{bmatrix} 1 \\ -\dfrac{1}{4{,}9} \end{bmatrix} = \lambda \begin{bmatrix} -1 \\ -\dfrac{-1}{4{,}9} \end{bmatrix}$$

se verifica para $\lambda > 0$.

Entonces $\bar{x} = [1 \;\; 4{,}9]$, califica las restricciones activas, con lo que se determina que el *Problema Estacionario* de Kuhn Tucker tiene solución.

En efecto:

$$\begin{aligned}
8.3.2\quad & 4+\bar{\mu}_1(-10+2\bar{x}_2)+\bar{\mu}_2(-1)+\bar{v}_1(-2\bar{x}_1) &= 0 \quad & a)\\
& -2\bar{x}_2+\bar{\mu}_1(-10+2\bar{x}_2)+\bar{\mu}_3(-1)+\bar{v}_1(2\bar{x}_2) &= 0 \quad & b)\\
& \bar{\mu}_1(-10\bar{x}_1+\bar{x}^2_1-10\bar{x}_2+\bar{x}^2_2+34)+\bar{\mu}_2(-\bar{x}_1)+\bar{\mu}_3(-\bar{x}_2) &= 0 \quad & c)\\
& -10\bar{x}_1+\bar{x}^2_1-10\bar{x}_2+x^2_2+34 &\leqslant 0 \quad & d)\\
& -x_1 &\leqslant 0 \quad & e)\\
& -x_2 &\leqslant 0 \quad & f)\\
& \mu_1,\ \mu_2,\ \mu_3 &\geqslant 0 \quad & g)
\end{aligned}$$

Para $\bar{x}_1 = 1$ y $\bar{x}_2 = 4{,}9$

$$\begin{aligned}
4 - 8\mu_1 - \bar{\mu}_2 - 2\bar{v}_1 &= 0 \quad & a)\\
-9{,}8-0{,}02\mu_1 - {}_3 - 9{,}8\,\bar{v}_1 &= 0 \quad & b)\\
-\bar{\mu}_2 - 4{,}9\bar{\mu}_3 &= 0 \quad & c)\\
\bar{\mu}_1,\ \bar{\mu}_2,\ \bar{\mu}_3 &\geqslant 0 \quad & g)
\end{aligned}$$

Verificando el sistema anterior:

$$\bar{\mu}_2 = \bar{\mu}_3 = 0;\ \bar{v}_1 = 1{,}00152;\ \mu_1 = 0{,}75038$$

8.4. *Condición suficiente para óptimo estacionario*

8.4.1. *θ es seudoconvexa en $\bar{x}$*

$$x^2 = [1 \quad 4{,}9]$$
$$x^1 = [x_1 \quad x_2]$$
$$\theta(x^2) \geqslant \theta(x^1) \longleftarrow (x^1)(x^2 - x^1) > 0$$
$$\theta(x^2) = 4 \cdot 4{,}9^2 - 12 = -32{,}01$$

$$-32{,}01 \geqslant 4x_1 - x_2^2 - 12 \longleftarrow [4 \quad -2x_2] \begin{bmatrix} 1 - x_1 \\ 4{,}9 - x_2 \end{bmatrix} > 0$$

$$-32{,}01 \geqslant 4x_1 - x_2^2 - 12 \longleftarrow 4(1 - x_1) + (-2x_2)(4{,}9 - x_2) > 0$$

$$\boxed{-32{,}01 \geqslant 4x_1 - x_2^2 - 12 \longleftarrow 4 - x_1 - 9{,}8x_2 + 2x_2^2 > 0}$$

Los puntos para los que debe evaluarse la expresión anterior corresponden a la igualdad $h_1(x) = 25 - x_1^2 - x_2^2 = 0$, de donde ejemplificativamente

x_1	1	2	3	4	4,9
x_2	4,9	4,6	4	3	1

Las cotas de x_1 y x_2 están determinadas por $g_1(x)$ para $h_1(x)$, ($1 \leqslant x_1 \leqslant 4{,}9$ y $4{,}9 \leqslant x_2 \leqslant 1$), región de factibilidad del problema 8.2.

8.4.2. g_1 *es cuasiconvexa en $\bar{x}$*

g_I es cuasiconvexa si:

$$g_I(x^2) \leqslant g_I(x^1) \longrightarrow \nabla g_I(x^1)(x^2 - x^1) \leqslant 0$$
$$x^2 = [1 \quad 4{,}9]$$
$$x^1 = [x_I \quad x_2]$$
$$g_I(x^2) = 0{,}01 \cong 0$$
$$g_I(x^1) = 10x_1 + x_1^2 - 10x_2 + x_2^2 + 34 \leqslant 0$$

$$\nabla g_I(x^1) = [-10 + 2x_1 \quad -10 + 2x_2]$$
$$(x^2 - x^1) = [1 - x_1 \quad 4{,}9 - x_2]^T$$

$$0 \leqslant -10x_1 + x_1^2 - 10x_2 + x_2^2 + 34 \longrightarrow (-10 + 2x_1)(1 - x_1) + \\ + (-10 + 2x_2)(4{,}9 - x_2) \leqslant 0$$

$$0 \leqslant -10x_1 + x_1^2 + 10x_2 + x_2^2 + 34 \longrightarrow$$
$$\longrightarrow -10 - 2x_1 + 10x_1 - 2x_1^2 - 49 + 9\ , \ 8x_2 + 10x_2 - 2x_2^2 \leqslant 0$$

La expresión se verifica para todo $x \varepsilon X$, es decir

$$1 \leqslant x_1 \leqslant 4{,}9 \quad \text{y} \quad 1 \leqslant x_2 \leqslant 4{,}9 \quad \text{en } h_1(x).$$

8.4.3. *h es cuasiconvexa y cuasicóncava en x*

Condición de cuasiconvexidad y cuasiconcavidad respectivamente.

$$h(x^2) \leqslant h(x^1) \longrightarrow \nabla h(x^1)\,(x^2 - x^1) \leqslant 0$$

$$h(x^2) \geqslant h(x^1) \longrightarrow \nabla h(x^1)\,(x^2 - x^1) \geqslant 0$$

con lo que

$$h(x^2) = h(x^1) \leftrightarrow \nabla h(x^1)\,(x^2 - x^1) = 0$$

para

$$\left.\begin{array}{lll} x^2 = [1 & 4{,}9] & 1 \leqslant x_1 \leqslant 4{,}9 \\ x^1 = [x_1 & x_2] & 1 \leqslant x_2 \leqslant 4{,}9 \end{array}\right\} \text{en } h_1(x)$$

8.2.1. Se verifica un óptimo global dado que

$$\bar{x} = [1 \quad 4{,}9] = \{x | x \varepsilon X^o, g_1(x) \leqslant 0, g_2(x) \leqslant 0, g_3(x) \leqslant 0, h_1(x) = 0\}$$
$$\theta(\bar{x}) = \min_{x \varepsilon X} \theta(x)$$

9. APLICACIONES

En esta sección se desarrollarán ejemplos simples de las aplicaciones más relevantes que se pueden hacer del algoritmo de Paviani y Himmelblau. Los ejemplos fueron elegidos con el objeto de indicar un camino para el planteo de otros similares. En trabajos posteriores expondremos soluciones de problemas más complejos incluyendo el poliedro de cotas. Sólo con la prueba de problemas similares a los de la realidad económica se podrá determinar fehacientemente la eficacia del algoritmo; el uso de equipos con más capacidad a la de la IBM 1130 arrojarán la certeza de que el M.T.F. es apto para resolver problemas de P.N.L.

9.1. *Programación Lineal*

Si el problema tiene solución, la eficacia de aplicar el algoritmo M T F es muy cercano al obtenido por el Simplex. En la presentación del problema no se incluyen variables de holgura, las que luego son calculadas cuando se haya obtenido el vector solución.

Ejemplificativamente,

$$\begin{aligned} \text{MAX } Z &= 3x_1 + 4x_2 \\ & 3x_1 + 2x_2 \leqslant 8 \\ & x_1 + 4x_2 \leqslant 10 \\ & x_1 , x_2 \geqslant 0 \end{aligned}$$

La solución según el algoritmo de Dantzig

$$x_1 = 1{,}2 \quad ; \quad x_2 = 2{,}2$$

Por el M.T.F.

$$x_1 = 1{,}197796 \quad ; \quad x_2 = 2{,}200520$$

9.2. *Programación Cuadrática*

En forma similar a la P.L. los resultados obtenidos por el M.T.F. son similares a los que surgen del algoritmo de P. Wolfe.

Tomemos el siguiente ejemplo: [17]

$$\begin{aligned} \text{MAX} &= 750\text{-}0{,}1(x_1-50)^2-0{,}2(x_2-50)^2 \\ & 5x_1 + x_2 \leqslant 200 \\ & x_1 + 2x_2 \leqslant 90 \\ & x_1 \leqslant 0 \\ & x_2 \geqslant 0 \end{aligned}$$

La solución es por Wolfe:

$$x_1 = 30$$
$$x_2 = 30$$

y por el M.T.F.

$$x_1 = 30{,}00951$$

o $\quad x_2 = 29{,}99562$

Incluímos seguidamente las salidas de los puntos 9.1 y 9.2.

[17] Tomado de Ref. (6), pág. 274 y siguientes.

PROGRAMA DE
PROGRAMACION NO LINEAL
RESTRINGIDA (MINIMIZACION) *Problema 9.1.*

ALGORITMOS UTILIZADOS

POLIEDRO FLEXIBLE (NELDER y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PROG. MATEMATICA

NUMERO DE VARIABLES INDEPENDIENTES 2
NUMERO DE IGUALDADES 0
NUMERO DE DESIGUALDADES 4
MAGNITUD DEL POLIEDRO INICIAL 0.1000 E 00
LA CONVERGENCIA DESEADA ES 0.1000 E-03
ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.100000E 01 0.100000E 01

EL CRITERIO DE TOLERANCIA INICIAL ES 0.200000E 00
LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS ES
0.000000E 00

VALOR DE LA FUNCION OBJETIVO= -0.7000000E 01

EL VECTOR ES
0.1000000E 01 0.1000000E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.3000000E 01 0.5000000E 01 0.1000000E 01 0.1000000E 01

****

ITERACION NUMERO = 12
CRITERIO DE TOLERANCIA = 0.334073E-01

VALOR DE LA FUNCION OBJETIVO = 0.1232708E 02

EL VECTOR ES
0.1235623E 01 0.2155054E 01

LOS VALORES DE LAS DESIGUALDADES SON
-0.1697731E-01 0.1441631E 00 0.2155054E 01 0.1235623E 01

****

ITERACION NUMERO = 24
CRITERIO DE TOLERANCIA = 0.221433E-02

VALOR DE LA FUNCION OBJETIVO = -0.1237693E 01

EL VECTOR ES
0.1207689E 01 0.2188467E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.0000000E 00 0.3844452E-01 0.2188467E 01 0.1207689E 01

****

ITERACION NUMERO = 36
CRITERIO DE TOLERANCIA = 0.554843E-03

VALOR DE LA FUNCION OBJETIVO = 0.1239514E 02

EL VECTOR ES
0.1200115E 01 0.2198700E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.2254486E-02 0.5084992E-02 0.2198700E 01 0.1200115E 01

NUMERO TOTAL DE ITERACIONES = 45
LIMITE DE CONVERGENCIA = 0.949866E-04

VALOR DE LA FUNCION OBJETIVO = -0.1239547E 02

EL VECTOR ES
0.1197796E 01 0.2200520E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.5571366E-02 0.1220703E-03 0.2200520E 01 0.1197796E 01

ESTOS SON RESULTADOS FINALES

****

PROGRAMA DE
PROGRAMACION NO LINEAL
RESTRINGIDA (MINIMIZACION)

Problema 9.2

ALGORITMOS UTILIZADOS

POLIEDRO FLEXIBLE (NELDER y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PROG. MATEMATICA

NUMERO DE VARIABLES INDEPENDIENTES	2
NUMERO DE IGUALDADES	9
NUMERO DE DESIGUALDADES	4
MAGNITUD DEL POLIEDRO INICIAL	0.10000E 00
LA CONVERGENCIA DESEADA ES	0.10000E 03

ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.100000E 01 0.100000 01

EL CRITERIO DE TOLERANCIA INICIAL ES 0.200000E 00
LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS ES
0.000000E 00

****

ITERACION NUMERO = 1
CRITERIO DE TOLERANCIA = 0.200000E 00

VALOR DE LA FUNCION OBJETIVO = -0.2970005E 02

EL VECTOR ES
0.10000000E 01 0.1000000E 01

LOS VALORES DE LAS DESIGUALDADES SON
0.1940000E 03 0.8700001E 02 0.1000000E 01 0.1000000E 01

****

ITERACION NUMERO = 12
CRITERIO DE TOLERANCIA = 0.3726776-01

VALOR DE LA FUNCION OBJETIVO = -0.6286173E 03

EL VECTOR ES
0.3315864E 02 0.2843386E 02

LOS VALORES DE LAS DESIGUALDADES SON
0.5772950E 01 -0.2636719E-01 0.3315864E 02 0.2843386E 02

****

ITERACION NUMERO = 24
CRITERIO DE TOLERANCIA = 0.372677E-01

VALOR DE LA FUNCION OBJETIVO = -0.6301300E 03

EL VECTOR ES
0.3007512E 02 0.2967877E 02

LOS VALORES DE LAS DESIGUALDADES SON
0.1964566E 02 -0.3265381E-01 0.3007512E 02 0.2997877E 02

****

ITERACION NUMERO = 36
CRITERIO DE TOLERANCIA = 0.120040E-01

VALOR DE LA FUNCION OBJETIVO = -0.6301022E 03

EL VECTOR ES
0.3003866E 02 0.2999348E 02

LOS VALORES DE LAS DESIGUALDADES SON
0.1981323E 02 -0.2561951E-01 0.3003866E 02 0.2999348E 02

****

ITERACION NUMERO = 48
CRITERIO DE TOLERANCIA = 0.763128E-03

VALOR DE LA FUNCION OBJETIVO = -0.6300030E 03

EL VECTOR ES
0.3000951E 02 0.2999562E 02

LOS VALORES DE LAS DESIGUALDADES SON
0.1995685E 02 -0.7476807E-03 0.3999951E 02 0.2999562E 02

ESTOS SON RESULTADOS FINALES

9.3. *Relación Primal-Dual en P.N.L.*

No vamos a abundar en la naturaleza de la especificación del dual en programación no lineal que bien se puede revisar en Mangasarian[18], ni en la interpretación económica de las variables primales en el dual que han delineado Balinski y Baumol[19].

Nuestro propósito aquí es sistematizar en qué casos, de acuerdo a la teoría y a la investigación empírica, los duales en P.N.L. han podido ser especificados. Seguimos en la siguiente tabla a P. Wolfe[20].

[18] Ver Ref. (3).
[19] Ver Ref. (8).
[20] Ver Ref. (4).

RESULTADOS DEL PRIMAL Y DEL DUAL

DUAL	PRIMAL			
	Tiene Solución	Acotado Sin Solución	No Acotado	Incompatible
Tiene solución	SI	?	NO	SI
Acotado sin solución	NO	?	NO	?
No acotado	NO	NO	NO	SI
Incompatible	NO	SI	SI	SI

La tabla anterior indica las relaciones entre el primal y el dual en P.N.L., cuatro alternativas se presentan:

a) que el problema tenga solución;

b) que las restricciones sean compatibles y la función a ser optimizada tenga cota en el sentido o dirección del óptimo, pero no exista solución;

c) que las restricciones sean compatibles pero la función sea no acotada en la dirección del óptimo; y

d) que las restricciones sean inconsistentes.

De los cruces de estas alternativas se genera el cuerpo de la tabla. Creemos importante hacer esta consideración para casos en que se desee determinar el dual por la importancia que los precios sombras tienen en Economía.

Por último, señalemos que el M.T.F. permite calcular el óptimo del dual de problemas condicionados por igualdades y que tradicionalmente se resolvían mediante la técnica de los multiplicadores de Lagrange.

9.4. *Optimización no restringida*

El M.T.F. no ha surgido del propósito de abordar este tipo de problemas, no obstante, es perfectamente aplicable a casos no restringidos.

Si la función presenta óptimo único, el algoritmo conduce a la solución global. De otra manera los óptimos relativos serán hallados según sea la elección del vector inicial. Aquí se torna de mucha utilidad el poliedro de cotas a que ya hicimos referencia en la sección 5.

Consideremos el siguiente ejemplo.

$$\text{MIN } y = 4(1-x_1)^2 + 2(2+4x_2)^2$$

La solución es: $x_1 = 1$

$$x_2 = -1/2$$

La solución según el M.T.F. es

$$x_1 = 1{,}000038$$
$$x_2 = -0{,}50002$$

Seguidamente incluímos la salida de computación.

PROGRAMA DE
PROGRAMACION NO LINEAL
RESTRINGIDA (MINIMIZACION) *Problema 9.4*

ALGORITMOS UTILIZADOS

POLIEDRO FLEXIBLE (NELDER Y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PROG. MATEMATICA

NUMERO DE VARIABLES INDEPENDIENTES 2
NUMERO DE IGUALDADES 0
NUMERO DE DESIGUALDADES 0
MAGNITUD DEL POLIEDRO INICIAL 0.10000E 00
LA CONVERGENCIA DESEADA ES 0.10000E—03
ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.100000E 01 0.100000E 01

EL CRITERIO DE TOLERANCIA INICIAL ES
0.200000E 00

LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS ES
0.200000E 00

****

ITERACION NUMERO = 1
CRITERIO DE TOLERANCIA = 0.200000E 00

VALOR DE LA FUNCION OBJETIVO = 0.7200001E 02

EL VECTOR ES
0.1000000E 01   0.1000000E 01

****

ITERACION NUMERO = 12
CRITERIO DE TOLERANCIA = 0.286107E-01

VALOR DE LA FUNCION OBJETIVO = 0.3381028E-01

EL VECTOR ES
0.1046332E 01   —0.5280756E 00

****

ITERACION NUMERO = 24
CRITERIO DE TOLERANCIA = 0.169494E-02

VALOR DE LA FUNCION OBJETIVO = 0.1678907E-04

EL VECTOR ES
0.1000290E 01   —0.5007171E 00

NUMERO TOTAL DE ITERACIONES = 34
LIMITE DE CONVERGENCIA = -0.4547176-04

VALOR DE LA FUNCION OBJETIVO = 0.1850321E-07

EL VECTOR ES

0.1000038E 01 —0.5000200E 00

ESTOS SON RESULTADOS FINALES

9.5. *Solución de sistemas de ecuaciones e inecuaciones de cualquier grado.*

Una aplicación de la P.N.L. es la solución de sistemas de ecuaciones e inecuaciones. Ya algo se vio cuando minimizamos T(X) en la sección 4.3, en que se buscaba un vector X tal que:

$$T(X) \leqslant \varnothing^{(k)}$$

Decíamos entonces, que cuando $\varnothing^{(k)} \rightarrow 0 \quad T(X) \rightarrow 0$ con lo que X sería un vector solución de todas las desigualdades e igualdades que actuaban como restricciones.

Si $T(X) = 0$, X es un *vector factible* de las restricciones y por lo tanto un vector solución. Recordemos:

$$T(X) = + \left[\sum_{i=1}^{m} h_i^2(X) + \sum_{i=m+1}^{p} U_i g_i^2(X) \right]^{1/2}$$

$$g_i(X) \geqslant 0 \rightarrow U_i = 0$$
$$g_i(X) < 0 \rightarrow U_i = 1$$

y que si T(X) es un funcional de igualdades y desigualdades podremos minimizarlo hasta determinar un vector X tal que $T(X) = 0$.

Según el algoritmo de la tolerancia flexible el vector solución será aquél que haga $T(X) = \varnothing^{(k)}$, donde $\varnothing^{(k)}$ es un valor en la etapa k de una función no creciente; este valor se determina en la iteración final según el criterio de convergencia ε, arbitrariamente elegido.

Para simplificar el funcional T(X) y no trabajar con el operador Heaviside U_i, se puede plantear la resolución de un sistema de ecuaciones de la siguiente forma:[21]

$$h_i(X) = 0 \qquad i = 1, 2, \ldots, m$$

[21] Ver Ref. (5), pág. 6.

$$T(X) = \sum_{i=1}^{m} h_i^2(X)$$

Cuando $T(X^{**}) \leqslant \varnothing^{(k^{**})}$, X^{**} es una solución al sistema de ecuaciones anterior con una aproximación dada por la diferencia entre $\varnothing^{(k^{**})}$ y cero. Se elevan al cuadrado las h_i con el objeto de evitar la cancelación de términos.

Cuando el sistema sea

$$g_i(X) \leqslant 0 \quad ; \quad i = m + 1, \ldots, p$$

se puede reconvertir las inecuaciones en ecuaciones adicionando una variable s de holgura; así:

$$g_i(X) + s_i = 0 \quad ; \quad \text{de donde}$$

$$T(X) = \sum_{i=m+1}^{p} (g_i(X) + s_i)^2$$

La explicación más interesante derivada de lo descripto es que un problema de programación no lineal puede ser abordado de dos maneras:

— a través del planteo en la forma descripta en la sección 7.
— mediante el sistema de ecuaciones e inecuaciones que surgen de las condiciones necesarias para un óptimo local planteados por Kuhn-Tucker, Fritz John, etc.[23]

Para ilustrar esta sección se han desarrollado dos ejemplos simples de sistemas de ecuaciones.

a) *Problema 9.5.1*

$$2x_1 + 4x_2 = 4$$
$$x_1 - 9x_2 = -9$$
$$X^* = [1 \; , \; 0]$$

Con un criterio de convergencia del 1% los resultados obtenidos son:

$$X^{**} = [1{,}092677 \qquad 0{,}01000091]$$

[22] Ver Ref. (3), págs. 93, 94 y 112.

PROGRAMA DE
PROGRAMACION NO LINEAL
RESTRINGIDA (MINIMIZACION) *Problema 9.5.1*

ALGORITMOS UTILIZADOS

POLIEDRO FLEXIBLE (NELDER y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PRUEBA PROG. NO LINEAL

NUMERO DE VARIABLES INDEPENDIENTES 2
NUMERO DE IGUALDADES 0
NUMERO DE DESIGUALDADES 0
MAGNITUD DEL POLIEDRO INICIAL 0.10000E 01
LA CONVERGENCIA DESEADA ES 0.10000E -02

ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.100000E 01 0.100000E 01

EL CRITERIO DE TOLERANCIA INICIAL ES 0.200000E 01
LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS ES
0.900000E 00

****

ITERACION NUMERO = 1
CRITERIO DE TOLERANCIA = 0.200000E 01

VALOR DE LA FUNCION OBJETIVO = 0.5000000E 01
EL VECTOR ES
0.1000000E 01 0.1000000E 01

****

ITERACION NUMERO = 12
CRITERIO DE TOLERANCIA = 0.533084E-01

VALOR DE LA FUNCION OBJETIVO = 0.4325812E-01

EL VECTOR ES
0.5161914E-01 0.1018107E 01

****

ITERACION NUMERO = 24
CRITERIO DE TOLERANCIA = 0.112035E-02

VALOR DE LA FUNCION OBJETIVO = 0.1717965E-04

EL VECTOR ES
-0.1541335E-02 0.1000218E 01

NUMERO TOTAL DE ITERACIONES = 25
LIMITE DE CONVERGENCIA = 0.771403E-03

VALOR DE LA FUNCION OBJETIVO = 0.6580330E-03

EL VECTOR ES
0.1092677E-02 0.1000091E 01

ESTOS SON RESULTADOS FINALES

b) *Problema 9.5.2*

$$2x^3_1 + 4x_2 x_3 = 0$$
$$4x_3 - x^2_2 = 0$$
$$\varepsilon = 1\%$$

$$X^{**} = \begin{bmatrix} 0{,}1338334 \\ 0{,}2233740 \\ 0{,}008145585 \end{bmatrix} \simeq \begin{bmatrix} 0{,}13 \\ 0{,}22 \\ 0 \end{bmatrix}$$

Reemplazando en el sistema originario

$$2 \times 0{,}13^3 + 4 \times 0{,}22 \times 0 = 0{,}002197 \simeq 0$$
$$4 \times 0 - 0{,}22^2 = 0{,}0484 \simeq 0$$

$$T(X^{***}) = 0{,}0004455033...$$

Haremos ahora una última observación sobre el método de resolución de ecuaciones e inecuaciones. La solución que provee el M. T. F. cuando el sistema es de grado superior a uno, no nos asegura la existencia o no de otras soluciones. Por otro lado, el desarrollo en la materia no provee suficientes elementos como para determinar la unicidad de la solución en estos casos. En sistemas lineales los desarrollos han determinado alternativas claras respecto a su solución,[23] aunque no se determina el método para generarla.

Finalmente indiquemos que el M.T.F. también puede ser empleado en la solución de ecuaciones o inecuaciones independientemente.

PROGRAMA DE
PROGRAMACION NO LINEAL
RESTRINGIDA (MINIMIZACION) *Problema 9.5.2*

ALGORITMOS UTILIZADOS

POLIEDRO FLEXIBLE (NELDER y MEAD)
SECCIONES MEDIAS (GOLDEN SECTIONS)
TOLERANCIA FLEXIBLE (PAVIANI-HIMMELBLAU)
INTERPOLACION DE LAGRANGE MODIFICADA

PRUEBA PROG. NO LINEAL

NUMERO DE VARIABLES INDEPENDIENTES	3
NUMERO DE IGUALDADES	0
NUMERO DE DESIGUALDADES	0
MAGNITUD DEL POLIEDRO INICIAL	0.10000E 01
LA CONVERGENCIA DESEADA ES	0.10000E -01

ALFA = 1.00
BETA = 0.50
GAMMA = 2.00

EL VECTOR INICIAL SELECCIONADO POR EL USUARIO ES
0.100000E 01 0.100000E 01 0.100000E 01

[23] Ver Ref. (7).

EL CRITERIO DE TOLERANCIA INICIAL ES 0.200000E 01
LA SUMA DE LAS RESTRICCIONES NO SATISFECHAS ES 0.000000E 00

****

ITERACION NUMERO = 1
CRITERIO DE TOLERANCIA = 0.200000E 01

VALOR DE LA FUNCION OBJETIVO = 0.4500000E 02

EL VECTOR ES
0.1000000E 01 0.1000000E 01 0.1000000 01

****

ITERACION NUMERO = 16
CRITERIO DE TOLERANCIA = 0.125049E 00

VALOR DE LA FUNCION OBJETIVO = 0.4542114E-01

EL VECTOR ES
0.2401374E 00 -0.4438830E-01 0.5357892E-01

NUMERO TOTAL DE ITERACIONES = 27
LIMITE DE CONVERGENCIA = 0.975640E-02

VALOR DE LA FUNCION OBJETIVO = 0.4455033E-03

EL VECTOR ES
0.1338334E 00 0.2233740E 00 0.8145585E-02

ESTOS SON RESULTADOS FINALES

Segunda Parte

1. *Programación General*

En esta parte del trabajo se va a plantear el caso más general de la programación restringida en donde se definen dominios discretos o limitados para las variables del problema.

Sea optimizar:

$$f(X)\varepsilon E^n$$

$$\left.\begin{array}{lll} h_i(X) = 0 & i = 1, \ldots\ldots, m \\ g_i(x) \quad 0 & i = m+1, \ldots, p \end{array}\right\} \textit{restricciones indirectas}$$

Se pueden plantear también *restricciones directas* que delimiten el dominio de las variables; éstas pueden ser:

a* — restricciones referidas al signo de la variable;

b* — restricciones que imponen a una variable ser entera o continua.

c* — restricciones que imponen que una variable sea múltiplo de un número;

d* — restricciones según las que una variable entera o múltiplo de un número debe asumir una cantidad finita de valores.

Para que una variable únicamente tenga soluciones enteras se incorpora la siguiente ecuación:

$$x_j - Ex_j = 0$$

donde E es un operador de integridad que traducido a FORTRAN puede equivaler a WHØLE $(x_j + 0,5) / x_j$, en la que WHØLE es una subrutina de truncamiento.

Si la restricción impone que x_j asuma en el vector solución solamente valores múltiplos de *w*, entonces se incorpora en el problema de programación la ecuación:

$$x_j - wEx_j = 0$$

donde *w* puede ser real o entera.

En los casos en que una variable debe limitarse a asumir determinados valores enteros o fraccionarios (R_j), se incluye una ecuación por variable de la siguiente característica:

$$\sum_{t=1}^{T} (x_j - R_j)^{T-(t-1)} = 0$$

donde:

T = número de valores que determinan el dominio de x_j
R_j = valores que puede asumir x_j

2. *Ejemplos*

2.1. $\min Z = (x_1 - 2.5)^2 + (x_2 - 3.5)^2$

$$(x_1 - 3)(x_2 - 3) - 1 \geqslant 0$$
$$x_1 \geqslant 0$$
$$x_2 \geqslant 0$$

solución en 19 iteraciones

$Z^* = 2.247111$
$x_1^* = 3.779551$
$x_2^* = 4.280936$

Criterio de convergencia = 0,001

Vector inicial = [0,0001 0,00001]

2.2. $\min Z = (x_1 - 2.5)^2 + (x_2 - 3.5)^2$

$$x_1^2 - x_1 = 0$$
$$(x_1 - 3)(x_2 - 3) - 1 \geqslant 0$$
$$x_1 \geqslant 0$$
$$x_2 \geqslant 0$$

solución en 18 iteraciones

$Z^* = 3.207228$
$x_1^* = 1.015112$
$x_2^* = 2.498833$

Criterio de convergencia = 0,01

Vector inicial = [0,0001 0,00001]

En este ejemplo x_1 tiene como valores opcionales $0 - 1$; de esta manera la inecuación $x_1 \geqslant 0$ resulta redundante, pero sirve de guía para la búsqueda del vector solución.

2.3. $\min Z = (x_1 - 2.5)^2 + (x_2 - 3.5)^2$

$$x_2 - Ex_2 = 0 \;;\; Ex_2 = \text{WH}\varnothing\text{LE}\,(x_2 + 0.5)$$
$$(x_1 - 1.5)\,(x_1 - 3.5) = 0$$
$$(x_1 - 3)\,(x_2 - 3) - \geqslant 0$$
$$x_2 \geqslant 0$$

solución en 22 iteraciones

$Z^* = 3.247429$
$x_1^* = 3.500168$
$x_2^* = 4.999032$

Criterio de convergencia = 0,001

Vector inicial = [0,1 0,1]

Este ejemplo exige que x_2 sea entera, según la primera ecuación mientras que x_1, puede asumir opcionalmente los valores 1.5 ó 3.5; la presentación de la restricción es la de una ecuación cuyas raíces son los valores antes señalados.

2.4. $\min Z = (x_1 - 2.5)^2 + (x_2 - 3.5)^2$

$$x_1 - Ex_1 = 0 \;;\; Ex_1 = \text{WH}\varnothing\text{LE}\,(x_1 + 0.5)$$
$$(x_1 - 3)\,(x_2 - 3) - \geqslant 0$$
$$x_1 \geqslant 0$$
$$x_2 \geqslant 0$$

solución en 16 iteraciones

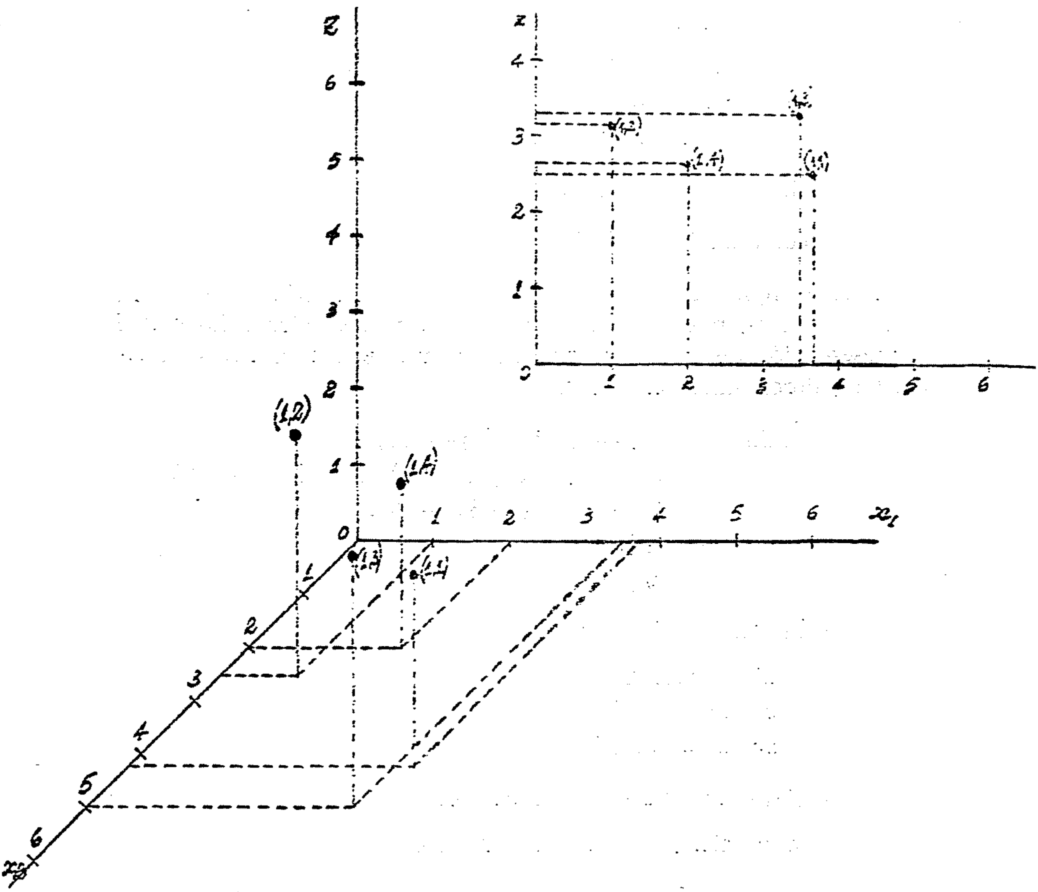
$Z^* = 2.474315$
$x_1^* = 1.991639$
$x_2^* = 2.011415$

Criterio de convergencia = 0,01

Vector inicial = [0,0001 0,00001]

La particularidad de este ejercicio radica en que por la primera ecuación se restringe a x_1, a ser entera.

De la comparación de los resultados podemos obtener una idea de la potencialidad del procedimiento para incluír restricciones que en forma de ecuaciones determinan el dominio de cada variable. La capacidad de la computadora y la elección de α, β, γ ordenados para este nuevo planteo indicarán el camino para la solución de problemas de programación general aún no abordados, y la medición de casos de P.L. aún discutidos especialmente en el campo de la programación entera mixta y 0-1.



Quizás la aplicación más inmediata de este sistema de programación general, sea el de la programación lineal entera, donde el método parece brindar mejores resultados que el de resolver el problema de P.L. y luego utilizar las ecuaciones de corte de Gomory.

En un simple ejemplo se puede visualizar la eficiencia del algoritmo de Himmelblau aplicado a un caso de programación toda entera.

$$\text{MAX } Z = 3x_1 + 4x_2$$ [24]

$$3x_1 + 2x_2 \leqslant 8$$
$$x_1 + 4x_2 \leqslant 10$$
$$x_1;\ x_2 \geqslant 0$$
$$x_1;\ x_2 \text{ enteros.}$$

La solución mediante el algoritmo de Gomory es:

$$x_1 = 1$$
$$x_2 = 2$$

mientras que utilizando el algoritmo de Himmelblau el vector final para un criterio de convergencia del 0,001 es:

$$x_1 = 1{,}000728$$
$$x_2 = 2{,}001173$$

Vector inicial = [1 1]

Número de iteraciones = 23.

10. LISTADO DE REFERENCIAS

(1) PAVIANI, D. y HIMMELBLAU, D. M., *Operations Research*, vol. 17, 1969.

(2) HIMMELBLAU, David M.,*Applied Nonlinear Programming*, Mc. Graw-Hill, Inc. 1972.

(3) MANGASARIAN, Olvi L., *Nonlinear Programming*, Mc Graw-Hill Book Company, 1969.

(4) WOLFE, Philip. *A Duality Theorem for Non-linear Programming*. The Rand Corporation Quarterly of Applied Mathematics, vol. 14, 1968.

(5) WILLARD, E. Zang Will, *Non linear Programming: A Unified Approach*, Prentice-Hall, Englewood Cliffs, N.J. 1967.

(6) NAYLOR, Thomas y J. VERMON, *La Economía de la Empresa*, Amorrortu Editores, Bs. As. 1973.

(7) ORBAN, Rolando F. *Sistemas Relacionados de Ecuaciones e Inecuaciones Lineales*. Instituto de Matemática y Estadística - Publicaciones 1973.

(8) BALINSKI, M. L. y BAUMOL, W. J., *The Dual in Non-linear Programming and its Economic Interpretations*, Review of Economic Studies.

(9) DONOLO, D. D., *Factibilidad en Programación Entera: Suscinta Recopilación y Casos Especiales*. Publicaciones Instituto de Econometría y Estadística - Facultad de Ciencias Económicas, UNC., 1975.

[24] Tomado de Ref. (9), pág. 6.