

01186

/

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTA DE INGENIERIA

**DESARROLLO DE UN NUEVO ALGORITMO PARA
LA OPTIMIZACIÓN GLOBAL DE FUNCIONES**

T E S I S

**QUE PARA OBTENER EL GRADO DE:
DOCTOR EN INGENIERIA**

MECANICA TEORICA Y APLICADA

PRESENTE

ANTONIO MONTALVO ROBLES

MEXICO, D.F., 2002

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

FACULTAD DE INGENIERIA
DIVISION DE ESTUDIOS SUPERIORES
SECCION DE MECANICA TEORICA Y APLICADA.

DESARROLLO DE UN NUEVO ALGORITMO
PARA LA
OPTIMIZACION GLOBAL DE FUNCIONES

Tesis que presenta

ANTONIO MONTALVO ROBLES

Para obtener el grado de

DOCTOR EN INGENIERIA

(MECANICA TEORICA Y APLICADA)

JURADO

Presidente: DR. JOSE A. NIETO R.

Vocal: DR. LUIS ESTEVA M.

Vocal: DR. PORFIRIO BALLESTEROS B.

Vocal: DR. JAIME CERVANTES DE G.

Secretario: DR. ALEJANDRO VELASCO L.

Suplente: DR. FELIPE OCHOA R.

Suplente: DR. LUIS FERRER A.

JEFE DE LA SECCION

DR. PORFIRIO BALLESTEROS B.

SECRETARIO ACADEMICO

DR. UBALDO BONILLA D.

Ciudad Universitaria, México, D.F.

2002

TESIS CON
FALLA DE ORIGEN

INDICE

RESUMEN	i
RECONOCIMIENTOS	iii
I INTRODUCCION	1
II DEFINICIONES Y POSTULADO DEL PROBLEMA	7
III REVISION DE OTROS ALGORITMOS	8
3.1 Modificación de Hardy al método de Branin	9
3.2 Algoritmo de Goldstein y Price	11
3.3 Métodos de arranque aleatorio (MRS y MMRS)	13
IV METODO DE TUNELIZACION	15
4.1 Fase de minimización	15
4.2 Fase de tunelización	16
4.3 Algunas propiedades del algoritmo de tunelización	23
4.4 Resumen del algoritmo	26
V EXPERIMENTOS NUMERICOS Y RESULTADOS	33
5.1 Condiciones de paro	33
5.2 Ejemplos	35
5.3 Criterio de evaluación	52
5.4 Resultados	53
VI CONCLUSIONES	66
6.1 Método de arranque múltiple aleatorio (MRS)	66
6.2 Método de arranque múltiple aleatorio modificado (MMRS)	66
6.3 Método de tunelización	66
VII REFERENCIAS	69

APENDICES

A. RESULTADOS ADICIONALES	71
B. METODOS DE MINIMIZACION LOCAL	86
C. ANALISIS DEL COMPORTAMIENTO DEL METODO DE GRADIENTE CONJUGADO	95
D. CONVERGENCIA GLOBAL DEL ALGORITMO DE TUNELIZACION	101

RESUMEN

Como primera parte de esta tesis se hace una revisión de varios de los métodos que han sido reportados en la literatura para resolver el problema de localizar el mínimo global de una función multimodal de varias variables. Si bien es cierto que algunos de los métodos reportados poseen algunas características que los hacen atractivos, también lo es que poseen algunas desventajas que limitan su aplicabilidad, siendo estas últimas las que predominan en algunas condiciones.

Lo expuesto anteriormente sirvió de motivación para una investigación cuyo propósito sería el desarrollar un nuevo método para localizar el mínimo global de una función. Sin embargo, este nuevo método debería tener algunas características especiales que lo hicieran superior a otros, que incorporara en lo posible las ventajas de estos últimos y que al mismo tiempo eliminara sus desventajas.

El resultado de la investigación fue el desarrollo del método de TUNELIZACION, el cual ha demostrado ser un algoritmo adecuado para resolver el problema mencionado en el primer párrafo. El método de tunelización que se presenta en las siguientes páginas, consiste de dos fases: Una fase de minimización, seguida de una fase de generación de nuevos nominales que favorecen el acercamiento del algoritmo al mínimo global (fase de tunelización). Estas fases, empleadas secuencialmente, tienen en conjunto las siguientes características:

- a) Se acercan monotónicamente al mínimo global
- b) Una vez que se ha encontrado un mínimo local, el método ignora, por construcción, todos los mínimos que existen por arriba del encontrado, sin importar cuántos son ni dónde están localizados.

Además de las características ya mencionadas, cuando el algoritmo de tunelización se compara con los ya reportados en la literatura, el primero resulta superior a los otros si la función que se minimiza satisface las siguientes condiciones:

- a) La región de interés es grande

- b) La dimensionalidad del problema es alta
- c) La función exhibe una gran densidad de mínimos locales, distribuidos uniformemente en la región de interés, o al menos estos mínimos se encuentran alrededor del global

Por otro lado, cuando las condiciones anteriores no se satisfacen, si bien es cierto que el algoritmo de tunelización sigue convergiendo al mínimo global, se ve superado por otro tipo de métodos los cuales localizan el mínimo global más rápidamente.

Complementando los puntos anteriores, se pudo observar que cuando se intenta resolver un problema de minimización global, no es muy recomendable emplear métodos de minimización local con propiedades derivadas en base a puntos anteriores, o con fuerte convergencia local, como son los métodos de gradiente conjugado o los derivados del de Newton. Esta aseveración está completamente justificada y detallada en la tesis, tanto teórica como experimentalmente.

Finalmente, se incluyen una serie de ejemplos los cuales ilustran las características mencionadas con anterioridad. Estos ejemplos contienen desde una hasta diez variables independientes y exhiben desde uno hasta 18 mínimos globales, con un total desde tres hasta miles de puntos estacionarios.

RECONOCIMIENTOS

El autor desea agradecer a las siguientes personas e instituciones su cooperación:

Petróleos Mexicanos, al otorgar las facilidades necesarias para poder terminar mis cursos y por permitir el uso de sus instalaciones de cómputo.

Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas de la UNAM, por las facilidades y apoyo brindado.

M. en C. José Luis Marroquín Z., autor del paquete de graficación con el cual se generaron las gráficas que se incluyen en esta tesis.

A los Srs. Miembros del Jurados, por su participación.

Dr. Alejandro Velasco Levy, por su ayuda y dirección durante todas las fases de esta investigación y por la amistad con la que me ha distinguido.

A mi esposa Angeles
por su constante estímulo
e inquebrantable fé

A mis hijas
Angélica y Andrea
por su ternura

I.- INTRODUCCION

El problema de localizar el mínimo global de una función ha recibido relativamente poca atención en la literatura a pesar de ser uno de los problemas más interesantes que se pueden encontrar en el campo de la programación no lineal (Ref. 4). Además del interés intrínseco que este problema tiene para un matemático, resulta por demás obvia la importancia que tiene en problemas del "mundo real", como los que se describen brevemente más adelante.

A nadie escapa que cuando se tiene un problema de minimización, siempre surgen las siguientes preguntas: "Si minimizo la función que describe mi problema, 1) ¿cómo sé que la función no tiene más de un mínimo? y 2) una vez que encuentro un mínimo, ¿cómo se que no hay ningún otro mínimo por debajo, o con igual valor de la función, que el que ya encontré?"

Las dos preguntas del párrafo anterior, las cuales normalmente no encuentran respuesta, son la principal motivación que se tuvo para desarrollar la presente tesis.

A continuación se describirán una serie de ejemplos tomados de diferentes disciplinas y cuyo único propósito es el de ilustrar la existencia de problemas que requieren del uso de un algoritmo de minimización global:

Reactor Químico Autotérmico (Ref. 16)

Considérese la operación de un reactor autotérmico como el que se muestra en la Fig. 1, en el cual se lleva a cabo una reacción de primer orden, irreversible, con una velocidad de reacción $r = kC_A$. Se desean averiguar cuáles son todos los posibles estados de operación estacionaria que se presentan en dicho reactor.

Si se definen las siguientes variables,

$$x = \frac{C_A^* - C_A}{C_A^*}$$

$$\theta = \frac{V}{F}$$

las ecuaciones que determinan las condiciones de operación del reactor, están dadas por

$$x = \frac{k\theta}{1 + k\theta} = \frac{\theta k_0 \exp(-E/RT)}{1 + \theta k_0 \exp(-E/RT)}$$

Y

$$x = \rho C_p (T - T_0) / C_A^* (-H)$$

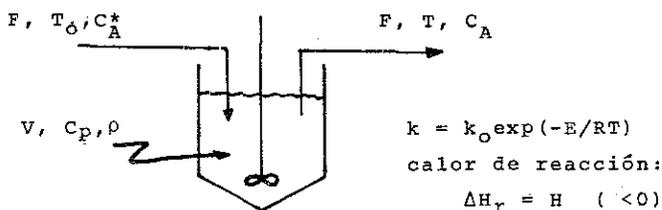


Figura 1.- Reactor Químico Autotérmico.

Este par de ecuaciones simultáneas, en x y T , exhiben una, dos o hasta tres soluciones, dependiendo del valor de las constantes que en ellas intervienen.

Existen al menos dos formas de encontrar los valores de x y de T que satisfagan las ecuaciones. La primera de ellas sería resolviéndolas una gran cantidad de veces y analizar los resultados que se hayan obtenido, confiando en que al final, se hubieran encontrado todas las soluciones del sistema. La segunda forma de hacerlo, sería encontrando los mínimos globales de la siguiente ecuación, la cual se obtiene a partir del sistema de ecuaciones simultáneas:

$$\min_T F(T) = \left\{ \frac{\rho C_p (T - T_0)}{C_A^* (-H)} = \frac{\theta k_0 \exp(-E/RT)}{1 + \theta k_0 \exp(-E/RT)} \right\}^2$$

ecuación que puede tener uno, dos o hasta tres mínimos globales, $F(T) = 0$, y hasta dos puntos de inflexión.

Ejemplos de esta naturaleza pueden encontrarse en biología

(relojes biológicos), en el diseño de reactores tubulares autotérmicos, así como en reactores de retromezclado en cascada.

Determinación de Estados de Equilibrio en Sistemas Mecánicos.

Considérese el sistema mecánico constituido por eslabones, el cual se muestra en la figura 2. Si la longitud original de cada resorte es d , las posibles condiciones de equilibrio del sistema están determinadas por los puntos extremos de la energía potencial asociada al sistema, i.e.,

$$V(\theta) = k(d - l_R)^2/2 + Wl\cos(\theta)/2$$

donde
$$l_R = \sqrt{\{l^2(1 - \cos\theta)^2 + (d - l\sin\theta)^2\}}$$

Se puede demostrar que si lo que se desea encontrar es la posición de equilibrio con el menor contenido de energía, esto es equivalente a encontrar el mínimo global de la energía potencial.

Para el caso de n eslabones de longitud l_j , unidos a una pared mediante resortes cuya constante es k_j , la energía potencial del sistema esta determinada por

$$V = \sum_i^n \{W_i \left[\sum_j^{i-1} l_j \cos\theta_j + l_i \cos\theta_i/2 \right] \} + \left\{ \sum_i^n k_i (d - l_{Ri})^2 \right\} / 2$$

donde
$$l_{Ri} = \sqrt{\left\{ \left[\sum_j^i l_j (1 - \cos\theta_j) \right]^2 + \left(d - \sum_j^i l_j \sin\theta_j \right)^2 \right\}}$$

y el estado de equilibrio conteniendo menor energía, corresponde al mínimo global de $V(\theta_1, \theta_2, \theta_3, \dots, \theta_{n-1}, \theta_n)$.

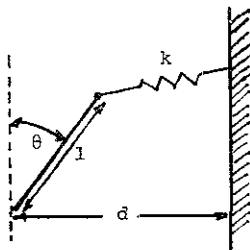


Figura 2.- Sistema Mecánico.

Diseño de Filtros Digitales (ref. 17)

Un problema común en el diseño de filtros digitales es encontrar los coeficientes del filtro digital recursivo tal que se ajuste a las especificaciones de magnitud deseada. Este problema se puede plantear de la siguiente forma:

Supóngase que las características de magnitud de un proceso se describen en un conjunto discreto de frecuencias $\omega_1, \omega_2, \omega_3, \dots, \omega_M$, como Y_i^d , $i = 1, 2, \dots, M$. Si

$$z_i = \exp(j\omega_i \pi) \quad i = 1, 2, \dots, M$$

$$Y(z) = \frac{\prod_{k=1}^K (1 + a_k z^{-1} + b_k z^{-2})}{\prod_{k=1}^K (1 + c_k z^{-1} + d_k z^{-2})}$$

se pretenden encontrar los valores de $(a_1, b_1, c_1, d_1, \dots, a_K, b_K, c_K, d_K)$, tales que

$$Q = \sum_{i=1}^M \{ |Y(z_i)| - Y_i^d \}^2$$

adquiera el mas bajo valor posible (mínimo global)

Cada uno de los factores que intervienen en la ecuación para $Y(z)$, corresponde a la realización en cascada que se muestra en la figura 3.

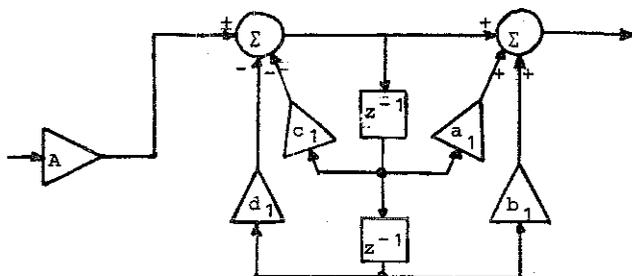


Figura 3.- Realización de un Filtro Recursivo

El problema que se ha planteado corresponde a un problema de minimización con restricciones (restricciones de mínima fase y de estabilidad). Sin embargo, las restricciones se pueden ignorar ya que se sabe que el mínimo global de Q siempre queda dentro de la región admisible.

Este problema pertenece a una clase más amplia de problemas

que en general se pueden describir como problemas de ajuste no lineal, área en la cual el algoritmo que se presenta en los siguientes capítulos tendrá una aplicabilidad hasta ahora restringida a otra clase de algoritmos.

Optimización de Ciclos de Energía Térmica (Refs. 12,13,14)

En una serie de trabajos recientes, Pope et al., describen varios problemas de minimización de funciones, cuyo formulación corresponde a problemas prácticos. Debido a lo extenso de la formulación de estos problemas, ésta no se ha incluido. Unicamente se debe señalar el hecho de que en todos los problemas, la función objetivo corresponde a una función multimodal.

La forma más común como se ha resuelto el problema de localizar el mínimo global de cada una de las aplicaciones mencionadas, es mediante el análisis de los resultados producidos mediante una serie de soluciones obtenidas con una computadora, en las cuales se localizan mínimos locales de la función objetivo. Si después de haber realizado dichos cálculos se tiene la certeza de que se han localizado todos los mínimos locales de la función objetivo, obviamente incluido en éstos deberá estar el mínimo global. Resulta también obvio que no hay forma de garantizar que se han localizado todos los mínimos locales, además de que cada vez que se realiza una nueva corrida, no se sabe de antemano si la función objetivo disminuirá o aumentará su valor en el nuevo mínimo local que se encuentre.

Todo lo anterior indica la necesidad de desarrollar un algoritmo que tenga como objetivo localizar el mínimo global de una función, o bien los mínimos globales si es el caso, siendo esto la investigación realizada y que se describe en las siguientes páginas.

Se podrían seguir mencionado ejemplos en otras áreas pero no es este el propósito que se persigue, sino simplemente el intentar dejar bien claro la necesidad de desarrollar un algoritmo que encuentre mínimos globales.

Finalmente, y para permitir al lector abordar el punto que más le interese en este trabajo, éste se ha dividido de la siguiente forma: En el segundo capítulo se dan una serie de definiciones de las

cuales frecuentemente se hará uso, además de plantear formalmente el problema que se pretende resolver. El capítulo tres, presenta una revisión de varios de los algoritmos reportados en la literatura para atacar el problema que se plantea en esta tesis. El siguiente capítulo contiene la parte medular de esta investigación ya que en él se describe el fundamento y los pasos que constituyen el método de tunelización; asimismo, se enuncian algunas propiedades del método y se especifican claramente los pasos que se deben seguir para implementarlo en una computadora. En el capítulo cinco, se describen los ejemplos que se emplearon para evaluar los diferentes algoritmos descritos en los capítulos tres y cuatro, indicándose, además, las condiciones experimentales que se usaron para correr los ejemplos. En este mismo capítulo se encuentran los resultados obtenidos, los cuales se reportan en forma de tabla para cada uno de los 18 ejemplos. En el último capítulo de la tesis se presentan las conclusiones que se obtuvieron al analizar los resultados del capítulo anterior, resultados que se obtuvieron al emplear el método de gradiente ordinario en las fases de minimización de los diferentes algoritmos de minimización global.

Por último, se incluyen tres apéndices cuyo contenido es el siguiente: En el apéndice A se reportan los resultados que se obtienen para los diferentes algoritmos cuando se usa en la fase de minimización los métodos de gradiente conjugado y Newton modificado. En el apéndice B se describen los métodos de gradiente ordinario, gradiente conjugado y Newton modificado tal y como se implementaron en los programas de cómputo. En el apéndice C se muestra el por qué no se deben emplear en la fase de minimización de un algoritmo de minimización global métodos como gradiente conjugado o métodos derivados del de Newton (tipo cuasi-Newton).

Durante la revisión de la presente, se añadió un cuarto apéndice en el cual se presenta la demostración teórica de la convergencia del algoritmo de tunelización hacia el mínimo global para el caso de funciones de una sola variable.

II.- DEFINICIONES Y POSTULADO DEL PROBLEMA.

Sea $f(x)$ una función escalar continua de n variables, con primeras y segundas derivadas continuas, definida en Ω , $\Omega = \{x \mid a \leq x \leq b\}$, y considérense las siguientes definiciones (Ref. 10):

Definición. Se dice que $f(x)$ exhibe un mínimo local en $x_k^* \in \Omega$ $k = 1, 2, \dots, K$, si

$$f(x_k^*) < f(x) \tag{1}$$

para $x \neq x_k^*$, $\|x - x_k^*\| < \delta$, $k = 1, 2, \dots, K$, $x \in \Omega$

Hay que hacer notar que si $f(x_k^*) = f(x_j^*)$, $k \neq j$, para alguna k , esta situación corresponde al caso en que la función $f(x)$ posee varios mínimos locales al mismo nivel.

Definición. Se dice que $f(x)$ exhibe un mínimo global estricto en $x^* \in \Omega$, si $f(x)$ exhibe un mínimo local en x^* y además

$$f(x^*) < f(x) \tag{2}$$

para $x \neq x^*$, $x \in \Omega$

Definición. Se dice que $f(x)$ exhibe un mínimo global en $x^* \in \Omega$, si $f(x)$ exhibe un mínimo local en x^* y además

$$f(x^*) \leq f(x) \tag{3}$$

para $x \neq x^*$, $x \in \Omega$

Esta situación se presenta cuando la función $f(x)$ exhibe varios mínimos globales al mismo nivel.

Con base en las anteriores definiciones, en la presente tesis se considera el problema de encontrar el mínimo global de la función

$$f = f(x) \tag{4}$$

donde x es un vector de n -componentes, $x \in \Omega$, $\Omega = \{x \mid a \leq x \leq b\}$, y, a y b son vectores prescritos de dimensión n . Se supondrá en lo sucesivo que tanto la función así como sus primeras y segundas derivadas existen y son continuas; por último, también se supondrá que el mínimo global existe, es decir, el problema que se plantea sí tiene solución.

III. - REVISION DE OTROS ALGORITMOS

En general, los métodos que han sido desarrollados y reportados en la literatura para resolver el problema de minimización global, pueden ser clasificados en alguno de los siguientes grupos.

a) Métodos de Ascenso (Hill Climbing).

En este tipo de métodos, todos los puntos extremos de una función (máximos, mínimos y silla) se encuentran secuencialmente conforme el método avanza. Si se establece un criterio de paro en el algoritmo, una vez que éste se ha alcanzado se comparan los valores que la función ha adquirido en todos los puntos estacionarios y se decide cual de ellos corresponde al mínimo global (Refs. 3,7)

b) Métodos de Modificación de la Función.

Esta clase de métodos se basan en la definición de una función auxiliar que se espera exhiba un mínimo local que corresponde a un valor de la función que se está minimizando, inferior al último mínimo encontrado. La localización de este mínimo de la función auxiliar puede ser empleado como punto de arranque de una nueva minimización de la función original. Usando este tipo de métodos en forma recursiva, es posible acercarse al mínimo global de la función (Ref. 6).

c) Métodos de Arranque Aleatorio.

En estos métodos, generalmente se dan una gran cantidad de puntos al azar, puntos que se usan como puntos de arranque para algún método de minimización cuyo propósito es encontrar mínimos locales. Al igual que los métodos de ascenso, la determinación de cuál de los mínimos locales encontrados corresponde al global se hace a posteriori (Ref. 1)

A continuación se describirán los algoritmos que se han empleado en este trabajo para resolver el problema planteado en el capítulo II de esta tesis. En lo que sigue, no se hace distinción (notación) entre escalares, vectores y matrices, como normalmente se acostumbra. La característica de cada una de las variables está en el contexto de lo que se esté describiendo en ese momento.

3.1.- Modificación de Hardy al Método de Branin (Refs. 3,7)

El problema de encontrar los puntos extremos de la función

$$f = f(x) \quad (5)$$

sujeta a las restricciones $a \leq x \leq b$, se puede reformular de la siguiente forma.

Sea

$$g(x) = \frac{\partial f}{\partial x} \quad (6)$$

donde $g(x)$, un vector de dimension n , es el gradiente de la función objetivo. De la teoría clásica de optimización, en un punto extremo x^* de $f(x)$, se satisface la condición de primer orden

$$g(x^*) = 0 \quad (7)$$

En otras palabras, se trata de encontrar un punto x^* , tal que la ecuación (7) se satisfaga dentro de una tolerancia ϵ y para tal fin se debe resolver el sistema de ecuaciones diferenciales ordinarias asociado a (7)

$$\frac{dx}{dt} = -(-1)^M \text{sign}(\det(H(x)))H^{-1}(x)g(x) \quad (8)$$

donde $M = 0, 1, 2, \dots$ es una variable que se incrementa cada vez que se alcanza un punto extremo de $f(x)$, y $H(x)$ es el Hessiano asociado a la misma función. Para resolver el sistema de ecuaciones (8), se considera la condición inicial

$$x(0) = x_0 \quad (9)$$

Para resolver el sistema de ecuaciones (8), se empleó un método del tipo predictor-corrector de cuarto orden con un método de Runge-Kutta del mismo orden para arrancarlo (Ref 8).

En el transcurso de obtener la solución del sistema de ecuaciones (8) se pueden presentar cualesquiera de los siguientes tres problemas; (a) reiniciar el algoritmo cada vez que se alcanza un punto estacionario de $f(x)$, (b) prevenir la falla del método cuando $H(x)$ se vuelve singular y (c) reiniciar el proceso cada vez que la solución de (8) alcance alguna de las fronteras. La forma de solucionar estos problemas fue la siguiente:



a) Proyección a través de un punto estacionario.

Sea $x(t_k)$ el valor del vector $x(t)$ en el k -ésimo paso de integración y que satisface

$$g^T(x(t_k))g(x(t_k)) \leq \epsilon_1 \quad (10)$$

y $x(t_{k-1})$ el valor del vector $x(t)$ en el $(k-1)$ -ésimo paso de integración en el cual no se satisface (10). Ya que $x(t_k)$ corresponde a la localización aproximada de uno de los puntos extremos de $f(x)$, si se continua el proceso de integración tal cual, resultará que el algoritmo se mantendrá indefinidamente en la vecindad de ese punto extremo y de aquí la necesidad de sacarlo de esta región para que se puede continuar la búsqueda de otro punto extremo. Con tal propósito, considérese el vector

$$y = \alpha x(t_k) + (1 - \alpha)x(t_{k-1}) \quad (11)$$

donde α es un escalar. El problema anterior se resuelve si se encuentra α (>1) tal que

$$g^T(y)g(y) > \epsilon_1 \quad (12)$$

lo que equivale, en cierta forma, a proyectar la trayectoria de acercamiento al punto extremo a través de éste. Una vez que se ha encontrado la α que haga que la ecuación (12) se satisfaga, se puede continuar con la generación de otra trayectoria que localice otro punto extremo, previo incremento de la constante M .

b) Proyección a través de un punto donde $H(x) = 0$.

Sean, nuevamente, $x(t_k)$ y $x(t_{k-1})$ los valores correspondientes al vector $x(t)$ en la k - y $(k-1)$ -ésima etapa de integración tales que

$$\det H(x(t_k)) \leq \epsilon_2 \quad (13)$$

$$y \quad \det H(x(t_{k-1})) > \epsilon_2 \quad (14)$$

respectivamente. Debido a los problemas de inestabilidad numérica que se presentan cuando se trata de invertir una matriz casi-singular, resulta que el proceso de integración no se puede continuar si la expresión (13) se satisface. Para continuar la búsqueda, se procede igual que en el caso anterior hasta encontrar un valor de α tal que

$$\det H(y) > \varepsilon_2 \quad (15)$$

Los puntos x que satisfacen

$$H(x) = 0 \quad (16)$$

$$g(x) \neq 0 \quad (17)$$

fueron llamados por Branin, en su artículo original, "singularidades extrañas" (Ref. 3)

c) La solución $x(t)$ alcanza la frontera de la región.

Al estar resolviendo las ecuaciones (8) puede suceder que se alcance alguno de los lados del (hiper-)cubo que sirve de frontera de la región. Esto puede ocurrir ya que si se analiza el algoritmo, no se tiene control alguno sobre la trayectoria que debe seguir $x(t)$. Cuando lo anterior ocurre, el último punto que se tenía antes de llegar a la frontera se refleja a través del centroide del cubo y se reinicia el proceso mediante la selección adecuada del valor de M . Esta selección se hace de manera tal que cuando se dá un paso de integración desde el nuevo punto de arranque, el siguiente punto esté dentro de la región de interés y no fuera de ella, en cuyo caso se incrementa en 1 (uno) el valor de la constante M .

3.2.- Algoritmo de Goldstein y Price (Ref. 6).

Se puede demostrar que este algoritmo converge al mínimo global de una función, siempre y cuando esta última sea un polinomio en una sola variable. Sin embargo, cuando alguna de estas dos características no se cumplen, lo anterior deja de ser cierto.

La derivación del algoritmo es la siguiente. Sea $f(x)$ una función de una sola variable que exhibe un mínimo local en x_1^* . Si $f(x)$ se expande en su serie de Taylor alrededor de x_1^* , se obtiene

$$f(x) = f(x_1^*) + \frac{f''(x_1^*)}{2!}(x-x_1^*)^2 + \frac{f'''(x_1^*)}{3!}(x-x_1^*)^3 + \dots \quad (18)$$

A partir de esta expansión, se define una función auxiliar $f_1(x)$ como

$$f_1(x) = \frac{f(x) - f(x_1^*)}{(x - x_1^*)^2} \quad (19)$$

Como se supuso que en x_1^* existe un mínimo local de $f(x)$, se tiene que $f''(x_1^*) > 0$, y entonces para alguna $\xi \neq x_1^*$, $f_1(\xi) \leq 0$, es decir $f(\xi) \leq f(x_1^*)$. Si se tiene éxito en localizar ξ , digamos en x_2^* , este punto puede ser empleado como punto de arranque de una nueva fase de minimización de $f(x)$, teniéndose la certeza de que si en x_2^* existe otro mínimo local de $f(x)$, se satisface que $f(x_2^*) \leq f(x_1^*)$. La forma de localizar x_2^* , cuando existe, consiste en encontrar un mínimo de $f_1(x)$. Si bien a primera vista el método resulta atractivo, se puede presentar la siguiente situación:

Supóngase que en ξ se localizó un mínimo de $f_1(x)$ tal que $f(\xi) > f(x_1^*)$. Resulta evidente que desde este nuevo punto no se puede reiniciar la búsqueda de un mínimo de $f(x)$ sin correr el peligro que al encontrarlo, suceda que $f(x_2^*) > f(x_1^*)$. Cuando esta situación se presenta, el proceso para definir $f_1(x)$, a partir de $f(x)$, se repite para generar ahora $f_2(x)$ a partir de $f_1(x)$, es decir, si en $\xi = x_2$ $f_1'(x_2) = 0$, $f_1''(x_2) > 0$ y $f_1(x_2) > 0$, $f_2(x)$ se define como sigue

$$f_1(x) = f_1(x_2) + \frac{f_1''(x_2)}{2!}(x - x_2)^2 + \frac{f_1'''(x_2)}{3!}(x - x_2)^3 + \dots \quad (20)$$

De esta expansión se obtiene $f_2(x)$ como

$$f_2(x) = \frac{f_1(x) - f_1(x_2)}{(x - x_2)^2} \quad (21)$$

Los argumentos para pasar de $f(x)$ a $f_1(x)$ y para el regreso a $f(x)$ se pueden extender fácilmente para pasar de $f_1(x)$ a $f_2(x)$ y regresarse a $f_1(x)$. Este procedimiento deberá ser repetido tantas veces como se requiera o bien se alcance algún criterio de convergencia (Goldstein y Price no establecen ningún criterio como condición de paro).

El análogo de la situación anterior en un problema n-dimensional es el siguiente. Sea

$$f(x) = f(x_1) + (x - x_1)^T H(x_1) (x - x_1) / 2 + \dots \quad (22)$$

donde en x_1 $f(x)$ exhibe un mínimo local, es decir, $H(x_1)$ es positiva definida. Siguiendo los lineamientos anteriores, $f_1(x)$ se define como

$$f_{.1}(x) = \frac{f(x) - f(x_1)}{(x - x_1)^T H(x_1) (x - x_1)} \quad (23)$$

y sobre esta función se busca un mínimo de $f_{.1}(x)$ que satisfaga que $f_{.1}(x) \leq 0$. Si este mínimo se localizo en x_2^o , este punto puede emplearse como punto de arranque para otra minimización de $f(x)$. Si en el punto x_2^o no se satisface que $f_{.1}(x_2^o) \leq 0$, el proceso descrito en los párrafos anteriores se repite tantas veces como sea necesario, o bien que se alcance alguna condición de paro del algoritmo.

3.3 Métodos de arranque aleatorio (Ref. 1).

Este tipo de métodos han sido los que más se han empleado para resolver el problema de minimización global. Se podría decir que la idea en la que se basan estos métodos es esencialmente la misma, y salvo ligeras variantes, esta consiste en suponer, a priori, que después de un cierto número de evaluaciones de la función, muy posiblemente el mínimo global de la función ha sido localizado. A continuación se describen dos de los métodos que se considera son razonablemente confiables para resolver el problema.

3.3.1.- Arranque Aleatorio Múltiple (MRS)

La forma en que este algoritmo fue implementado es la siguiente: Para un algoritmo de minimización dado, éste se corre empezando de diferentes puntos de arranque, seleccionados al azar, almacenándose el valor y la localización del mínimo local de $f(x)$ que haya sido encontrado. Después de un cierto tiempo de ejecución se detiene el algoritmo y posiblemente dentro de los varios mínimos locales encontrados se encuentre el mínimo global.

3.3.2.- Arranque Aleatorio Múltiple Modificado (MMRS)

Como se mencionó con anterioridad, este algoritmo tiene mas o menos la misma filosofía que el anterior. La única diferencia entre ellos es la siguiente. Sea x^* la localización de algún mínimo local de $f(x)$, cuyo valor es $f(x^*)$. Para continuar la búsqueda de otro mínimo, se genera un punto aleatorio, por ejemplo x^o , el cual se usa para iniciar la búsqueda de otro mínimo local, siempre y cuando suceda que en el nuevo punto se satisfaga que



$$f(x^0) - f(x^*) < \epsilon \quad (24)$$

Si para el punto x^0 la condición (24) no se satisface, se genera otro nuevo punto x^0 , repitiéndose este procedimiento hasta que la condición (24) es satisfecha, o bien, se alcance el tiempo límite de ejecución prescrito para el algoritmo.

Antes de pasar al siguiente capítulo, vale la pena hacer notar que de los cuatro algoritmos que se han descrito hasta este punto, los de Goldstein y Price y el de MMRS satisfacen la siguiente propiedad cada vez que encuentran un nuevo mínimo local de $f(x)$:

Sean x_1^* , x_2^* , ... las localizaciones de los mínimos encontrados en sucesivas etapas de minimización; si $f(x_1^*)$, $f(x_2^*)$, ... son los valores que adquiere la función en cada uno de estos puntos, fácilmente se ve que estos valores satisfacen la relación

$$f(x_1^*) \geq f(x_2^*) \geq \dots \quad (25)$$

IV.- METODO DE TUNELIZACION

El algoritmo de tunelización que a continuación se presenta, consiste de dos fases; una fase de minimización y una fase de tunelización. Estas dos fases son usadas alternadamente por el algoritmo y tienen como propósito el acercarse al mínimo global de $f(x)$.

4.1 Fase de minimización.

En esta fase, para un punto de arranque x^0 , teóricamente cualquier algoritmo de minimización que posea propiedad de descenso local sobre $f(x)$ puede ser empleado para localizar algún mínimo local de $f(x)$, por ejemplo en x^* . Lo anterior, no necesariamente es cierto en la práctica; de hecho, en el apéndice C, se presentan algunas recomendaciones sobre que clase de métodos de minimización pueden ser empleados (En la presente investigación todas las conclusiones están basadas en el método de gradiente ordinario)

NOTA: Debido a que se requiere que los puntos que genera el método de minimización permanezcan dentro de la región de interés $a \leq x \leq b$, durante todo esta fase, los desplazamientos que genera el método de minimización se controlan de la siguiente manera: Sea x el último punto producido por el algoritmo de minimización tal que $a \leq x \leq b$ y sea $\bar{x} = x + \Delta x$ el siguiente punto generado por el algoritmo y es tal que alguna de las fronteras de la región es violada. Para este caso, se encuentra un tamaño de paso $\Delta \bar{x} = \alpha \Delta x$ ($\alpha < 1$) tal que $\bar{x} = x + \Delta \bar{x}$ está en la frontera de la región y además se satisface que $f(\bar{x}) < f(x)$ (propiedad de descenso del algoritmo). Si la última desigualdad no se satisface se usa un proceso de bisección sobre α hasta que es satisfecha. Una vez que el algoritmo se encuentra en la frontera de la región, se calcula una nueva dirección de desplazamiento Δx , la cual varía según el método que se esté empleando, haciendo iguales a cero todos aquellos componentes del desplazamiento que tienden a violar alguna de las fronteras de la región; cuando esto último se hace, se tiene la certeza de que ninguna de las fronteras será violada, además de que no se pierde la propiedad de descenso si se usan como métodos de minimización gradiente ordinario o bien gradiente conjugado. Resulta fácil de concluir que si después de haber modificado el vector de desplazamiento este se convierte en un vector nulo, dentro de una cierta

tolerancia, se ha localizado un mínimo en la frontera de la región. Procediendo en la forma arriba indicada, la secuencia de puntos $\{x_k\}_{k=0}^{\infty}$, producida por el algoritmo de minimización, permanecerá siempre dentro de la región $a \leq x \leq b$.

Obviamente, si la fase de minimización produce un mínimo sobre la frontera de la región de interés, la fase de minimización deberá darse por terminada, tomando como punto de arranque en la fase de tunelización la localización del mínimo restringido que se localizó mediante el procedimiento anterior.

4.2.- Fase de tunelización.

El propósito principal de la fase de tunelización es la de encontrar un nuevo punto x^o que se usará como punto de arranque de la siguiente fase de minimización. El punto x^o debe ser tal que si se emplea como punto de arranque de la siguiente fase de minimización, el nuevo mínimo local de $f(x)$, localizado en x^{**} , sea tal que

$$f(x^{**}) \leq f(x^*) \quad (26)$$

donde x^* corresponde a la localización del mínimo encontrado en la anterior fase de minimización.

Para conseguir el objetivo mencionado, se debe tratar de localizar algún cero de la función de tunelización, definida como

$$T(x, I) = \frac{f(x) - f^*}{\{(x-x_m)^T(x-x_m)\}^{\lambda_0} \prod_{i=1}^n \{(x-x_i^*)^T(x-x_i^*)\}^{\eta_i}} \quad (27)$$

donde el significado de cada término se presenta a continuación. Además del significado de cada término, y por considerar de importancia la derivación de la función de tunelización, ésta se presenta en detalle. Lo anterior está motivado por la creencia de que el desglosar completamente las ideas que dieron como resultado dicha función, puede motivar al lector interesado en el tema para que intente mejorar el método en conjunto, lo que redundará en beneficio de todos aquellos investigadores que trabajan en esta área.

4.2.1.- Función de tunelización.

La función de tunelización $T(x, \Gamma)$ se puede derivar siguiendo el siguiente razonamiento:

a) Sea

$$T_1(x, f^*) = f(x) - f^* \quad (28)$$

donde $f^* = f(x^*)$ y $a \leq x^* \leq b$ es la localización de alguno de los mínimos locales de $f(x)$, localizado en la fase de minimización.

Es fácil observar que cualquier punto $x^o \neq x^*$ que satisfaga $T_1(x^o, f^*) = 0$ puede ser aceptado como punto de arranque de la siguiente etapa de minimización ya que x^o está localizado fuera de la región de atracción del mínimo localizado en x^* . Esto último garantiza que si x^o se usa como punto de arranque de la siguiente fase de minimización la cual converge a x^{**} , la condición (26) se verá satisfecha. Geométricamente, lo expuesto implica que el algoritmo, como un todo, tuneliza por debajo de todos los puntos estacionarios irrelevantes, acercándose de esta forma al mínimo global de $f(x)$.

b) Sea

$$T_2(x, f^*, x_1^*, \dots, x_l^*, \lambda_1, \dots, \lambda_l) = \frac{T_1(x, f^*)}{\prod_{i=1}^l \{(x-x_i^*)^T(x-x_i^*)\}^{\lambda_i}} \quad (29)$$

donde $\lambda_i > 0$, $i = 1, 2, \dots, l$ son escalares que se escogen de acuerdo a las reglas que se dan en la sección 4.2.2 y x_i^* , $i = 1, 2, \dots, l$ son las localizaciones de los mínimos encontrados durante l fases de minimización anteriores y que satisfacen

$$f(x_1^*) = f(x_2^*) = \dots = f(x_l^*) = f^* \quad (30)$$

es decir, la función exhibe al menos l mínimos locales al mismo nivel f^* .

Al igual que en el punto anterior, se observa que cualquier punto x^o que satisface $T_2(x^o; f^*, \dots) = 0$, puede ser usado como punto de arranque de la siguiente fase de minimización. Por otro lado, se observa que, por el hecho de escoger todas la $\lambda_i > 0$, $i = 1, 2, \dots, l$, no se introduce ningún cero adicional en la función, cumpliéndose además que

$$T_1(x^o, f^*) = T_2(x^o, f^*, \dots) = 0 \quad (31)$$

c) Sea

$$T_3(x, f^*, x_1^*, \dots, x_1^*, \lambda_1, \dots, \lambda_1, x_m, \lambda_0) = \frac{T_2(x, f^*, x_1^*, \dots, x_1^*, \lambda_1, \dots, \lambda_1)}{\{(x-x_m)^T(x-x_m)\}^{\lambda_0}} \quad (32)$$

donde x_m y λ_0 se escogen según se establece en las reglas que se mencionan en la sección 4.2.2.

Las observaciones hechas para la función definida por la ecuación (29), son nuevamente válidas para la función definida en este inciso, teniéndose además que

$$T_1(x^0, f^*) = T_2(x^0, f^*, \dots) = T_3(x^0, f^*, \dots) = 0 \quad (33)$$

d) Defínase

$$I = \{f^*, x_1^*, \dots, x_1^*, \lambda_1, \dots, \lambda_1, x_m, \lambda_0\} \quad (34)$$

Si se emplean las definiciones dadas en los puntos anteriores, se obtiene la expresión general de la función de tunelización, la cual está dada por

$$T(x, I) = \frac{f(x) - f^*}{\{(x-x_m)^T(x-x_m)\}^{\lambda_0} \prod_{i=1}^n \{(x-x_i^*)^T(x-x_i^*)\}^{\lambda_i}} \quad (35)$$

El propósito de cada uno de los términos contenidos en esta última expresión es el siguiente: El restarle a $f(x)$ el valor de f^* elimina como posible solución de esta fase a todos aquellos puntos para los cuales $f(x) > f^*$. El divisor introducido en $T_2(x, f^*, \dots)$ tiene el propósito de prevenir que esta fase localice como solución alguno de los puntos ya identificados como minimizadores de $f(x)$ cuando ésta adquiere el valor f^* . Finalmente, el divisor que se introduce en la definición de $T_3(x, f^*, \dots)$ tiene el propósito de impedir que la búsqueda de una solución se detenga en un punto estacionario de $T_2(x, f^*, \dots)$ punto que es completamente irrelevante como solución de $T(x, I)$, es decir, un punto en el cual se anula el gradiente de $T_2(x, f^*, \dots)$ sin que $T_2(x, f^*, \dots) = 0$.

4.2.2 Determinación de I

En la fase de tunelización, se puede emplear cualquier algoritmo de búsqueda de ceros, siempre y cuando posea propiedad de descenso local sobre $T(x, I)$; el objetivo es encontrar x^0 tal que $T(x^0, I) = 0$. En particular, en este trabajo se usó el algoritmo de restauración desarrollado por Miele et al., (Ref.11). A continuación, se dan las reglas que determinan el valor correcto de cada uno de los parámetros contenidos en I. Estos parámetros son dinámicos ya que conforme el algoritmo de búsqueda de ceros avanza, son determinados y ajustados automáticamente por la computadora.

Paso 1.- Determinación de f^*

Una vez que se ha localizado un mínimo de $f(x)$ en x^* , se pretende que el algoritmo de tunelización ignore todos los mínimos locales cuyo valor excede el de $f(x^*)$, ya que éstos son irrelevantes para localizar el mínimo global de la función. Este objetivo se alcanza si al empezar cada fase de tunelización se escoge el valor de f^* igual al del mínimo más bajo que se haya encontrado hasta este punto, correspondiendo este al localizado en x^* , i.e., $f(x^*)$. Con esta selección de f^* se consigue tunelizar por debajo de todos los mínimos locales irrelevantes, sin importar cuántos son ni dónde están localizados.

Paso 2.- Determinación de x_1^* y λ_1 , $i=1,2,\dots,l$

Para ilustrar claramente este paso del algoritmo, supóngase que λ_0 se escoge como cero y además $l = 1$, esto es, por primera vez se ha encontrado un mínimo de $f(x)$ el cual está localizado en x_1^* . De acuerdo al paso 1, se escoge $f^* = f(x_1^*)$. Como se mencionó anteriormente, se trata de encontrar un punto $x^0 \neq x_1^*$ tal que $T(x, I) = 0$ empezando en x_1^* . Para lograr esto último, se emplea la ecuación (35) que en este caso se reduce a

$$T(x, I) = T(x, f^*, x_1^*, \lambda_1) = \frac{f(x) - f^*}{\{(x-x_1^*)^T(x-x_1^*)\}^{\lambda_1}} \quad (36)$$

donde $\lambda_1 > 0$ se encuentra iterativamente empezando con $\lambda_1 = 1$, y mediante incrementos finitos $\Delta\lambda$, hasta que se logra la siguiente pro-

piedad de descenso

$$T(\bar{x}, f^*, x_1^*, \lambda_1) < T(x_1^* + \varepsilon, f^*, x_1^*, \lambda_1) \quad (37)$$

donde ε es un vector escogido al azar que satisface $\|\varepsilon\| < 1$, $\bar{x} = (x_1^* + \varepsilon) + \Delta x$, y $\Delta x = \alpha A(\lambda_1)$ es el desplazamiento actual obtenido con ε suficientemente pequeña, siendo $A(\lambda_1)$ la dirección de desplazamiento producida por el algoritmo de búsqueda de ceros.

Si para una cierta λ_1 la condición (37) no se satisface, λ_1 se incrementa en $\Delta\lambda$ hasta que dicha condición es satisfecha.

Para el caso de $l \geq 1$, es decir, la función exhibe al menos l mínimos locales al mismo nivel f^* , y l aplicaciones de la fase de minimización han identificado a $x_1^*, x_2^*, \dots, x_l^*$ como localizaciones de mínimos de $f(x)$ al nivel f^* , se procede de la siguiente manera: Conforme cada $x_i^*, i = 1, 2, \dots, l$ se encuentra, el proceso descrito para el caso de $l = 1$ se repite para determinar el correspondiente valor de $\lambda_i, i = 1, 2, \dots, l$. Por ejemplo, para $l = 2$, es decir, $f(x_1^*) = f(x_2^*) = f^*$ y habiéndose determinado como se explicó el valor de λ_1 , el valor de λ_2 se determina análogamente hasta que la condición

$$T(\bar{x}, f^*, x_1^*, x_2^*, \lambda_1, \lambda_2) < T(x_2^* + \varepsilon, f^*, x_1^*, x_2^*, \lambda_1, \lambda_2) \quad (38)$$

se satisface. Se observa que en este caso, $T(x, l)$ toma la forma

$$T(x, l) = T(x, f^*, x_1^*, x_2^*, \lambda_1, \lambda_2) = \frac{f(x) - f^*}{\prod_{i=1}^l \{(x - x_i^*)^T (x - x_i^*)\}^{\lambda_i}} \quad (39)$$

donde nuevamente a λ_0 se le ha dado el valor de cero.

Para problemas en los cuales l es grande, por ejemplo cuando la función exhibe una gran cantidad de mínimos locales al mismo nivel f^* , el denominador de la ecuación (35) puede hacerse muy grande si $\|x - x_i^*\| > 1$, para alguna i , lo que provoca que la función de tunelización $T(x, l)$ se vuelva plana y muy cercana a cero. Debido a que esta geometría particular de $T(x, l)$ reduce la velocidad de avance del algoritmo de búsqueda de ceros, la función de tunelización, ecuación (35), debe ser modificada para acelerar nuevamente la velocidad del



algoritmo.

La implementación final de la fase de tunelización se basa en el uso de la siguiente modificación de la ecuación (35), de acuerdo a

$$T(x, \Gamma) = \frac{f(x) - f^*}{\left\{ (x-x_m)^T (x-x_m) \right\}^{\lambda_0} \prod_{i=1}^l \left\{ (x-x_i^*)^T (x-x_i^*) \right\}^{\eta_i}} \quad (40)$$

donde los exponentes $\eta_i, i = 1, 2, \dots, l$ están definidos como

$$\eta_i = \begin{cases} \lambda_i & ; \text{si } r_i < d - \varepsilon_1 \\ \lambda_i \{1 + (d - r_i) / \varepsilon_1\} / 2 & ; \text{si } d - \varepsilon_1 < r_i < d + \varepsilon_1 \\ 0 & ; \text{si } r_i > d + \varepsilon_1 \end{cases} \quad (41)$$

donde $r_i = \|x - x_i^*\|$ y ε_1 ($\ll 1$) es un número pequeño predeterminado. Se observa que la ecuación (41) es una función-rampa y que $\eta_i(x_i)$ modifica el denominador de la ecuación (40) cuando se cruza alguno de los círculos de radio $d, d < 1$, los cuales rodean a cada uno de los puntos $x_i^*, i = 1, 2, \dots, l$

Paso 3.- Determinación de x_m y λ_0

Siempre que se empieza una fase de tunelización, el vector I se define como

$$I = \{f^*, x_1^*, \dots, x_l^*, \lambda_1, \dots, \lambda_l, x_1^*, 0\} \quad (42)$$

es decir, x_m , se localiza en el mismo punto que la posición del último mínimo encontrado y se hace además, $\lambda_0 = 0$.

Conforme el algoritmo de búsqueda de ceros avanza, x_m y λ_0 se actualizan como sigue: Sea Δx el desplazamiento producido por el algoritmo de búsqueda de ceros, el cual mueve x a \bar{x} , siendo x la localización del punto al principio de una iteración y $\bar{x} = x + \Delta x$ es tal que la propiedad de descenso

$$T(\bar{x}, I) < T(x, I) \quad (43)$$

se satisface.



Con estos dos últimos puntos, x y \bar{x} , x_m se define de acuerdo a

$$\bar{x}_m = \begin{cases} x & \text{si } ||x - \bar{x}|| \leq 1 \\ \beta x + (1 - \beta)\bar{x} & \text{si } ||x - \bar{x}|| > 1 \end{cases} \quad (44)$$

donde $0 < \beta < 1$ se escoge de manera tal que $||\bar{x} - \bar{x}_m|| < 1$.

Considérese ahora la determinación de λ_0 . Sean x y I el valor del vector posición y los parámetros al principio de una iteración del algoritmo. Sean $A(\lambda_0)$ la dirección del desplazamiento producido por el algoritmo de búsqueda de ceros y $\Delta x(\lambda_0) = \alpha A(\lambda_0)$ el desplazamiento actual obtenido con α suficientemente pequeña. Con este desplazamiento se calculan $\bar{x} = x + \Delta x(\lambda_0)$ y $T(x, I)$, de manera tal que la condición (43) se satisfaga. Si para α suficientemente pequeña se pierde la propiedad de descenso del algoritmo, se modifica λ_0 incrementandola en $\Delta\lambda_0$ y se repite el procedimiento hasta que la propiedad de descenso local es satisfecha nuevamente. Por el contrario, si para el valor de λ_0 al principio de la iteración se satisface la propiedad de descenso (43), se calcula $A(\lambda_0)$ haciendo, tentativamente, $\lambda_0 = 0$; con esto, el valor de λ_0 se actualiza de acuerdo a

$$\lambda_0 = \begin{cases} 0 & \text{si } A^T(\lambda_0)A(0) > 0 \\ \lambda_0 & \text{si } A^T(\lambda_0)A(0) \leq 0 \end{cases} \quad (45)$$

Vale la pena hacer notar que la determinación de λ_0 y cada una de las $\lambda_i, i = 1, 2, \dots, l$, tiene como fundamento que si para un cierto desplazamiento generado por el algoritmo de búsqueda de ceros, este va a resultar ventajoso, al menos la derivada direccional de $T(x, I)$ debe satisfacer que

$$\frac{dT(x, I)}{ds} < 0 \quad (46)$$

donde s es un desplazamiento a lo largo de la dirección de búsqueda.

Paso 4.- Condición de paro de la fase de tunelización.

Caso 1.- Según se desprende de lo mencionado en los puntos anteriores, el algoritmo de búsqueda de ceros se debe detener cuando se satisfaga que $T(x, I) = 0$. En la implementación actual del



algoritmo, esta última condición se ha reemplazado por

$$T(x, \Gamma) \leq \varepsilon_3 \quad (47)$$

la cual es una condición de paro más relajada que la anterior y que tiene la siguiente característica. Cualquier punto x que satisface la condición (47) puede ser empleado como punto de arranque de la siguiente fase de minimización. Por otro lado, el satisfacer la condición (47) requiere de menor esfuerzo computacional que pedir que se satisfaga $T(x, \Gamma) = 0$, debido esto último a que el espacio solución de (47), si existe, es al menos igual que el de la última condición, ya que mientras no se ha alcanzado el mínimo global de $f(x)$, siempre existe más de un cero de la función de tunelización.

Caso 2.- Este caso ocurre cuando para un cierto vector ε , mencionado en la sección 4.2.2, paso 2, no se encuentra ningún punto x^0 que satisfaga la condición (47) en un número finito de iteraciones N_s . Si esto sucede, se está en posibilidad de escoger otro vector ε , diferente del anterior, y repetir la fase de tunelización desde este nuevo punto de arranque, no teniéndose que determinar nuevamente el valor de λ_0 . Este proceso se puede repetir un número finito de veces N_ε . En la implementación actual del algoritmo, $N_\varepsilon = 5n + 1$, donde cada una de las N_ε se escogen de acuerdo a

- a) ε_1 se escoge al azar con $||\varepsilon_1|| \ll 1$
- b) ε_i , $i = 2, 3, \dots, 2n+1$, se escogen coincidentes con cada uno de los ejes coordenados, en ambas direcciones y $||\varepsilon_i|| \ll 1$
- c) ε_i , $i = 2n+2, 2n+3, \dots, 5n+1$, se escogen de forma tal que $a < x_1^* + \varepsilon_i < b$, eliminándose la restricción $||\varepsilon_i|| \ll 1$.

NOTA: Debido a que se requiere que la secuencia de puntos $\{x_k\}_{k=0}^\infty$ se mantenga siempre dentro de la región de interés durante toda la fase de tunelización, los desplazamientos producidos por el algoritmo de búsqueda de ceros se controlan en forma análoga al control descrito en la sección 4.1. Sin embargo cuando la secuencia de puntos generados en esta fase converge a una esquina de la región de interés, ésta se debe dar por terminada procediéndose entonces como en el caso 2 que se acaba de describir, empleando un nuevo vector ε .

4.3. Algunas Propiedades del Algoritmo de Tunelización.



A continuación se presentan las propiedades mas relevantes que exhibe el algoritmo de tunelización no todas las cuales están presentes en otros algoritmos que tienen el mismo propósito. Estas propiedades son las siguientes:

4.3.1.- Propiedad de descenso en la fase de minimización.

En la fase de minimización, cualquier algoritmo con propiedad de descenso local sobre la función, puede ser empleado, de donde se concluye que si la i -ésima fase de minimización usa como punto de arranque x_i^0 , el mínimo local que se encuentre, por ejemplo en x_i^* , cumplirá con la siguiente relación:

$$f(x_i^*) \leq f(x_i^0), \quad i = 1, 2, \dots, G \quad (48)$$

4.3.2.- Propiedad de descenso en la fase de tunelización.

Si en la i -ésima fase de tunelización, el algoritmo descrito en la sección 4.2 se usa como punto de arranque x_i^* , eventualmente se encontrará un punto x_{i+1}^0 , $i = 1, 2, \dots, G-1$, tal que $T(x_{i+1}^0, \Gamma) < \varepsilon_3$ y en el que se satisface la relación

$$f(x_{i+1}^0) \leq f(x_i^*), \quad i = 1, 2, \dots, G-1 \quad (49)$$

4.3.3.- Propiedad de descenso generalizada del algoritmo de tunelización.

Combinando las relaciones (48) y (49) se obtiene la relación general

$$f(x_1^0) \geq f(x_1^*) \geq f(x_2^0) \geq \dots \geq f(x_{G-1}^*) \geq f(x_G^0) \geq f(x_G^*) \quad (50)$$

o si se eliminan los puntos intermedios x_i^0 , se obtiene la propiedad de descenso generalizada del algoritmo de tunelización, la cual se escribe como

$$f(x_i^*) \geq f(x_{i+1}^*) \quad (51)$$

$$i = 1, 2, \dots, G-1$$

donde x_G^* indica la localización del mínimo global.

4.3.4.- Secuencia de puntos acotada.

Como se describió en las dos secciones anteriores, cada desplazamiento Δx_{k+1} producido en cualesquiera de las dos fases del algoritmo se controla de manera tal que cada nuevo punto que se genera queda dentro de la región de interés, es decir, $a \leq x_{k+1} \leq b$, donde $x_{k+1} = x_k + \Delta x_{k+1}$. Haciendo esto, se logra que la secuencia de puntos $\{x_k\}_{k=0}^{\infty}$ esté acotada.

4.3.5.- Condición final de paro.

Algunos autores (Ref. 2) han establecido condiciones necesarias y suficientes para que una función exhiba un mínimo global en x^* . Sin embargo, estas condiciones son difícilmente implementables en una computadora (sino es que imposible). En la presente tesis se propone la siguiente condición necesaria y suficiente para que $f(x)$ exhiba un mínimo global en x^* , en lugar de la mencionada en (Ref. 2)

$$T(x, \Gamma) > 0, \forall x \in \Omega = \{x \mid a \leq x \leq b\} \quad (52)$$

donde Γ contiene toda la información concerniente al último mínimo relativo encontrado en x^* .

Para todo fin práctico, la implementación de la condición (52) es equivalente a violar la condición establecida en la ecuación (47). Sin embargo, y debido a que la ecuación (47) se viola únicamente un número finito de veces, la condición de paro empleada, ecuación (47), resulta ser una condición necesaria pero no suficiente.

Una de las características más importantes del método de tunelización, es que posee una condición de paro implementable en una computadora, que si bien es cierto que únicamente es necesaria, también lo es que los otros métodos o no la poseen o no es implementable. Así por ejemplo, para el método de Branin no hay criterio alguno que indique cuando se debe parar la búsqueda, sucediendo lo mismo con el método de Goldstein y Price. Por otro lado, la implementación de la condición de paro que poseen los métodos de MRS y MMRS resulta no implementable en una computadora debido, principalmente, a la gran cantidad de memoria que se requiere para poder hacerlo así como por el problema de poder determinar una cota superior para la matriz de

segundas derivadas.

4.4.- Resumen del Algoritmo.

En esta sección, se presenta un resumen de cada uno de los pasos que se requieren durante el algoritmo de tunelización. El propósito de esto es permitir que el lector pueda fácilmente elaborar un programa de cómputo del algoritmo si se siguen las indicaciones que se dan enseguida.

4.4.1.- Fase de minimización.

Como se mencionó en la sección 4.1, cualquier algoritmo con propiedad de descenso local sobre $f(x)$, se puede emplear en esta fase. En esta tesis se empleó el método de gradiente ordinario, mostrándose además algunos resultados con gradiente conjugado y con Newton modificado (Apéndice A).

0) Seleccione los valores de ϵ_2 , la tolerancia en la convergencia de la fase de minimización, y N_b , el máximo número de bisecciones permitidas sobre el tamaño del paso.

1) Para una x dada, evalúe $f(x)$ y $f'_x(x)$

2) Si $f''_x(x)f'_x(x) < \epsilon_2$, se ha alcanzado un mínimo de $f(x)$, y deberá procederse con la fase de tunelización. De lo contrario, continúe con el paso 3.

3) Ajuste el tamaño del paso α igual a 1.

4) Evalúe el desplazamiento de referencia Δx_r , de acuerdo al método de minimización que se esté empleando.

5) Calcule el punto variado \bar{x} como

$$\Delta x = -\alpha \Delta x_r \quad (53)$$

$$\bar{x} = x + \Delta x \quad (54)$$

6) Si para el tamaño del paso que se seleccionó resulta que el punto variado \bar{x} viola alguna de las fronteras de la región, calcúlese un nuevo tamaño de paso de manera tal que no se viole ninguna de las fronteras. Este control sobre α se logra de la siguiente manera: Sea γ_i el valor que α debe tomar para alcanzar la frontera b_i , y θ_i el correspondiente para alcanzar la frontera a_i , esto es,

$$\gamma_i = \frac{x_i - b_i}{(\Delta x_r)_i} \quad (55)$$

$$\theta_i = \frac{x_i - a_i}{(\Delta x_r)_i} \quad (56)$$

para $(\Delta x_r)_i \neq 0$, $i = 1, 2, \dots, n$.

El valor correcto del tamaño del paso se toma como

$$\alpha = \min_i \{\gamma_i, \theta_i\} \quad (57)$$
$$\gamma_i > 0, \quad \theta_i > 0$$

continuándose con el paso α , armenos que el tamaño del paso obtenido mediante la ecuación (57) haya sido cero, en cuyo caso se pueden presentar las siguientes situaciones:

a) Si $\gamma_i = \theta_i = 0$ y $(\Delta x_r)_i \neq 0$, $i = 1, 2, \dots, n$, se ha alcanzado un mínimo en una de las esquinas de la región de interés y se deberá proceder como en el paso 2.

b) Si $\gamma_i = \theta_i = 0$ para algunos de los componentes del desplazamiento y para el resto de ellos se satisface que $\sum (\Delta x_r)_i^2 < \varepsilon_2$, se ha alcanzado un mínimo en la frontera, no en las esquinas, de la región y se deberá proceder como en el paso 2.

c) Si $\gamma_i = \theta_i = 0$ para algunos de los componentes del desplazamiento y para al menos uno de los restantes se cumple que $(\Delta x_r)_i$ no se anula, se hacen iguales a cero todos aquellos componentes del desplazamiento que tienden a violar alguna de las fronteras, se hace $\alpha = 1$ y se continua con el paso 7.

7) Si para el valor deado al tamaño del paso, con el cual no se violan ninguna de las fronteras de la región, se satisface que

$$f(\bar{x}) < f(x) \quad (58)$$

dicho valor del tamaño del paso se acepta como bueno y se repite el procedimiento desde el paso 1, tomando como x el valor calculado de \bar{x} y como $f(x)$ el de $f(\bar{x})$. Si la desigualdad (58) no se satisfce, se bisecta el valor del tamaño del paso y se repite el procedimiento

desde el paso 5. Los pasos 5-7 se repiten hasta N_b veces.

NOTA: Cuando se usan métodos de minimización que requieren de una selección óptima del tamaño del paso (gradiente conjugado, cuasi-Newton) cuando se viola una frontera, se determina el tamaño de paso como en la etapa 6 y se reinicia el proceso.

4.4.2.- Fase de tunelización.

En esta sección se muestran los pasos que se requieren para implementar la fase de tunelización cuando se usa como algoritmo de búsqueda de ceros el presentado por Miele et al. (Ref. 11).

0) Seleccione los valores de ϵ_3 , la tolerancia en la convergencia de la fase de tunelización; N_b , el número máximo de bisecciones que se permiten en el tamaño del paso; N_g , el número máximo de pasos de búsqueda que se pueden dar después de que se ha seleccionado un vector ϵ ; N_e , el número máximo de vectores ϵ que se pueden tomar para iniciar la búsqueda de un cero; β_R , el valor límite del tamaño del paso para determinar los valores de λ_0 y λ_i , $i = 1, 2, \dots, l$ (se toma igual a 2^{-N_b}); λ_{\max} , el máximo valor que λ_1 puede alcanzar, y λ_0^* el máximo valor que λ_0 puede alcanzar. Además de lo anterior, se debe dar un tiempo máximo T_{\max} hasta el cual se le permite correr a un algoritmo para que encuentre el mínimo global.

1) La fase de tunelización se empieza comparando las características del último mínimo encontrado, $f(x^*)$, localizado en x^* , con las del mínimo anterior, localizado en x_1^* , de acuerdo a las siguientes reglas:

a) Si es la primera tunelización que se va a llevar a cabo, se hacen $l = 1$, $x_1^* = x^*$ y $f^* = f(x^*)$ y se procede con el paso 2.

b) Si la tunelización que se va a llevar a cabo no es la primera, existen las siguientes posibilidades:

b1) Si $f(x^*) = f^*$, se hacen $l = l + 1$ y $x_1^* = x^*$. Esta situación implica que el nuevo mínimo está localizado al mismo nivel que los $(l - 1)$ mínimos anteriores.

b2) Si $f(x^*) < f^*$, se hacen $l = 1$, $x_1^* = x^*$ y $f^* = f(x^*)$, procediéndose al paso 2. Cuando esto sucede, se están ignorando todos

los mínimos encontrados con anterioridad, así como sus localizaciones.

2) Se escoge un vector ε , haciendo $x_m = x_1^*$, $\lambda_0 = 0$ y se empieza la fase con $x = x_1^* + \varepsilon$.

3) Si el valor de λ_1 todavía no se ha determinado, se procede con los pasos 4-8, empezando con $\lambda_1 = 1$. Si λ_1 ya se determinó, se procede con los pasos 9-28.

4) Evalúense $T(x, \Gamma)$ y $T_x(x, \Gamma)$

5) Se evalúa el paso unitario de desplazamiento $A = \Delta x / \beta$, de acuerdo a

$$A = - \frac{T(x, \Gamma) T_x(x, \Gamma)}{T_x^T(x, \Gamma) T_x(x, \Gamma)} \quad (59)$$

6) Para un tamaño de paso igual al de referencia (2^{-N_b}) se calcula

$$\bar{x} = x + \beta_R A \quad (60)$$

7) Se procede en forma similar al paso 6 de la fase de minimización para verificar que \bar{x} no viola ninguna de las fronteras. Si \bar{x} viola alguna frontera, se procede con el paso 2 hasta un límite de N_ε veces, dándose por terminada la fase de tunelización si esto sucede.

8) Una vez verificado que \bar{x} no viola las fronteras, se evalúa $T(\bar{x}, \Gamma)$. Si se satisface que

$$T(\bar{x}, \Gamma) < T(x, \Gamma) \quad (61)$$

se acepta el valor de λ_1 como correcto, ya que la propiedad de descenso (61) se satisface y se continúa con el paso 9. Si la desigualdad (61) no se satisface, se incrementa el valor de λ_1 en $\Delta \lambda_1 = 1$ y se repite el procedimiento desde el paso 4. Los pasos 4-8 se repiten hasta que λ_1 ha alcanzado el valor λ_{\max} , en cuyo caso el proceso de tunelización se da por terminado, o bien hasta que la relación (61) se satisface. Se debe señalar que en ninguno de los ejemplos analizados se alcanzó el valor λ_{\max} .

9) Una vez que el valor correcto de λ_1 ha sido determinado, la última \bar{x} calculada en el paso 6 se toma como punto de arranque x en los pasos que siguen.

10) Evalúense $T(x, \Gamma)$ y $T_x(x, \Gamma)$

11) Se evalúa el paso unitario de desplazamiento $A = \Delta x / \beta$, de acuerdo a la ecuación (59) y el punto variado \bar{x} de acuerdo a la ecuación (60), verificándose, como ya se indicó, que \bar{x} esté en la región de interés.

12) Se evalúa $T(\bar{x}, \Gamma)$

13) Si en el punto variado \bar{x} se satisface la desigualdad (61), se continúa con el paso 15.

14) Si se violó la desigualdad (61), se incrementa el valor de λ_0 en $\Delta\lambda_0 = 0.01$ y se repite el procedimiento desde el paso 10, hasta que la ecuación (61) se satisface, o bien, λ_0 alcanza su valor máximo λ_0^* , lo que indica que el algoritmo no ha podido producir la propiedad de descenso desada, continuándose entonces con el paso 28.

15) Si $\lambda_0 = 0$, o el valor de λ_0 se modificó en el paso 14, se continúa con el paso 20, tomando como valor correcto para el desplazamiento, el calculado en el paso 11.

16) Defínase Γ^* igual a Γ , en todos sus componentes excepto λ_0 , la cual se fija en cero.

17) Evalúese un nuevo paso unitario B como

$$B = \frac{T(x, \Gamma^*) T_x(x, \Gamma^*)}{T_x^T(x, \Gamma^*) T_x(x, \Gamma^*)} \quad (62)$$

18) Actualícese el valor del desplazamiento unitario, de acuerdo a

$$A_{correcta} = \begin{cases} A & \text{(ecuación 60); si } A^T B \leq 0 \\ B & \text{(ecuación 62); si } A^T B > 0 \end{cases} \quad (63)$$

19) Actualícese el valor de λ_0 de acuerdo a

$$\lambda_0 = \begin{cases} \lambda_0 & \text{si } A^T B \leq 0 \\ 0 & \text{si } A^T B > 0 \end{cases} \quad (64)$$

20) Se procede a determinar \bar{x} , tomando como tamaño de paso $\beta=1$

21) Para los valores actuales de β y A, se evalúan

$$\Delta x = \beta A \quad (65)$$

$$\bar{x} = x + \Delta x \quad (66)$$

22) Se procede como en el paso 6 de la fase de minimización para verificar que \bar{x} no viola ninguna de las fronteras. Si sucede que \bar{x} coincide con una esquina de la región, se procede con el paso 28.

23) Para el valor de \bar{x} calculada de acuerdo a (66), o en el paso 22, se evalúa $T(\bar{x}, I)$. Si el valor de $T(\bar{x}, I)$ satisface la desigualdad (61), se acepta el valor de \bar{x} como correcto. Si la desigualdad (61) no se satisface, se bisecta el valor de β y se repite el procedimiento desde el punto 21. Si se alcanza el máximo de bisecciones N_b , se procede con el paso 28.

24) Una vez que la desigualdad (61) se satisfizo y además se cumple que

$$T(\bar{x}, I) < \epsilon_3 \quad (67)$$

se ha tenido éxito en localizar no necesariamente un cero de $T(x, I)$, sino un punto admisible que puede ser empleado como punto de arranque de la siguiente fase de minimización. Si (67) no se cumple, se continúa con el paso 25.

25) Se actualiza el valor de x_m de acuerdo a

$$x_m = \begin{cases} x; & \text{si } \|\bar{x} - x\| < 1 \\ y; & \text{si } \|\bar{x} - x\| \geq 1, \text{ donde } y \text{ se determina} \\ & \text{como sigue: empezando con } \xi = 1 \text{ y usando} \\ & \text{un esquema de bisección, se encuentra} \\ & \text{el primer valor de } \xi \text{ en} \\ & y = \xi x + (1 - \xi)\bar{x} \\ & \text{tal que } \|\bar{x} - y\| < 1 \end{cases} \quad (68)$$

$$(69)$$

26) El nuevo nominal x , se hace igual al último punto variado calculado en el paso 21, y se actualiza x_m como $x_m = \bar{x}_m$.

27) Los pasos 9-26 se repiten hasta que la desigualdad (67) se satisface, o bien, se ha alcanzado el máximo número de iteraciones N_s .

28) Si después de N_s iteraciones no se logró que la desigualdad (67) fuera satisfecha, se escoge un nuevo vector ϵ y se repite el pro-

cedimiento desde el paso 9, empezando con $x = x_1^* + \varepsilon$. Si después de N_ε iteraciones no se logra que la desigualdad (67) se satisfaga, el proceso se dá por terminado, indicando esto que, muy posiblemente, $T(x, \Gamma)$ no exhiba cero alguno y por lo tanto se ha localizado el mínimo global de $f(x)$ en los puntos x_i^* , $i = 1, 2, \dots, l$, con un valor de la función igual a f^* .

V.- EXPERIMENTOS NUMERICOS Y RESULTADOS

El propósito de esta sección es mostrar las características de los métodos presentados en los capítulos III y IV. Para tal fin, se resolvieron 18 ejemplos numéricos en una computadora CDC-6400 usando aritmética de precisión sencilla (14 dígitos) y programándose todos los algoritmos en FORTRAN IV.

5.1.- Condiciones de Paro.

5.1.1.- Algoritmo de tunelización.

Los valores asignados a los diferentes parámetros descritos en los pasos 0 (cero) de las secciones 4.4.1 y 4.4.2 fueron los siguientes:

a) Fase de minimización.

El criterio de paro se escogió como

$$\epsilon_1 = 10^{-9} \quad (70)$$

y el criterio de no convergencia se estableció en

$$N_D = 30 \quad (71)$$

b) Fase de tunelización.

El criterio de convergencia se escogió como

$$\epsilon_3 = 10^{-3} \quad (72)$$

y los de no convergencia en

$$N_D = 30 \quad (73a)$$

$$N_S = 100 \quad (73b)$$

$$\lambda_{\text{máx}} = 5 \quad (73c)$$

$$\lambda_0^* = 1 \quad (73d)$$

$$N_C = 5n + 1 \quad (73e)$$

donde n denota la dimensionalidad del problema.

Los parámetros de activación alrededor de los círculos localizados en x_i^* , $i = 1, 2, \dots, l$, se fijaron en

$$\epsilon_1 = 0.05 \quad (74)$$

$$d = 1 \quad (75)$$

mientras que el paso de referencia se fijó en

$$\beta_R = 2^{-20} \quad (76)$$

5.1.2.- Versión de Hardy del método de Branin.

En este algoritmo un punto estacionario se alcanza si

$$\bar{f}_x^T(x) f_x(x) \leq \epsilon_2 = 10^{-6} \quad (77)$$

observándose que, por construcción del método, se pueden alcanzar máximos, mínimos o puntos silla (nunca dos mínimos o dos máximos consecutivos). Además, se consideró que $H(x)$ es casi-singular si

$$\det(H(x)) \leq 10^{-5} \quad (78)$$

5.1.3.- Método de Goldstein y Price.

Debido a que este método está constituido de unicamente fases de minimización consecutivas, se usó la siguiente condición de paro: Si $g(x)$ denota el gradiente de $f(x)$, o de cualquiera de sus funciones auxiliares $f_1(x)$, $f_2(x)$, ..., se logra convergencia si

$$g^T(x)g(x) \leq 10^{-6} \quad (78)$$

El criterio de no convergencia se fijó cuando el método de minimización requiere de más de 30 bisecciones en el tamaño del paso para lograr que la función objetivo descienda.

5.1.4.- MRS

Como se describió en el capítulo III, este algoritmo está constituido por una fase de minimización y un generador de números aleatorios que dá la localización de un nuevo punto de arranque para la siguiente minimización. Las condiciones de convergencia y no convergencia se establecieron, por tanto, iguales a las descritas en 5.1.1. (a).

5.1.5.- MMRS

Además de las anotaciones hechas para el método MRS, este método emplea la siguiente condición de reinicio. Si

$f(x^*)$ denota el valor de $f(x)$ en el último mínimo encontrado en una fase de minimización, la siguiente fase de minimización no se empieza sino hasta que

$$f(x^o) - f(x^*) \leq 10^{-3} \quad (79)$$

donde x^o es uno de los números aleatorios generados por el algoritmo. Si la desigualdad (79) no se satisface para algún punto x^o , se continúan generando números hasta que la condición (79) se satisface.

5.2.- Ejemplos.

En esta sección se describen los ejemplos que se emplearon para comparar los algoritmos de minimización global descritos en los capítulos anteriores. En cada uno de los ejemplos que se incluyen se señalan las siguientes características:

- a) región de interés
- b) número de mínimos relativos
- c) número y localización de mínimos globales

Ejemplo 1.- Función de Goldstein (Ref. 6)

$$f(x) = x^6 - 15x^4 + 27x^2 + 250 \quad (80)$$

$$-4 \leq x \leq 4$$

Esta función exhibe tres mínimos locales, dos de los cuales son también globales y están localizados en

- | | | |
|-------------|--------------|------|
| a) $x = -3$ | $f(x) = 7$ | |
| b) $x = 0$ | $f(x) = 250$ | (81) |
| c) $x = 3$ | $f(x) = 7$ | |

Ejemplo 2.- Función de Shubert (Ref. 15)

$$f(x) = - \sum_{i=1}^5 \text{icos}\{(i+1)x + i\} \quad (82)$$

$$-10 \leq x \leq 10$$

Esta función exhibe 19 mínimos locales, tres de los cuales son también globales, siendo la localización de éstos últimos

- | | |
|---------------------|------|
| a) $x = 7.7083144$ | |
| b) $x = -1.4251289$ | (83) |
| c) $x = 4.8580559$ | |

El valor de la función en estos puntos es $f(x) = -12.870885$.

Los ejemplos 3,4 y 5 están generados a partir de la expresión general

$$f(x,y) = \left(\sum_{i=1}^5 \text{icos}\{(i+1)x+i\} \right) \left(\sum_{i=1}^5 \text{icos}\{(i+1)y+i\} \right) + \beta \{ (x + 1.42513)^2 + (y + 0.80032)^2 \} \quad (84)$$

$-10 \leq x, y \leq 10$

y se observa que esta función se ha obtenido a partir de (82).

Ejemplo 3. - Función de Shubert en dos dimensiones (Ref. 9)

Caso 1.- $\beta = 0$. Con este valor de β , la función exhibe 760 mínimos locales, 18 de los cuales son también globales. En la localización de éstos últimos, la función adquiere el valor

$f(x,y) = -186.7309$, siendo las coordenadas las siguientes:

	<u>x</u>	<u>y</u>
a)	-7.08350	-7.70831
b)	-0.80032	-7.70831
c)	+5.48286	-7.70831
d)	-7.70831	-7.08350
e)	-1.42513	-7.08350
f)	+4.85805	-7.08350
g)	-7.08350	-1.42513
h)	-0.80032	-1.42513
i)	+5.48286	-1.42513
j)	-7.70831	-0.80032
k)	-1.42513	-0.80032
l)	+4.85805	-0.80032
m)	-7.08350	+4.85805
n)	-0.80032	+4.85805
ñ)	+5.48286	+4.85805
o)	-7.70831	+5.48286
p)	-1.42513	+5.48286
q)	+4.85805	+5.48286

Las figuras 4-7 muestran diferentes detalles de esta función de acuerdo a lo siguiente: Fig. 4, $-f(x,y)/100 \geq 1$; Fig. 5, $-f(x,y)/100 \geq 0.5$; Fig. 6, $-f(x,y)/100 \geq 0$; Fig. 7, $-f(x,y)/100$

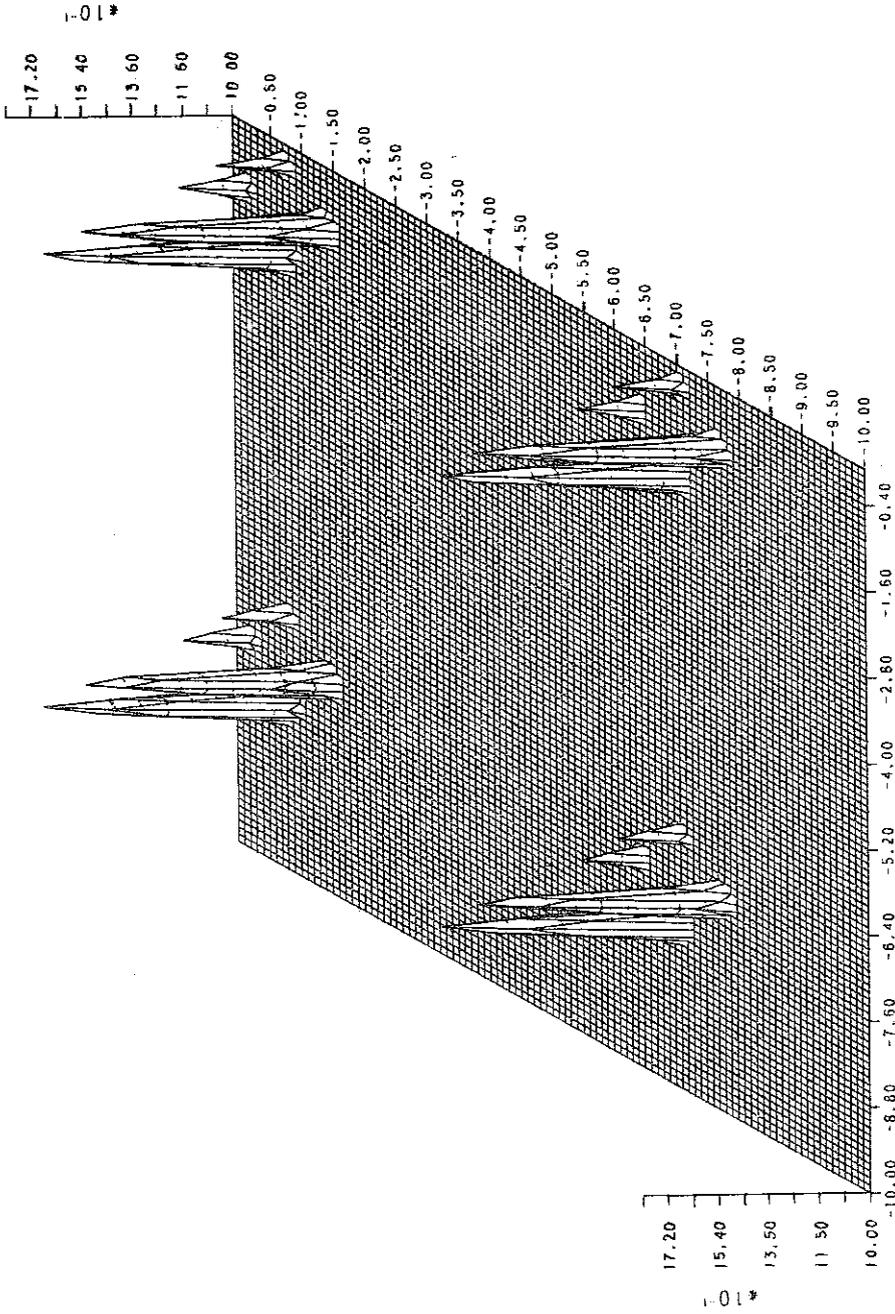


Figura 4.- Función de Shubert en dos dimensiones, $\beta = 0$
 $-f(x,y)/100 \geq 1$, $-10 \leq x,y \leq 0$

TESIS CON
FALLA DE ORIGEN

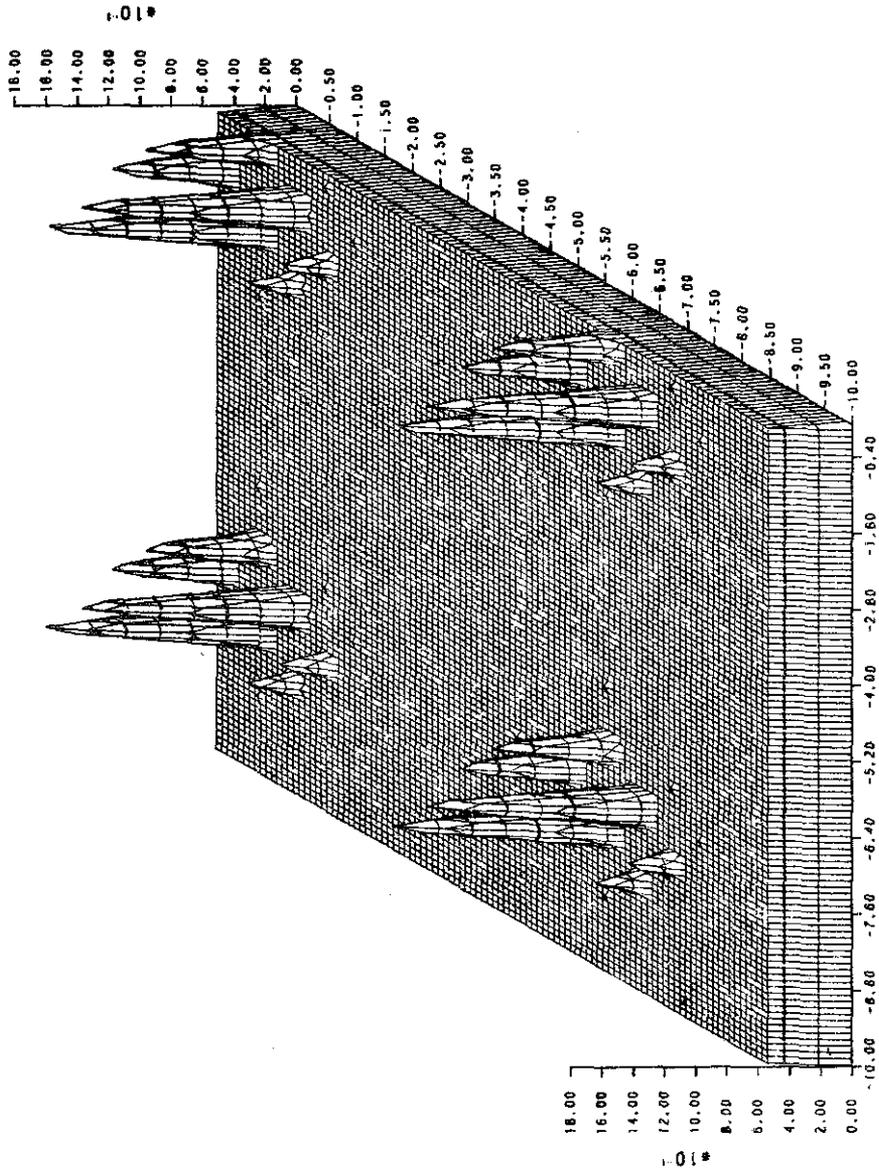


Figura 5.- Función de Shubert en dos dimensiones, $\beta = 0$

$$-f(x,y)/100 \geq 0.5 \quad -10 \leq x,y \leq 0$$

TESIS CON
FALLA DE ORIGEN

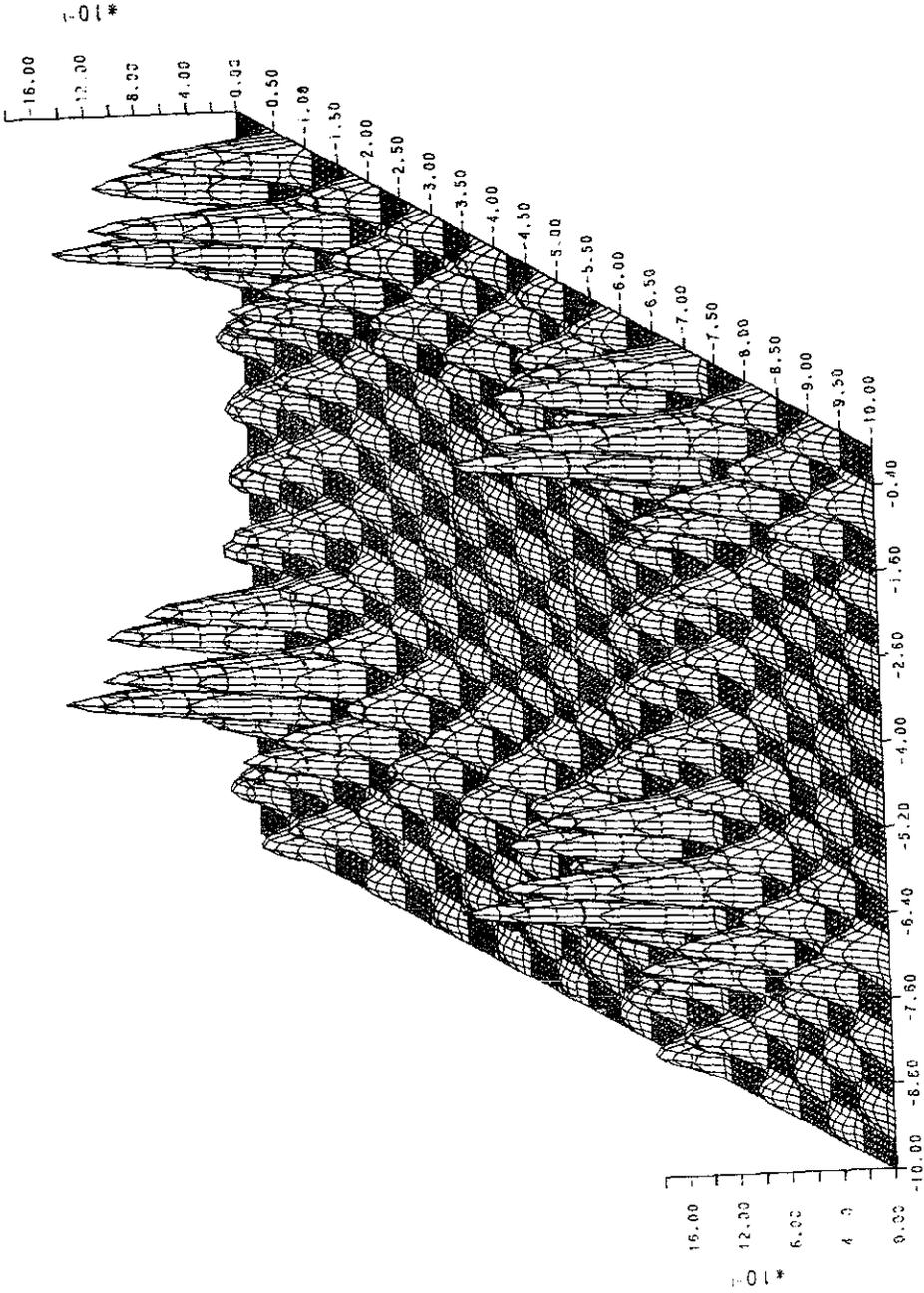


Figura 6.- Función de Shubert en dos dimensiones, $\beta = 0$
 $-f(x,y)/100 \geq 0$ $-10 \leq x, y \leq 0$

TESIS CON
FALLA DE ORIGEN

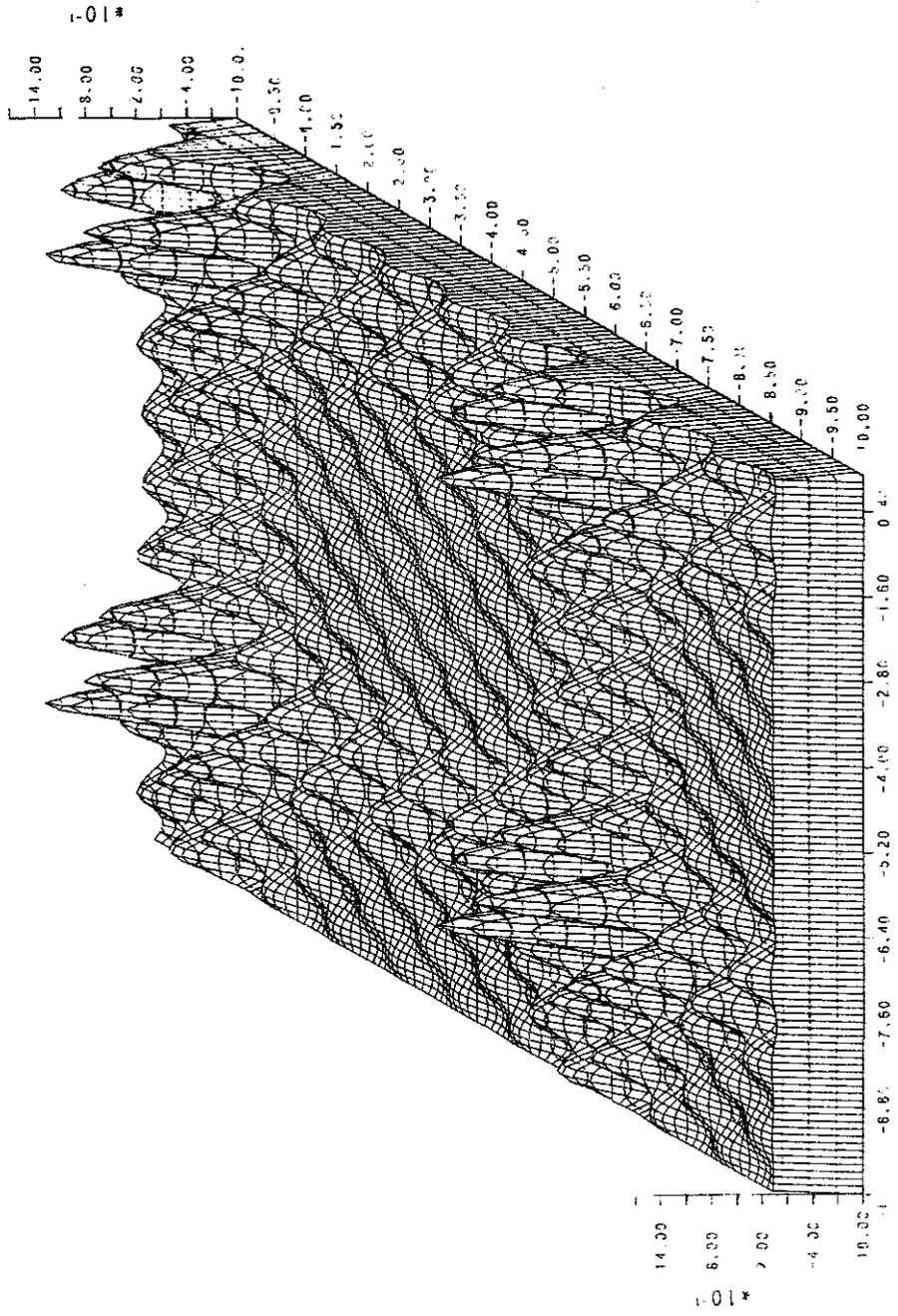


Figura 7.- Función de Shubert en dos dimensiones, $\beta = 0$
 $-10 \leq x, y \leq 0$
 $-f(x,y)/100$

TESIS CON
FALLA DE ORIGEN

Todas estas figuras, abarcan únicamente una cuarta parte de la región de interés señalada en la ecuación (84).

Ejemplo 4.- Función de Shubert en dos dimensiones

Caso 2.- $\beta = 0.5$. Cuando se usa este valor de β , la función posee las mismas características generales que la función del ejemplo 3, excepto que 17 de los mínimos globales desaparecen. El mínimo global que se preserva está localizado en $x = -1.42513$, $y = -0.80032$, con $f(x,y) = -186.7309$

Ejemplo 5.- Función de Shubert en dos dimensiones

Caso 3.- $\beta = 1.0$. Cuando se usa este valor de β , la función posee las mismas características que las del ejemplo 4, incluyendo la localización y el valor de la función en el mínimo global.

El propósito de haber usado varios valores de β es el de cambiar la curvatura de la función de Shubert, pudiéndose así estudiar el efecto de tener varios mínimos globales o un solo mínimo global con varios casi-globales.

Ejemplo 6.- Función del camello de seis jorobas (Ref 4)

$$f(x,y) = (4 - 2.1x^2 + x^4/3)x^2 + xy + (-4 + 4y^2)y^2 \quad (85)$$
$$-3 \leq x \leq 3, \quad -2 \leq y \leq 2$$

Esta función exhibe 6 mínimos locales (de aquí su nombre), dos de los cuales son también globales y están localizados en

$$\begin{aligned} \text{a)} \quad & x = -0.08983 \quad y = 0.7126 \\ \text{b)} \quad & x = 0.08983 \quad y = -0.7126 \end{aligned} \quad (86)$$

siendo el valor de la función en estos puntos $f(x,y) = -1.0316285$.

En la figura 8 se muestra la gráfica de esta función; en la figura 9 se muestra el resultado de ver la figura 8 en un espejo colocado en $y = 1.5$. Además de estas dos figuras, y debido a la simplicidad de la función, en la figura 10 se muestra $T(x,\bar{I})$ cuando $l = 1$, $x_1^* = (0.08983, -0.7126)$, $\lambda_0 = 0$, $\lambda_1 = 1$, $f^* = -1.0316285$, $x = x_1^* + \epsilon$; en la figura 11 se muestra el análogo de la figura 9. En estas dos últimas figuras, se señala la localización de $T(x,\bar{I}) = 0$

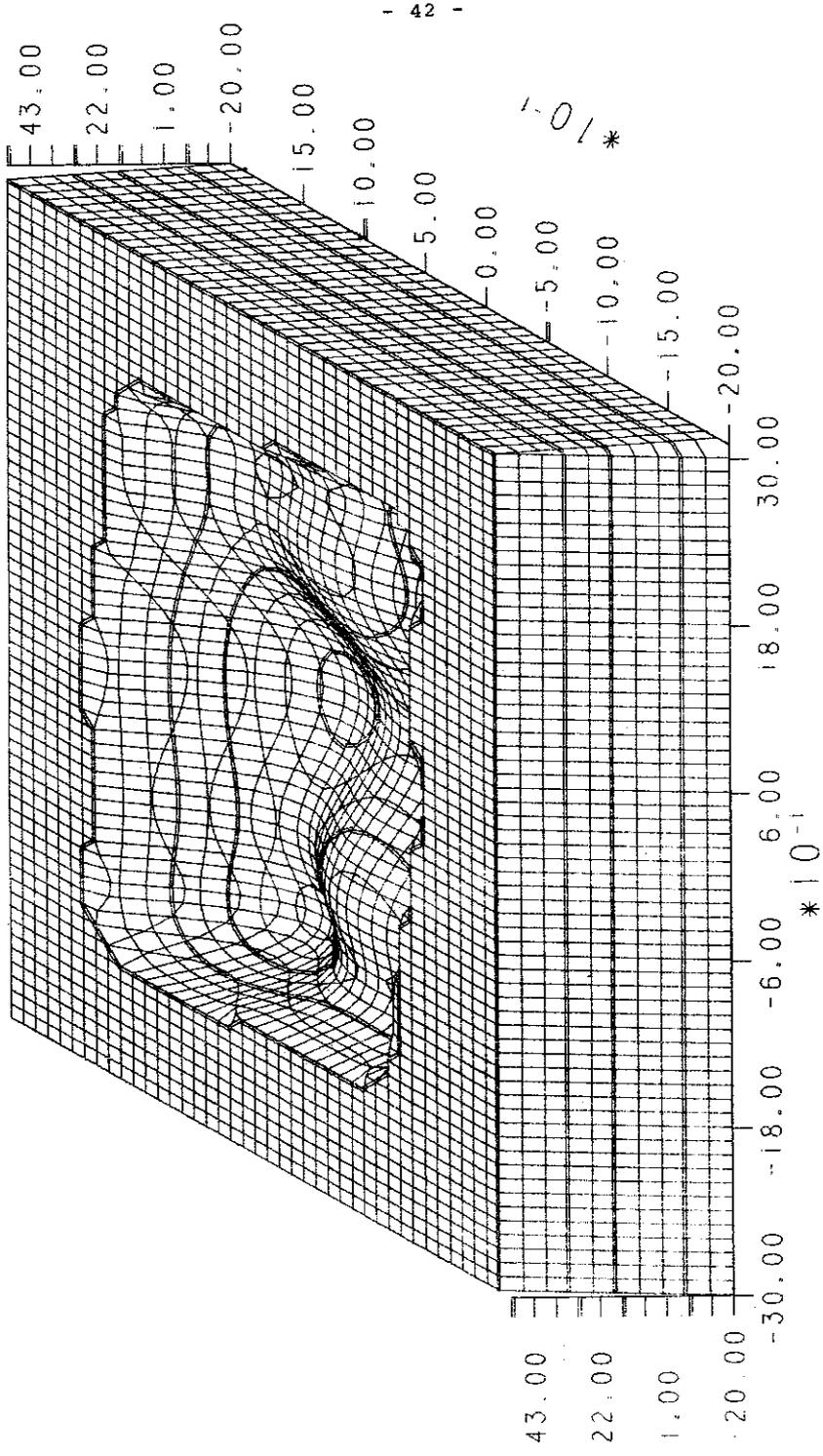


Figura 8.- Función del camello de seis jorobas

10f(x,y)

TESIS CON
 FALLA DE ORIGEN

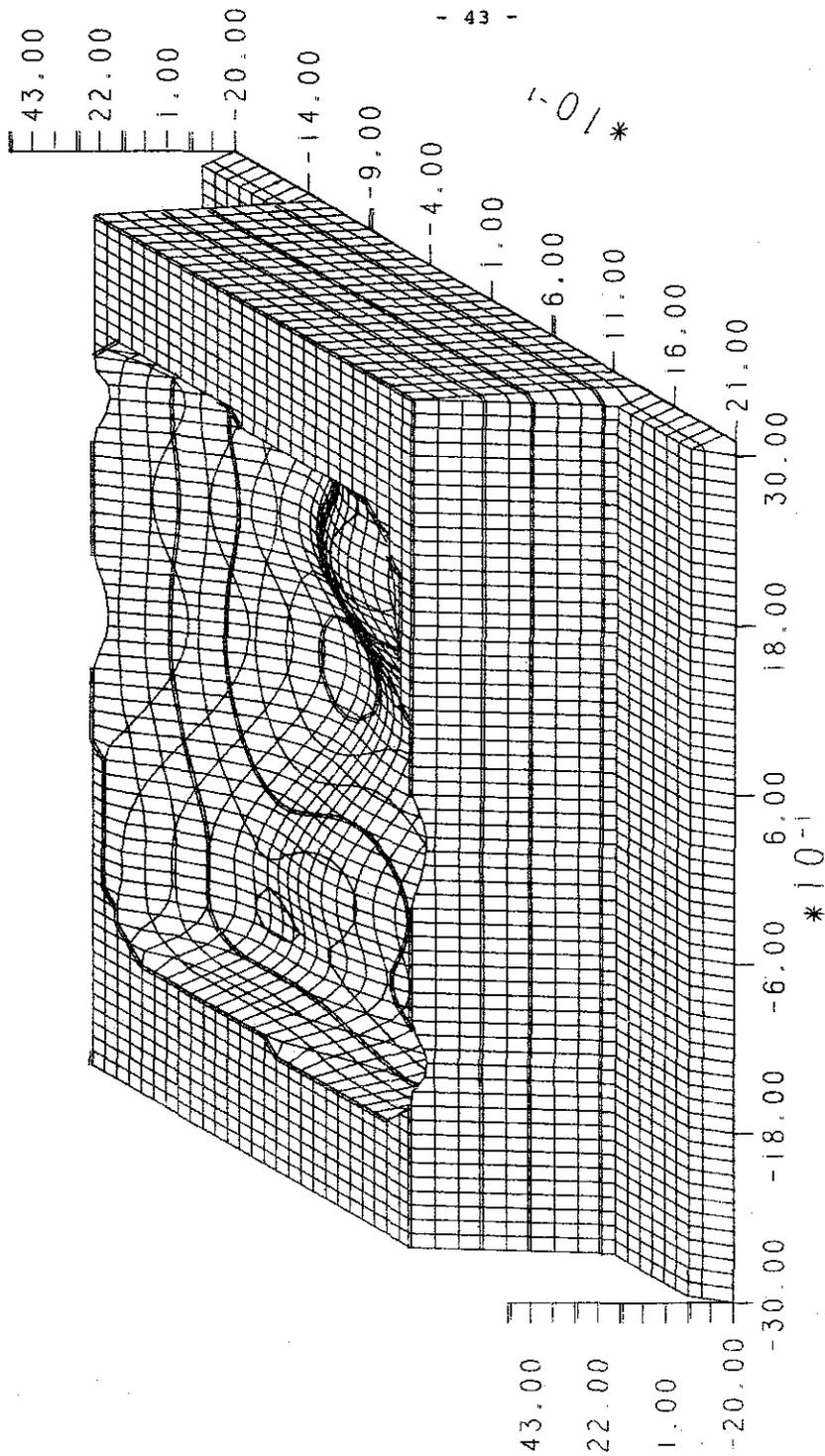


Figura 9.- Función del camello de seis jorobas
 Reflejo de la figura 8 en $y = 1.5$
 $10f(x,y)$

TESIS CON
 FALLA DE ORIGEN

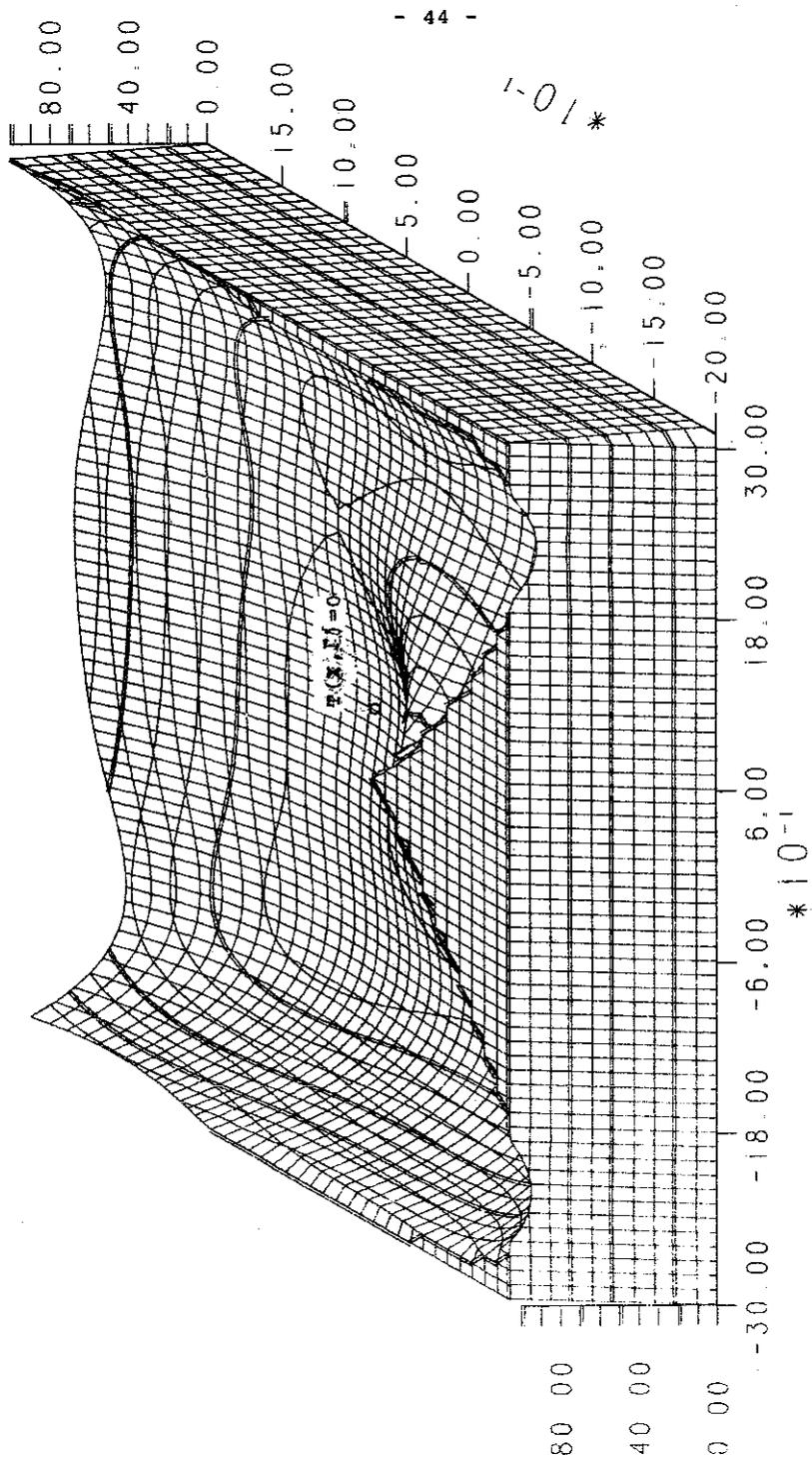


Figura 10.- $F(x, \Gamma)$ para el camello de seis jorobas
 $10T(x, \Gamma)$

TESIS CON
 FALLA DE ORIGEN

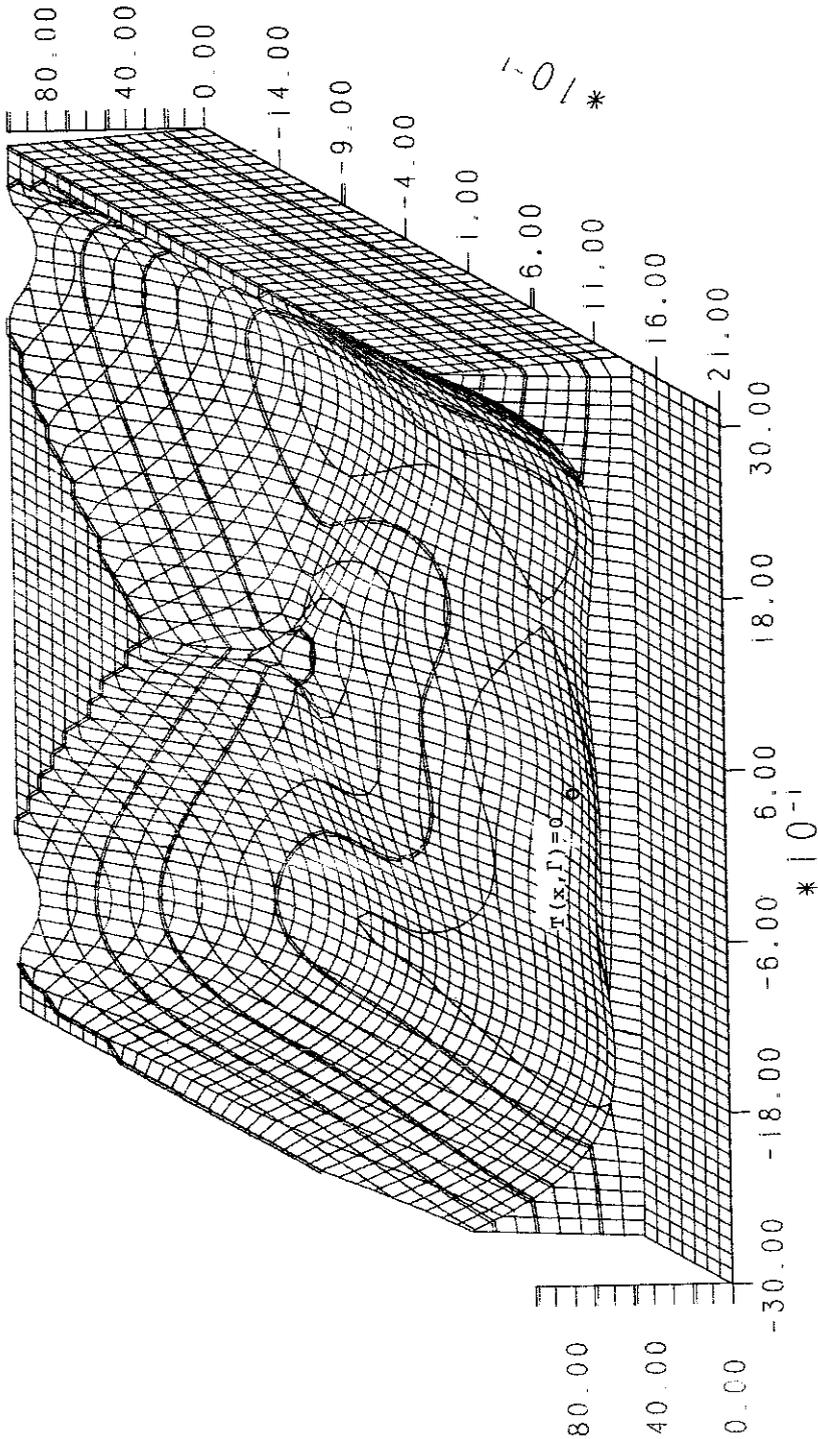


Figura 11.- $\tau(x, \Gamma)$ para el camello de seis jorobas
Reflejo de la figura 10 en $y = 1.5$
 $10\tau(x, \Gamma)$

TESIS CON
FALLA DE ORIGEN

Los ejemplos 7 - 9 están tomados de la fórmula general

$$f(x) = \pi \{ k \sin^2 \pi y_1 + \sum_{i=1}^{n-1} \{ (y_i - A)^2 (1 + k \sin^2 \pi y_{i+1}) \} + (y_n - A)^2 \} / n \quad (87)$$

$$\text{donde } y_i = 1 + (x_i - 1)/4 \quad (88)$$

$$-10 \leq x_i \leq 10, i = 1, 2, \dots, n$$

donde las constantes k y A se han fijado en 10 y 1 respectivamente, indicando n la dimensionalidad del problema.

La función (87) exhibe del orden de $(20/4)^n$ mínimos locales, con únicamente un mínimo global, localizado en

$$x_i = 1, i = 1, 2, \dots, n \quad (89)$$

siendo el valor de la función en este punto $f(x) = 0$, sin importar la dimensionalidad del problema.

En las figuras 12 y 13 se muestra la función (87) con diferente resolución. En éstas figuras se han suprimido los siguientes factores: (π/n) en la ecuación (87) y $(1/4)$ de la ecuación (88).

Ejemplo 7.- Ecuación (87) con $n = 2$

Ejemplo 8.- Ecuación (87) con $n = 3$

Ejemplo 9.- Ecuación (87) con $n = 4$

Los ejemplos 10 - 12 están tomados de la fórmula general

$$f(x) = \pi \{ k \sin^2 \pi x_1 + \sum_{i=1}^{n-1} \{ (x_i - A)^2 (1 + k \sin^2 \pi x_{i+1}) \} + (x_n - A)^2 \} / n \quad (90)$$

$$-10 \leq x_i \leq 10, i = 1, 2, \dots, n$$

En este caso, las constantes se fijaron en $k = 10$, $A = 1$ y n indica la dimensionalidad del problema.

Se observa que esta función es del mismo tipo que la definida por la ecuación (87), pero con una frecuencia cuatro veces más alta, de aquí que el número de mínimos relativos que exhibe la función (90) sea del orden de $(10)^n$, con un solo mínimo global en $x_i^* = 1, i = 1, 2, \dots, n$ y $f(x) = 0$.

Ejemplo 10.- Ecuación (90) con $n = 5$

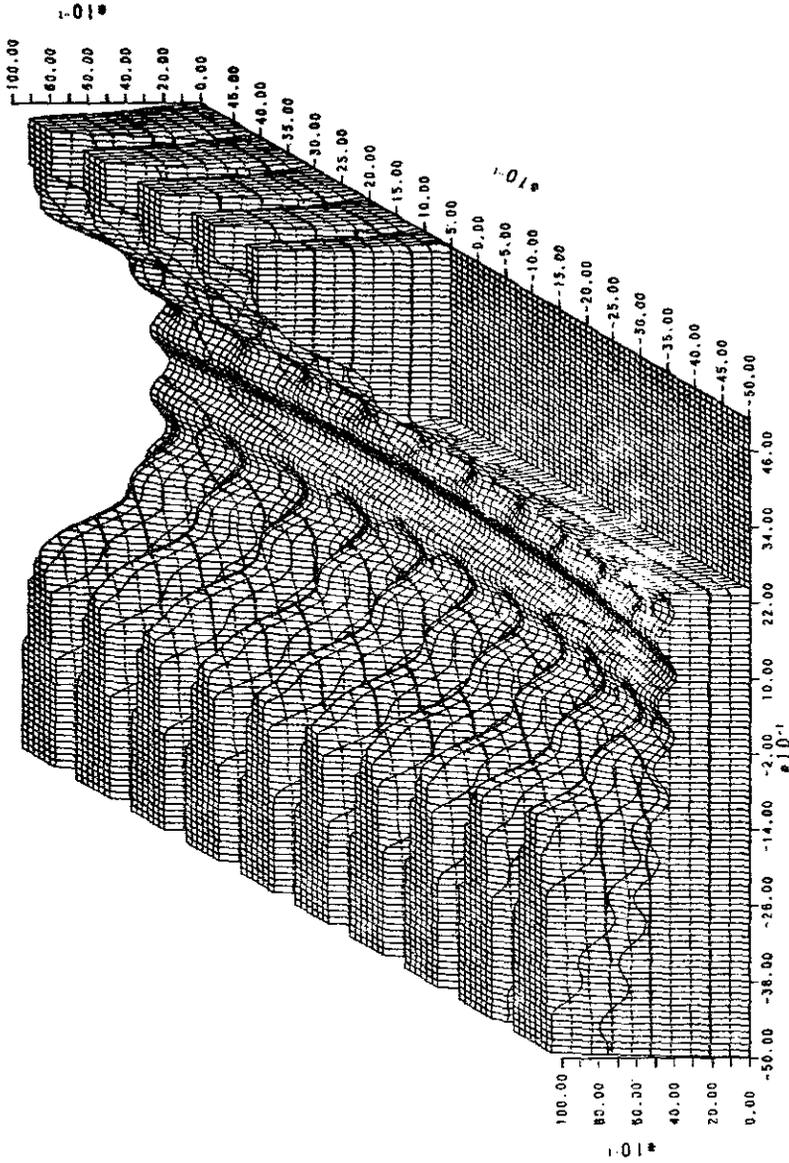


Figura 12.- Ecuación (87) para $n = 2$
(Ver notas en el texto)
 $-5 \leq x_1, x_2 \leq 5$

TESIS CON
FALLA DE ORIGEN

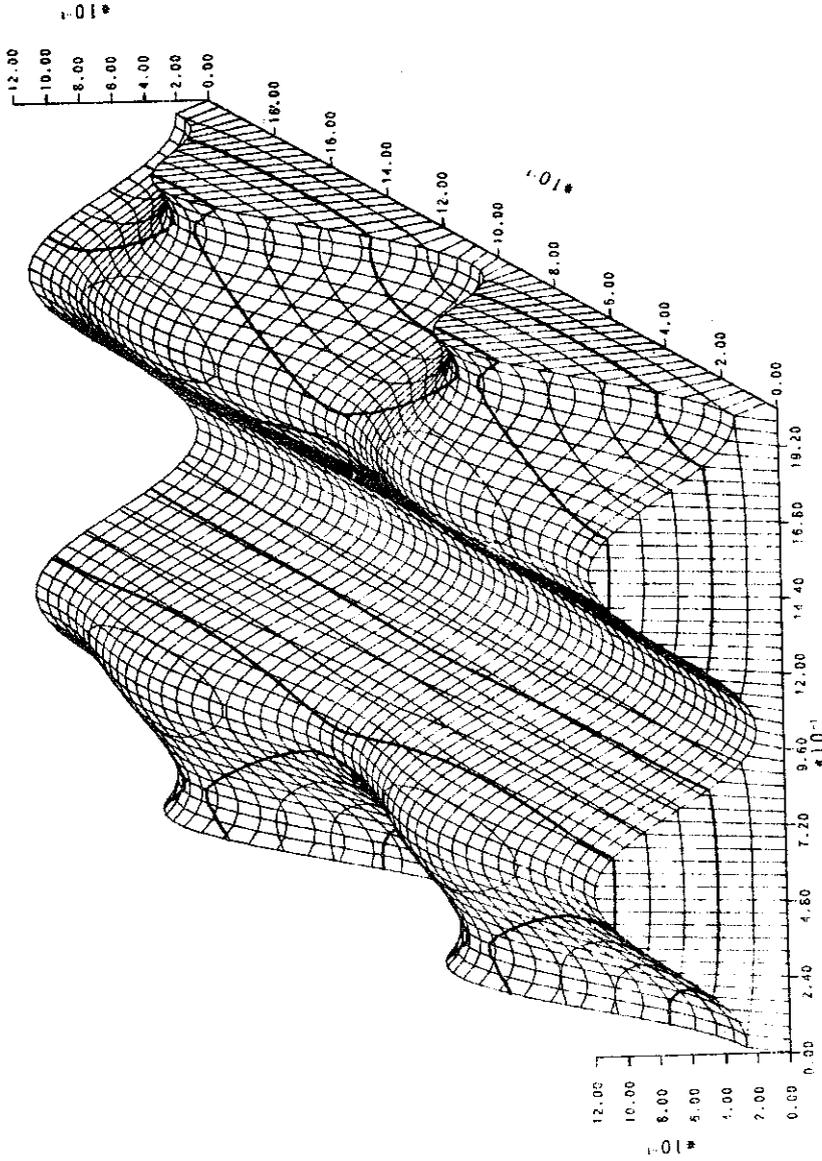


Figura 13.- Ecuación (B7) para $n = 2$

(Ver notas en el texto)

$$0 \leq x_1, x_2 \leq 2$$

TESIS CON
FALLA DE ORIGEN

Ejemplo 11. - Ecuación (90) con $n = 8$

Ejemplo 12. - Ecuación (90) con $n = 10$

Los ejemplos 13 - 18 están generados a partir de la fórmula general

$$f(x) = k_1 \sin^2 \pi l_0 x_1 + k_1 \sum_{i=1}^{n-1} \{(x_i - A)^2 (1 + k_0 \sin^2 \pi l_0 x_{i+1})\} \\ + k_1 (x_n - A)^2 (1 + k_0 \sin^2 \pi l_1 x_n) \quad (91)$$

donde las constantes de la ecuación se han fijado en $k_0 = 1$, $k_1 = 0.1$
 $A = 1$, $l_0 = 3$ y $l_1 = 2$.

Los ejemplos que se generan a partir de esta última ecuación exhiben del orden de $(30)^n$ mínimos locales, con un solo mínimo global, localizado en $x_i^* = 1$, $i = 1, 2, \dots, n$ y $f(x) = 0$. En las figuras 14 y 15 se muestra la ecuación (91) para $n = 2$ en los intervalos $0 \leq x_1, x_2 \leq 2$ y $-1 \leq x_1, x_2 \leq 3$, respectivamente.

Finalmente un comentario acerca de los ejemplos. Resulta obvia la similitud entre las ecuaciones (87) y (91), donde la única diferencia esencial está en el último término de ambas ecuaciones. El efecto que tiene el último término de la ecuación (91), es eliminar el valle longitudinal que exhibe la ecuación (87) el cual está claramente definido en las figuras 12 y 13.

Ejemplo 13. - Ecuación (91) con $n = 2$, $-10 \leq x_i \leq 10$

Ejemplo 14. - Ecuación (91) con $n = 3$, $-10 \leq x_i \leq 10$

Ejemplo 15. - Ecuación (91) con $n = 4$, $-10 \leq x_i \leq 10$

Ejemplo 16. - Ecuación (91) con $n = 5$, $-5 \leq x_i \leq 5$

Ejemplo 17. - Ecuación (91) con $n = 6$, $-5 \leq x_i \leq 5$

Ejemplo 18. - Ecuación (91) con $n = 7$, $-5 \leq x_i \leq 5$

NOTA: Los ejemplos 16 - 18 poseen $(15)^n$ mínimos locales en lugar de los $(30)^n$ como se indicó con anterioridad. Esto se debe al tamaño del intervalo de interés que se está usando.

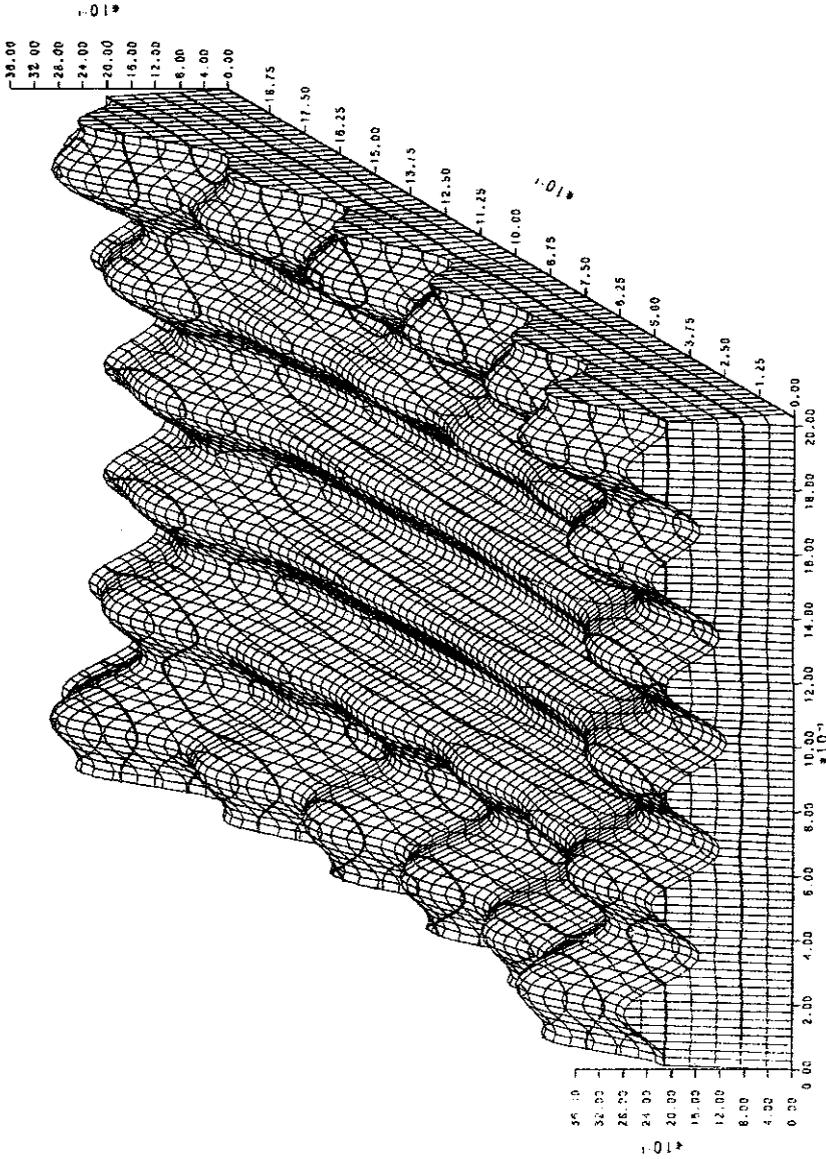


Figura 14.- Ecuación (91), $n = 2$

$$0 \leq x_1, x_2 \leq 2$$

TESIS CON
FALLA DE ORIGEN

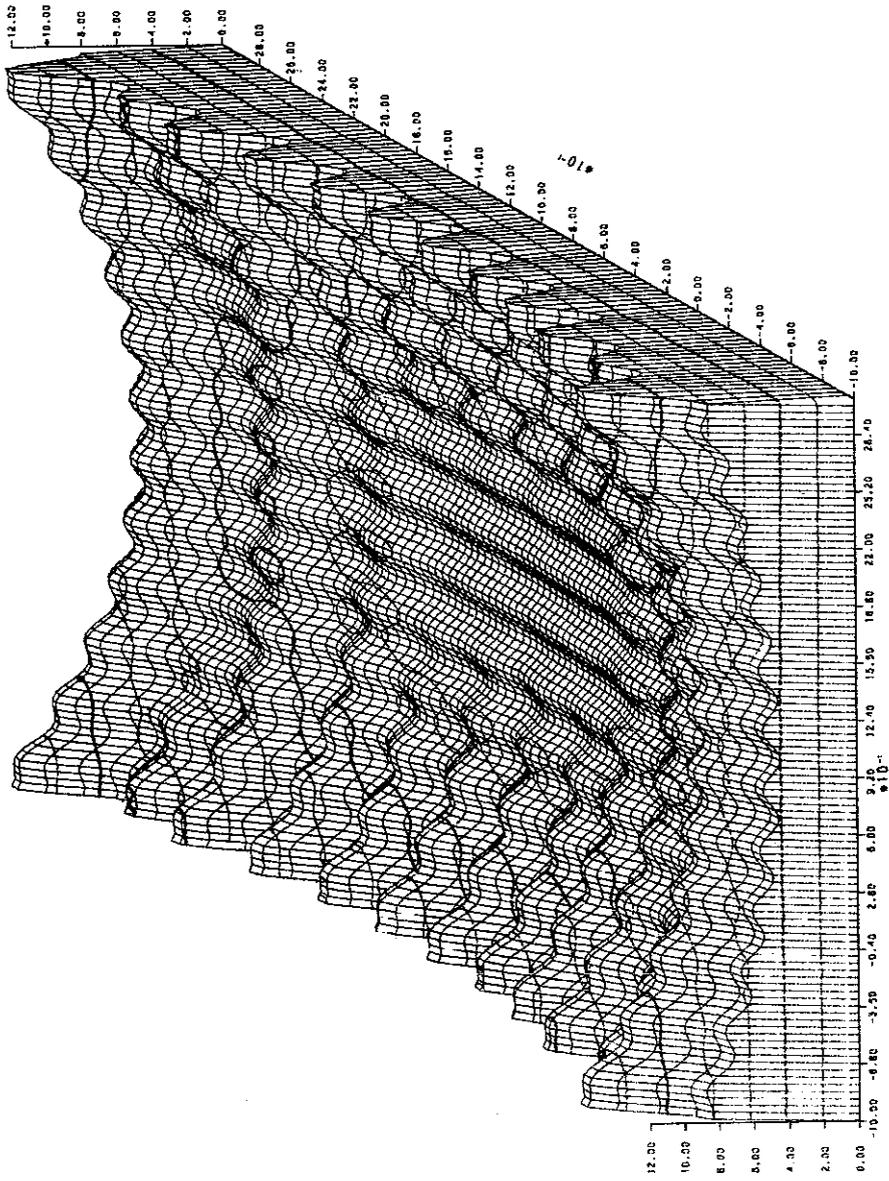


Figura 15.- Ecuación (91), n = 2

$$-1 \leq x_1, x_2 \leq 3$$

TESIS CON
FALLA DE ORIGEN

5.3.- Criterio de Evaluación.

Como se observa en la sección anterior, existen dos problemas en una sola variable (ejemplos 1 y 2), seis ejemplos en dos dimensiones (ejemplos 3-7 y 13), uno de los cuales exhibe 18 mínimos globales, seis ejemplos que exhiben un solo mínimo global localizado en el fondo de un valle orientado a lo largo del eje X_n (ejemplos 7-12) y seis ejemplos, también con un solo mínimo global el cual está completamente rodeado de mínimos locales (ejemplos 13-18)

Independientemente de las características de los ejemplos, existe la necesidad de usar alguna base que haga posible la comparación de los diferentes algoritmos de optimización global, siendo esta la siguiente: Cada uno de los ejemplos fue resuelto para diferentes puntos de arranque (tres para los ejemplos unidimensionales y cuatro para los demás), usando primeramente el algoritmo de tunelización, según se detalla mas adelante. Si cada una de las corridas empleó t_i , $i = 1,2,3,4$, segundos de procesador central en localizar el mínimo global, o en el caso de no encontrarlo el tiempo que empleó en alcanzar el mínimo más bajo posible, se calculó el tiempo promedio por corrida, usando la definición convencional

$$t_{av} = \frac{1}{N_R} \sum_1^{N_R} t_i \quad (92)$$

Es necesario hacer notar que los tiempos de corrida t_i , no incluyen la verificación de la condición de paro dada por la ecuación (52).

Una vez que se ha obtenido el tiempo de corrida promedio para el algoritmo de tunelización, t_{av} , el resto de los algoritmos son corridos durante t_{av} segundos, excepto si logran localizar todos los mínimos globales antes de ese tiempo, en cuyo caso se reportará el tiempo que emplearon en lograrlo. (Dado el caracter aleatorio de los métodos MRS y MMRS, se consideró que una solo corrida era representativo de lo que podía lograr en t_{av} segundos. Esto se comprobó experimentalmente)

NOTA: Si se quisiera hacer una comparación rigurosa entre los diferentes algoritmos, cada uno de ellos debería de correrse hasta veri-

ficar su propia condición de paro. Lo anterior no fue posible implementarlo ya que o bien no se dispone de condición de paro, o la condición de paro no es implementable debido a los requerimientos de memoria y al consumo de tiempo de máquina por lo que se optó por la base de comparación explicada anteriormente. Lo anterior no debe pasarse por alto, ya que experimentalmente se observó que la condición de paro del algoritmo de tunelización consume, aproximadamente, otro tanto de t_{av} , ignorándose por completo el tiempo que consumen los otros algoritmos en satisfacer su condición de paro.

5.4.- Resultados.

En esta sección se presentan los resultados obtenidos para los ejemplos presentados en la sección 5.2 cuando se resuelven usando los algoritmos de los capítulos III y IV, acoplados con el método de gradiente ordinario en su fase de minimización. En esta presentación resulta notable la ausencia de resultados para los métodos de Branin y de Goldstein y Price siendo las razones las que a continuación se exponen.

Método de Goldstein y Price.

Este método se usó para resolver el más sencillo de los 18 ejemplos, función de Goldstein, para el cual el algoritmo encuentra los dos mínimos globales en 2.8 segs, tiempo que resulta 5 veces mayor que lo que consume el algoritmo de tunelización. Al resolver el ejemplo 2, función de Shubert, el algoritmo localizó uno de los tres mínimos globales en 64 segs., lo que significa que es del orden de 15 veces más lento que el algoritmo de tunelización. La tendencia mostrada por estos dos ejemplos indican que el algoritmo de Goldstein y Price requiere altos tiempos de cómputo. Además de lo anterior, el esfuerzo de programación es bastante grande ya que no se sabe de antemano hasta cual de las funciones auxiliares $f_i(x)$, $i = 1, 2, \dots$, se va a requerir. Por lo anterior, el resto de los ejemplos no fueron resueltos mediante este método, concluyéndose que el método de Goldstein y Price no es muy robusto para resolver problemas de minimización global.

Versión de Hardy del método de Branin.

Este método se usó para resolver la función del camello de

seis jorobas (ejemplo 6), función relativamente sencilla. Para este ejemplo, el método de Branin no localiza ninguno de los mínimos globales de la función si se le deja correr durante t_{av} segs. Sin embargo, para comparar la velocidad relativa de este método, se seleccionó un punto de arranque que generara una trayectoria en la cual estuvieran localizados todos los puntos extremos de la función (por tanto los dos mínimos globales), y que consumiera el menor tiempo posible. La trayectoria generada se muestra en la figura 16. Incluso en esta situación tan favorable para el método de Branin, este requirió de 34 segs., lo que significa que es aproximadamente 18 veces más lento que el algoritmo de tunelización. Este método se empleo para resolver la función de Shubert en dos dimensiones (ejemplo 3), resultando que no encuentra más que uno de los mínimos globales en 512 segs. de corrida (esto siempre y cuando el punto de arranque estuviera en el valle donde se localiza el mínimo global). Nuevamente, la tendencia mostrada por estos dos ejemplos, indica que el método de Branin requiere de tiempos de cómputo relativamente altos, por lo que se decidió no usar este método para resolver el resto de los ejemplos.

En lo que sigue, se presentan los resultados obtenidos por los algoritmos de tunelización, MRS y MMRS, usando en sus correspondientes fases de minimización el método de gradiente ordinario para localizar mínimos locales. Con el propósito de tener la mayor cantidad posible de información sobre el comportamiento de los métodos, se reportan los siguientes resultados para cada uno de los 18 ejemplos:

- a) tiempo total empleado en alcanzar los mínimos globales,
- b) evaluaciones totales de la función y sus derivadas,
- c) tiempo parcial consumido en la fase de minimización,
- d) evaluaciones de la función y sus derivadas durante la fase de minimización,
- e) número de minimizaciones requeridas para alcanzar el mínimo global (fases de minimización),
- f) número de veces que se alcanza el mínimo global. Aquí es necesario hacer la siguiente diferenciación:

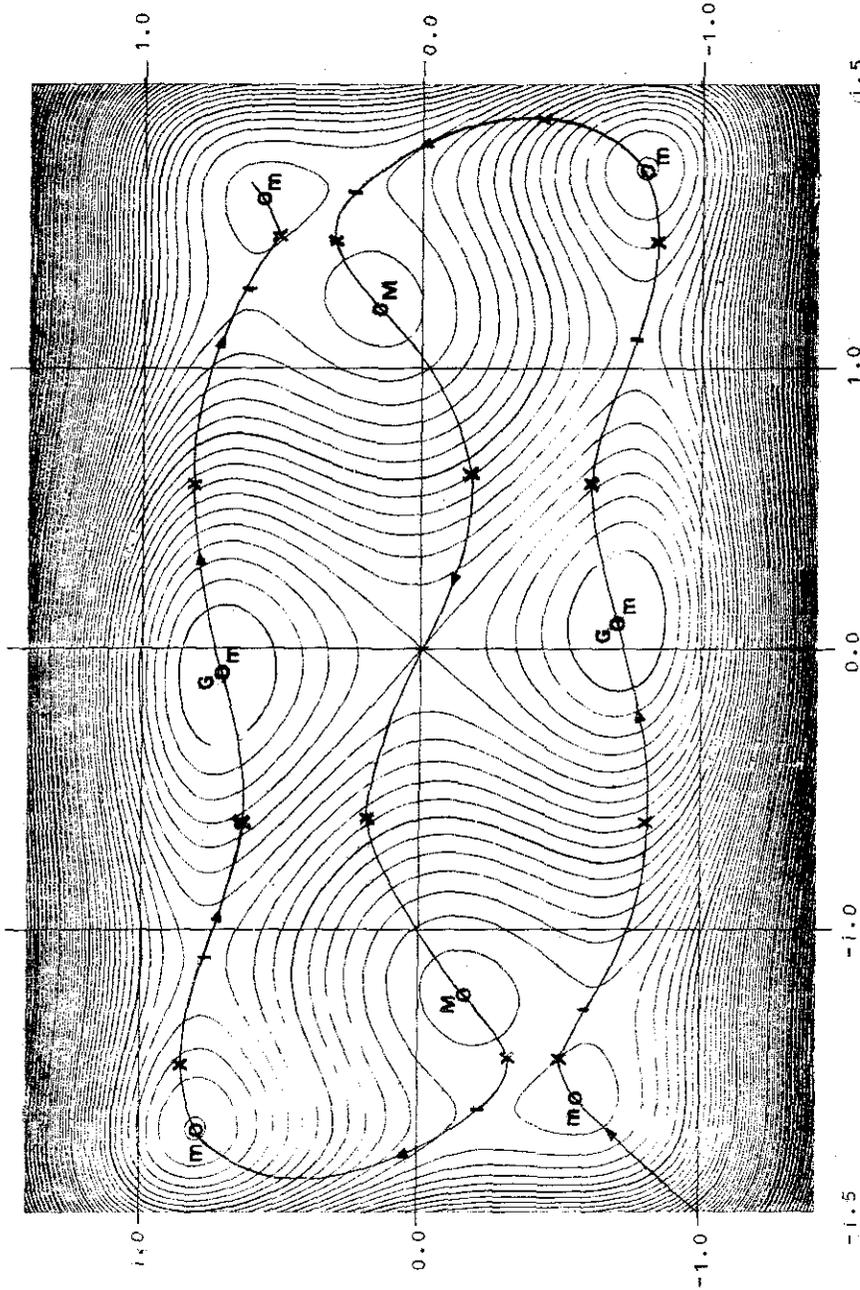


Figura 16.- Trayectoria generada por el método de Branin para la función del camello de seis jorobas.

x: $\det(H) = 0$; G: mínimo; M: mínimo global; I: inflexión

TESIS CON
FALLA DE ORIGEN

- f1) Para funciones que exhiben un solo mínimo global, el valor que se indica para el método de tunelización puede variar entre cero y uno, ya que este valor corresponde al número de veces que en promedio se alcanza el mínimo global, empezando de diferentes puntos de arranque. Para los métodos de MRS y MMRS, el valor reportado es uno o cero (encuentra el mínimo global o no lo encuentra)
- f2) Para funciones que exhiben más de un mínimo global, el valor que se reporta es el promedio de mínimos globales diferentes que se encuentran en las corridas realizadas.

Los resultados obtenidos para los 18 ejemplos analizados se reportan en las Tablas 1-18, indicándose, además, en cada una de las tablas el número de mínimos locales y globales que ese ejemplo en particular exhibe.

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t_{av} ---	0.551	0.097	0.604
Evaluaciones totales de la función ---	758	466	2235
Evaluaciones totales del gradiente ---	129	52	25
tiempo en minimización ---	0.096	0.095	0.057
Evaluaciones de la función en minimización ---	448	466	222
Evaluaciones del gradiente en minimización ---	51	52	25
Número de minimizaciones ---	2.33	3	2
Número de globales ---	2	2	1

Tabla 1.- Ejemplo 1
Mínimos relativos: 3
Mínimos globales: 2

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	4.038	4.108	4.049
Evaluaciones totales de la función ---	1502	2816	3760
Evaluaciones totales del gradiente ---	213	680	7
Tiempo en minimización ---	0.1800	4.002	0.065
Evaluaciones de la función en minimización ---	147	2816	55
Evaluaciones del gradiente en minimización ---	19	680	7
Número de minimizaciones ---	3.33	44	1
Número de globales ---	3	2	1

Tabla 2.- Ejemplo 2
Mínimos relativos: 19
Mínimos globales: 3

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	87.045	88.089	87.066
Evaluaciones totales de la función ---	12160	35479	50462
Evaluaciones totales del gradiente ---	1731	9572	7
Tiempo en minimización ---	2.113	87.845	0.148
Evaluaciones de la función en minimización ---	1047	35479	75
Evaluaciones del gradiente en minimización ---	97	9572	7
Número de minimizaciones ---	17	244	1
Número de globales ---	17	9	1

Tabla 3.- Ejemplo 3
Mínimos relativos: 760
Mínimos globales: 18

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t_{av} ---	8.478	5.197	4.313
Evaluaciones totales de la función ---	2912	-	-
Evaluaciones totales del gradiente ---	390	-	-
Tiempo de minimización ---	1.054	-	-
Evaluaciones de la función en minimización ---	525	-	-
Evaluaciones del gradiente en minimización ---	53	-	-
Número de minimizaciones ---	3	10	2
Número de globales ---	1	0	0

Tabla 4.- Ejemplo 4 NOTA: MRS y MMRS fallan
 Mínimos relativos: 760 después del número de -
 Mínimos globales: 3 minimizaciones indicadas.
 (*)

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t_{av} ---	5.984	2.094	6.018
Evaluaciones totales de la función ---	2180	-	3350
Evaluaciones totales del gradiente ---	274	-	19
Tiempo de minimización ---	1.409	-	0.292
Evaluaciones de la función en minimización ---	710	-	136
Evaluaciones del gradiente en minimización ---	69	-	19
Número de minimizaciones ---	3	4	1
Número de globales ---	1	0	0

Tabla 5.- Ejemplo 5 NOTA: MRS falla después
 Mínimos relativos: 760 del número de minimiza-
 Mínimos globales: 1 ciones indicadas. (*)

(*); Se alcanza el límite de bisecciones sobre α ; ecuación (71)

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	1.984	0.036	2.036
Evaluaciones totales de la función ---	1496	61	6000
Evaluaciones totales del gradiente ---	148	19	10
Tiempo de minimización ---	0.033	0.034	0.016
Evaluaciones de la función en minimización ---	57	61	32
Evaluaciones del gradiente en minimización ---	17	19	10
Número de minimizaciones ---	2	2	1
Número de globales ---	2	2	1

Tabla 6.- Ejemplo 6

Mínimos relativos: 6

Mínimos globales: 2

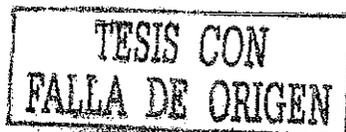
CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	3.283	64.0	1.062
Evaluaciones totales de la función ---	2443	-	1241
Evaluaciones totales del gradiente ---	416	-	287
Tiempo en minimización ---	0.947	-	0.997
Evaluaciones de la función en minimización ---	1116	-	1195
Evaluaciones de gradiente en minimización ---	273	-	287
Número de minimizaciones ---	2.75	1	3
Número de globales ---	1	0	1

Tabla 7 - Ejemplo 7

Mínimos relativos: 25

Mínimos globales: 1

NOTA: MRS falla después del número de minimizaciones indicadas. (Pág. 58)



CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	12.915	3.516	4.024
Evaluaciones totales de la función ---	7325	2861	3443
Evaluaciones totales del gradiente ---	1328	784	726
Tiempo en minimización ---	4.435	3.485	3.231
Evaluaciones de la función en minimización ---	3823	2861	2647
Evaluaciones del gradiente en minimización ---	848	784	726
Número de minimizaciones ---	3.25	3	3
Número de globales ---	1	1	1

Tabla 8.- Ejemplo 8

Mínimos relativos: 125

Mínimos globales: 1

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	20.450	3.391	4.335
Evaluaciones totales de la función ---	4881	-	3076
Evaluaciones totales del gradiente ---	1317	-	457
Tiempo en minimización	1.91	-	2.289
Evaluaciones de la función en minimización ---	1126	-	1369
Evaluaciones del gradiente en minimización ---	376	-	457
Número de minimizaciones ---	4.25	6	2
Número de globales ---	1	0	1

Tabla 9.- Ejemplo 9

Mínimos relativos: 625

Mínimos globales: 1

NOTA: MRS falla después del número de minimizaciones indicadas. (Pág. 58)

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	11.885	8.107	11.925
Evaluaciones totales de la función ---	7540	-	9458
Evaluaciones totales del gradiente ---	1122	-	171
Tiempo en minimización ---	9.320	-	2.112
Evaluaciones de la función en minimización ---	6471	-	1506
Evaluaciones del gradiente en minimización ---	881	-	171
Número de minimizaciones ---	2	1	1
Número de globales ---	1	0	0

Tabla 10.- Ejemplo 10 NOTA: MRS falla después
Mínimos relativos: 10⁵ del número de minimiza-
Mínimos globales: 1 ciones indicadas. (Pág. 58)

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	45.474	38.091	45.535
Evaluaciones totales de la función ---	19366	17229	22193
Evaluaciones totales del gradiente ---	2370	2143	1771
Tiempo en minimización ---	35.644	38.091	30.757
Evaluaciones de la función en minimización ---	16138	17229	14126
Evaluaciones del gradiente en minimización ---	2011	2143	1771
Número de minimizaciones ---	2 5	2	1
Número de globales ---	1	1	0

Tabla 11.- Ejemplo 11
Mínimos relativos: 10⁸
Mínimos globales: 1

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	68.22	192.0	68.26
Evaluaciones totales de la función ---	23982	-	25966
Evaluaciones totales del gradiente ---	3272	-	2913
Tiempo en minimización ---	67.283	-	62.47
Evaluaciones de la función en minimización ---	22191	-	23093
Evaluaciones del gradiente en minimización ---	3135	-	2913
Número de minimizaciones ---	2.5	1	1
Número de globales ---	1	0	0

Tabla 12.- Ejemplo 12
 Mínimos relativos: 10^{10}
 Mínimos globales: 1
 NOTA: MRS falla después del número de minimizaciones indicadas (Pág. 58)

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	4.364	6.308	1.792
Evaluaciones totales de la función ---	2613	6851	1867
Evaluaciones totales del gradiente ---	322	876	250
Tiempo en minimización ---	0.762	6.308	1.406
Evaluaciones de la función en minimización ---	736	6851	1441
Evaluaciones del gradiente en minimización ---	158	876	250
Número de minimizaciones ---	2.5	5	4
Número de globales ---	0.5	0	1

Tabla 13.- Ejemplo 13
 Mínimos relativos: 30^2
 Mínimos globales: 1

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	12.378	13.291	2.975
Evaluaciones totales de la función	6955	10566	2316
Evaluaciones totales del gradiente ---	754	1652	359
Tiempo en minimización	3.927	13.227	2.139
Evaluaciones de la función en minimización ---	3142	10566	2316
Evaluaciones del gradiente en minimización ---	479	1652	359
Número de minimizaciones ---	4.25	9	3
Número de globales ---	0.75	0	1

Tabla 14.- Ejemplo 14
Mínimos relativos: 30³
Mínimos globales: 1

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	8.350	9.851	8.376
Evaluaciones totales --- de la función ---	3861	6659	6234
Evaluaciones totales del gradiente ---	588	740	273
Tiempo en minimización	3.076	9.821	2.294
Evaluaciones de la función en minimización ---	1863	6659	1419
Evaluaciones del gradiente en minimización ---	390	740	273
Número de minimizaciones ---	3.75	2	2
Número de globales ---	0.75	0	0

Tabla 15.- Ejemplo 15
Mínimos relativos: 30⁴
Mínimos globales: 1

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	28.330	51.707	28.362
Evaluaciones totales de la función ---	10715	28347	17339
Evaluaciones totales del gradiente ---	1507	4002	1098
Tiempo en minimización ---	7.249	51.712	11.150
Evaluaciones del función en minimización ---	3565	28347	5746
Evaluaciones del gradiente en minimización ---	797	4002	1098
Número de minimizaciones ---	5.5	7	3
Número de globales ---	0.75	0	0

Tabla 16.- Ejemplo 16
Mínimos relativos: (15)⁵
Mínimos globales: 1

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	33.173	41.065	33.231
Evaluaciones totales de la función ---	12786	19301	18985
Evaluaciones totales del gradiente ---	1777	2784	132
Tiempo en minimización	17.282	41.028	1.263
Evaluaciones de la función en minimización ---	7839	19301	479
Evaluaciones del gradiente en minimización ---	1329	2784	132
Número de minimizaciones ---	3.5	2	2
Número de globales ---	1	0	0

Tabla 17.- Ejemplo 17
Mínimos relativos: (15)⁶
Mínimos globales: 1

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	71.981	92.615	72.027
Evaluaciones totales de la función ---	16063	38483	36195
Evaluaciones totales del gradiente ---	2792	5411	435
Tiempo en minimización ---	15.350	92.546	5.977
Evaluaciones de la función en minimización ---	6142	38483	2132
Evaluaciones del gradiente en minimización ---	1013	5411	435
Número de minimizaciones ---	7.5	8	2
Número de globales ---	0.75	0	0

Tabla 18.- Ejemplo 18

Mínimos relativos: (15)⁷

Mínimos globales: 1

VI.- CONCLUSIONES

A partir de los resultados que se reportan en las Tablas 1-18 del capítulo anterior, se pueden obtener las siguientes conclusiones.

6.1.- Método de Arranque Múltiple Aleatorio (MRS)

Este método resultó ser superior al método de tunelización, si la función que se minimiza satisface los siguientes requerimientos:

- a) La dimensionalidad del problema es baja
- b) La densidad de mínimos relativos alrededor del mínimo global es baja y no uniforme

Estos requerimientos se satisfacen en los ejemplos 1, 6, 8 y 10. Por el contrario, cuando las dos condiciones anteriores no se cumplen, el método MRS no puede encontrar el mínimo global en un tiempo razonable. Esto último se puede observar en la mayoría de los ejemplos analizados.

6.2.- Método de Arranque Múltiple Aleatorio Modificado (MMRS)

Las condiciones bajo las cuales este método es superior al método de tunelización son similares a las mencionadas en 6.1, es decir,

- a) La dimensionalidad del problema es baja
- b) La densidad de mínimos relativos alrededor del mínimo global es baja y no uniforme

Estas dos propiedades se cumplen en los ejemplos 7, 8, 9, 13 y 14. Al igual que para el método MRS, el método MMRS no puede encontrar el mínimo global cuando los requerimientos arriba señalados no se satisfacen, existiendo una característica adicional la cual indica que el método MMRS es altamente ineficiente si la función exhibe más de un mínimo global

6.3.- Método de Tunelización

Los resultados mostrados en las Tablas 1-18 señalan una clara superioridad del método de tunelización sobre los métodos MRS y MMRS, encontrándose que esta superioridad es manifiesta

cuando la función que se minimiza exhibe las siguientes características: (la tercera es restrictiva para el método MMRS)

- a) La dimensionalidad del problema es alta
- b) La densidad de mínimos relativos alrededor del mínimo global es alta y uniforme
- c) Existe más de un mínimo global en la función.

Basándose en lo expuesto en los puntos anteriores, se puede concluir que el método de TUNELIZACION posee las siguientes características que lo convierten en un método atractivo para localizar mínimos globales:

- a) Es superior a algunos de los métodos reportados previamente en la literatura ya que, en general, localiza los mínimos globales más rápidamente que los otros métodos. (Al principio de este párrafo se empleó la palabra algunos, ya que seguramente existen reportados en la literatura más métodos con el mismo propósito y que no fueron analizados). También hay que señalar que cuando el método de tunelización se ve rebasado en eficiencia, de cualquier forma este método sigue localizando el mínimo global de la función.
- b) Si bien es cierto que el tiempo de cómputo aumenta al aumentar la dimensionalidad del problema, la eficiencia del método de tunelización no se ve deteriorada por esta característica de la función objetivo. Esta propiedad del algoritmo de tunelización no la exhiben los otros métodos analizados.
- c) Experimentalmente se observó que la eficiencia del método de tunelización no se ve reducida por la densidad de mínimos relativos que existan en el problema, mientras que los otros métodos sí se ven afectados en forma negativa por esta propiedad.

Para terminar esta tesis, un último comentario. En opinión de muchos investigadores, existen dos formas para generar nuevos algoritmos enfocados a la resolución de un problema específico. El pri-

mero consiste en implementar el algoritmo directamente y analizar a posteriori si tuvo o no éxito, haciendo modificaciones a la idea original, conforme vayan surgiendo problemas no previstos. La segunda forma consiste en demostrar a priori que el algoritmo va a funcionar y hasta después de haberlo hecho se implementa el algoritmo, lo cual servirá para corroborar o rectificar lo demostrado. En la presente investigación se ha optado por la primera forma para generar el algoritmo de tunelización, siendo los resultados que se han obtenido por demás alentadores.

Durante la elaboración de este trabajo, se discutió con algunos investigadores la posibilidad de demostrar teóricamente el porqué funciona tan confiablemente el algoritmo de tunelización, llegando a la conclusión de que la demostración que se requiere justifica, por sí sola, otra tesis doctoral, además de que podría consumir una gran cantidad de tiempo para lograrlo.

VII.- REFERENCIAS

- 1.- Anderson, R.S., and Bloomfield, P., "Properties of the Random Search in Global Optimization". Journal of Optimization Theory and Applications, 16, Nos. 5/6, 1975
- 2.- Avriel, M., "Nonlinear Programming Analysis and Methods", Prentice Hall Inc., Englewood Cliffs, New Jersey, 1976
- 3.- Branin, F.H.Jr., "Widely Convergent Method for Finding Multiple Solutions of Simultaneous Nonlinear Equations", IBM Journal of Research and Development, (504-522), September, 1972
- 4.- Dixon, L., and Szego, G.P., Eds., "Towards Global Optimization", North-Holland Pub. Co., 1975
- 5.- Gaviano, M., "On the Convergence of Random Search Algorithms for Minimization Problems", Dixon, L., and Szego, G.P. Eds., in "Towards Global Optimization", North-Holland Pub. Co., 1975
- 6.- Goldstein, A.A., and Price, J.F., "On Descent from Local Minima". Maths. of Computation, 25, No. 115, 1971
- 7.- Hardy, J., "An Implemented Extension of Branin's Method", in "Towards Global Optimization", Dixon, L., and Szego, G.P., Eds., North-Holland Pub. Co., 1975
- 8.- IBM Scientific Subroutines Package Manual
- 9.- Levy, A.V., and Montalvo, A., "The Tunneling Algorithm for the Global Minimization of Functions", Presentado en Dundee Biennial Conference on Numerical Analysis, Dundee, Escocia, 1977
- 10.- Luenberger, D.G., "Introduction to Linear and Nonlinear Programming", Addison-Wesley Pub. Co., Reading Mass., 1973
- 11.- Miele, A., Huang, H.Y., and Heideman, J.C., "Sequential Gradient-Restoration Algorithm for the Minimization of Constrained Functions-Ordinary and Conjugate Gradient Version", Journal of Optimization Theory and Applications, 4, No. 4, 1969
- 12.- Pope, W.L., Pines, H.S., Silvester, L.F., Green, M.A., and Williams, J.D., "Multiparameter Optimization Studies on Geothermal Energy Cycles" Presentado en 12th Intersociety Energy Conversion Engineering Conference, Washington, D.C., 1977

- 13.- Pope, W.L., Green, M.A., Doyle, P.A., Pines, H.S., and Silvester, L.F., "The Optimization of Alternative Energy Cycles Using Program Geothm", Presentado en Miami International Conference on Alternative Energy Sources, Miami, Fla., 1977
- 14.- Pope, W.L., Pines, H.S., Fulton, R.L., and Doyle, P.A., "Heat Exchanger Design - Why Guess a Design Fouling Factor when it can be Optimizaed?", Aceptado para su publicación por ASME Journal, 1978
- 15.- Shubert, B.O., "A Sequential Method Seeking the Global Maximum of a Function" SIAM Journal of Numerical Analysis, 9, No. 3, 1972
- 16.- Smith, J.M., "Chemical Engineering Kinetics" 2nd Ed, Mc-Graw Hill-Kogakusha, 1970
- 17.- Steiglitz, K., "Computer-Aided Design of Recursive Digital Filters", IEEE Trans, Audio Electracoust., AU-18, (123-129), June, 1970

APENDICE A

RESULTADOS ADICIONALES

Cuando se presentó un reporte preliminar de este trabajo, varios investigadores criticaron fuertemente el uso del método de gradiente ordinario en la fase de minimización de los algoritmos de tunelización. Según estas críticas, se suponía que si se usaba un método de optimización más poderoso que gradiente ordinario, sucedería que la fase de minimización del algoritmo de tunelización consumiría menos tiempo, dando como resultado que los otros algoritmos de minimización global, formados por puras minimizaciones, dispondrían de tiempo adicional para localizar el mínimo global. Como resultado de estas críticas, se implementaron los métodos de Gradiente Conjugado y de Newton Modificado en las fases de minimización de los algoritmos de tunelización, MRS y MMRS.

Lo expuesto hasta aquí y que en principio parece razonable, está basado en las obsevaciones hechas a lo largo de muchos años de investigación, por una serie interminable de autores. Sin embargo, debe recordarse que las conclusiones que se han obtenido son válidas únicamente para funciones unimodales y de ninguna manera se pueden extender a funciones multimodales, que son el tema de esta tesis.

Con el objeto de disipar cualquier duda razonable que pudiera existir a este respecto, se procedió a implementar los métodos de minimización local ya mencionados en la fase de minimización de los algoritmos MRS, MMRS y Tunelización (En el apéndice B se detalla cada uno de los algoritmos de minimización local empleados en esta tesis)

Los resultados obtenidos se muestran en las Tablas A1-A10, para el método de gradiente conjugado, y en las Tablas A11-A20, para el método de Newton. El análisis de los resultados que se presentan en dichas tablas, corroboran las conclusiones expuestas en el capítulo VI de esta tesis, ya que en general se observa que el método de Tunelización es superior a MRS y MMRS. Además de esta conclusión, se pueden hacer una serie de observaciones bastante importantes, y que arrojan luz sobre que tipo de métodos de minimización local se deben usar cuando se está resolviendo un problema de minimización global.

A1.- Sobre el Método de Gradiente Conjugado.

La implementación práctica de este método resultó, sin lugar a dudas, la más difícil, ya que aparecieron una serie de dificultades al intentar la implementación. El principal problema que se encontró fue el siguiente:

Al tratar de correr los ejemplos 3-12, para el único problema que se obtuvieron resultados satisfactorios, fue para el 6 (cabello de seis jorobas), mientras que para los otros, el método no podía encontrar un solo mínimo. Analizando experimentalmente el comportamiento del método (Apéndice C), se llegó a la conclusión de que para iniciar la búsqueda del tamaño de paso tal que

$$\frac{df(x + \alpha s)}{d\alpha} > 0$$

donde x es el punto actual y s la dirección de búsqueda, se debe empezar con una α que depende del problema que se esté resolviendo. Específicamente, para los ejemplos 3, 4 y 5, se encontró que con $\alpha = 0.01$, el algoritmo se estabilizaba, mientras que para los ejemplos 7-12, empezar con $\alpha = 0.1$, era suficiente. Complementando lo anterior, al intentar correr los ejemplos 13-18, no se pudo encontrar el valor adecuado de α que hiciera que el método se estabilizara, razón por la cual no se reportan resultados para estos ejemplos. (Se usaron los siguientes valores de α : 1., 0.5, 0.1 y 0.05)

Por otro lado, ya que el método de gradiente conjugado se basa en la determinación de direcciones $f_{xx}(x)$ -conjugadas (en el caso ideal de una función cuadrática $f_{xx}(x) = \text{Const.}$), cuando este método se usa en problemas de optimización global, al pasar éste de un valle a otro, el comportamiento ideal del método deja de ser válido, no teniendo sentido el obtener direcciones $f_{xx}(x)$ -conjugadas, perdiéndose, además, las características de convergencia cuadrática.

A2.- Sobre el Método de Newton Modificado.

En las tablas A11-A20 tampoco se reportan resultados para los ejemplos 13-18, usando este método, siendo la razón la siguiente: La función que describe estos ejemplos posee, en toda la región de interés, una distribución uniforme de mínimos con valles mu-

cho más suaves que los que exhibe la función de los ejemplos 7-12, Lo anterior provoca que el algoritmo quede atrapado en cada uno de los valles de la función, incrementándose demasidado el tiempo de CPU.

Para ilustrar lo anterior se puede analizar la siguiente situación: Cuando se intentó localizar el mínimo global del ejemplo 14, se efectuaron ocho corridas diferentes, de las cuales solo una alcanzó el mínimo global, mientras que las otras siete, y después de 64 segs. de CPU, el método se encontraba aún "muy lejos" del mínimo global, habiendo pasado la mayor parte de esos 64 segs. en fases de minimización. Hay que mencionar que la corrida que tuvo éxito se inició en un punto cercano al mínimo global. Finalmente, para el ejemplo 18 se realizaron tres corridas, y después de 312 segs. de CPU, el método se hallaba "muy lejos" del mínimo global.

Como conclusión de las experiencias tenidas en la implementación del método de Newton Modificado, podría decirse que la robustez que tiene para localizar mínimos locales, se convierte en una desventaja cuando se intenta localizar mínimos globales.

A3.- Sobre el Comportamiento de los Métodos de Gradiente Ordinario, Gradiente Conjugado y Newton Modificado Acoplados al Algoritmo de Tunelización.

En la Tabla A21 se resume el comportamiento de los diferentes métodos de minimización local acoplados al método de Tunelización. Del análisis de estos resultados, podría concluirse que el método más eficiente resulta ser el de Gradiente Conjugado. Sin embargo, para llegar a los resultados que se reportan, se requirió de un análisis preliminar exhaustivo, y lo que se reporta para los ejemplos 3-12 no se puede extender a otros problemas, como sucedió con los ejemplos 13-18.

Con respecto al método de Newton Modificado, se observa que éste es más eficiente que Gradiente Ordinario para los ejemplo 6-9, no siendo esto válido para el resto de los ejemplos. Además, como se explicó con anterioridad, para otro tipo de ejemplos (3, 4, 5 y 13-18) el método de Newton resulta ineficiente (lento), para localizar el mínimo global de las funciones, aunque eso sí, resultó muy eficiente para localizar mínimos relativos. Desafortunadamente, no es éste último el objetivo que se persigue.

Si todo lo anterior se pone en otras palabras, podría usarse el siguiente razonamiento por demás simplista pero que puede pagar muy buenos dividendos para el problema que en esta tesis se ha planteado: "Si el propósito de la fase de tunelización es el de ignorar todos los mínimos que estén por arriba del último que se encontró, mucho podría ayudar al algoritmo si desde la fase de minimización se tuneliza, ignorando todos los valores de la función que estén por arriba del valor actual (propiedad de descenso local)". Lo anterior es equivalente a pedirle al método de minimización local que, si puede, ignore todos los valles intermedios que le sea posible, los cuales, a fin de cuentas, son irrelevantes en la búsqueda del mínimo global.

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	105.474	58.003	105.524
Evaluaciones totales de la función ---	11862	18778	56556
Evaluaciones totales del gradiente ---	1742	5361	5
Tiempo en minimización ---	4.283	57.04	0.059
Evaluaciones de la función en minimización ---	313	18778	17
Evaluaciones del gradiente en minimización ---	107	5361	5
Número de minimizaciones ---	19.75	599	1
Número de globales ---	17.25	18	1

Tabla A1.- Ejemplo 3

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	4.696	4.786	4.726
Evaluaciones totales de la función ---	1287	1567	2393
Evaluaciones totales del gradiente ---	224	396	21
Tiempo en minimización ---	0.643	4.436	0.291
Evaluaciones de la función en minimización ---	199	1567	103
Evaluaciones del gradiente en minimización ---	53	396	21
Número de minimizaciones ---	6.75	54	2
Número de globales ---	1	0	0

Tabla A2.- Ejemplo 4

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	5.944	6.01	0.185
Evaluaciones totales de la función ---	1699	2208	534
Evaluaciones totales del gradiente ---	254	534	16
Tiempo en minimización ---	0.446	5.648	0.166
Evaluaciones de la función en minimización ---	137	2208	54
Evaluaciones del gradiente en minimización ---	39	534	16
Número de minimizaciones ---	4.75	68	2
Número de globales ---	1	0	1

Tabla A3.- Ejemplo 5

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	0.586	0.102	0.610
Evaluaciones totales de la función ---	486	191	1650
Evaluaciones totales del gradiente ---	61	36	24
Tiempo en minimización ---	0.070	0.097	0.084
Evaluaciones de la función en minimización ---	97	191	127
Evaluaciones del gradiente en minimización ---	23	36	24
Número de minimizaciones ---	2	2	1
Número de globales ---	2	2	1

Tabla A4.- Ejemplo 6

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	1.052	0.058	0.093
Evaluaciones totales de la función ---	544	46	87
Evaluaciones totales del gradiente ---	64	11	10
Tiempo en minimización ---	0.091	0.052	0.054
Evaluaciones de la función en minimización ---	68	46	43
Evaluaciones del gradiente en minimización ---	16	11	10
Número de minimizaciones ---	3.75	2	2
Número de globales ---	1	1	1

Tabla A5.- Ejemplo 7

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	7.218	0.446	0.263
Evaluaciones totales de la función ---	3273	275	180
Evaluaciones totales del gradiente ---	292	63	30
Tiempo en minimización ---	0.187	0.407	0.222
Evaluaciones de la función en minimización ---	119	275	151
Evaluaciones del gradiente en minimización ---	24	63	30
Número de minimizaciones ---	3	6	3
Número de globales ---	1	1	1

Tabla A6.- Ejemplo 8

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} --	11.782	0.352	0.412
Evaluaciones totales de la función ---	4169	197	251
Evaluaciones totales del gradiente ---	527	39	34
Tiempo en minimización ---	0.323	0.347	0.302
Evaluaciones de la función en minimización ---	170	197	172
Evaluaciones del gradiente en minimización ---	35	39	34
Número de minimizaciones ---	3.125	2	2
Número de globales ---	1	1	1

Tabla A7.- Ejemplo 9

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	15.850	5.348	1.223
Evaluaciones totales de la función ---	6030	2977	757
Evaluaciones totales del gradiente ---	584	554	131
Tiempo en minimización ---	1.593	5.342	1.206
Evaluaciones de la función en minimización ---	817	2977	634
Evaluaciones del gradiente en minimización ---	174	554	131
Número de minimizaciones ---	5.25	11	3
Número de globales ---	1	1	1

Tabla A8.- Ejemplo 10

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	24.773	20.897	24.735
Evaluaciones totales de la función ---	7343	7609	12079
Evaluaciones totales del gradiente ---	885	1498	144
Tiempo en minimización ---	6.001	20.881	1.951
Evaluaciones de la función en minimización ---	2115	7609	12079
Evaluaciones del gradiente en minimización ---	444	1498	144
Número de minimizaciones ---	4.75	4	2
Número de globales ---	1	1	0

Tabla A9.- Ejemplo 11

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	118.737	120.668	118.733
Evaluaciones totales de la función ---	27312	34780	48443
Evaluaciones totales del gradiente ---	2857	7414	106
Tiempo en Minimización ---	7.890	120.522	1.814
Evaluaciones de la función en minimización ---	2160	34870	526
Evaluaciones del gradiente en minimización ---	511	7414	106
Número de minimizaciones ---	11.25	26	2
Número de globales ---	1	0	0

Tabla A10.- Ejemplo 12

ESTA TESIS NO SALI
DE LA BIBLIOTECA

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	128.656	128.690	128.763
Evaluaciones totales de la función ---	14071	19113	68710
Evaluaciones totales del gradiente ---	2129	9419	48
Tiempo en minimización ---	3.210	124.132	0.632
Evaluaciones de la función en minimización ---	487	19113	93
Evaluaciones del gradiente en minimización ---	237	9419	48
Evaluaciones del Hessiano en minimización ---	219	8982	45
Número de minimizaciones ---	18	438	3
Número de globales ---	17	16	1

Tabla A11.- Ejemplo 3

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	8.173	8.421	8.140
Evaluaciones totales de la función ---	1995	1263	4165
Evaluaciones totales del gradiente ---	406	598	16
Tiempo en minimización ---	2.157	8.305	0.188
Evaluaciones de la función en minimización ---	345	1263	20
Evaluaciones del gradiente en minimización ---	150	598	16
Evaluaciones del Hessiano en minimización ---	143	576	12
Número de minimizaciones ---	7	22	4
Número de globales ---	1	0	0

Tabla A12.- Ejemplo 4

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	10.466	31.06	10.509
Evaluaciones totales de la función ---	2668	-	5438
Evaluaciones totales del gradiente ---	476	-	13
Tiempo en minimización ---	1.699	-	0.164
Evaluaciones de la función en minimización ---	228	-	16
Evaluaciones del gradiente en minimización ---	127	-	13
Evaluaciones del Hessiano en minimización ---	104	-	11
Número de minimizaciones ---	7	17	2
Número de globales ---	1	0	0

Tabla A13.. Ejemplo 5 NOTA: MRS falla después del número de minimizaciones indicadas. (Pág.58)

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	0.430	0.396	0.464
Evaluaciones totales de la función ---	249	241	1115
Evaluaciones totales del gradiente ---	59	187	22
Tiempo en minimización ---	0.061	0.387	0.058
Evaluaciones de la función en minimización ---	23	241	24
Evaluaciones del gradiente en minimización ---	19	187	22
Evaluaciones del Hessiano en minimización ---	19	174	18
Número de minimizaciones ---	3.5	13	4
Número de globales ---	2	2	1

Tabla A14.- Ejemplo 6

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	0.327	0.268	0.113
Evaluaciones totales de la función ---	97	57	52
Evaluaciones totales del gradiente ---	39	52	14
Tiempo en minimización ---	0.110	0.246	0.113
Evaluaciones de la fun- ción en minimización --	27	57	14
Evaluaciones del gra- diente en minimización ---	22	52	14
Evaluaciones del Hessiano en minimización ---	20	47	11
Número de minimizaciones ---	2.125	5	3
Número de globales ---	1	1	1

Tabla A15.- Ejemplo 7

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av}	5.868	0.284	0.376
Evaluaciones totales de la función ---	2556	39	244
Evaluaciones totales del gradiente ---	245	35	16
Tiempo en minimización ---	0.377	0.257	0.119
Evaluaciones de la fun- ción en minimización ---	146	39	17
Evaluaciones del gra- diente en minimización ---	34	35	16
Evaluaciones del Hessiano en minimización ---	31	30	13
Número de minimizaciones ---	2.75	5	3
Número de globales ---	1	1	1

Tabla A16.- Ejemplo 8

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	5.290	1.336	9.218
Evaluaciones totales de la función ---	1102	369	2689
Evaluaciones totales del gradiente ---	431	96	709
Tiempo en minimización ---	3.480	1.331	6.937
Evaluaciones de la función en minimización ---	438	369	982
Evaluaciones del gradiente en minimización ---	363	96	709
Evaluaciones del Hessiano en minimización ---	361	92	704
Número de minimizaciones ---	2	2	3
Número de globales ---	1	1	0

Tabla A17.- Ejemplo 9

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	46.195	60.053	46.228
Evaluaciones totales de la función ---	16555	-	33920
Evaluaciones totales del gradiente ---	1596	-	66
Tiempo en minimización ---	6.829	-	0.958
Evaluaciones de la función en minimización ---	2512	-	133
Evaluaciones del gradiente en minimización ---	264	-	66
Evaluaciones del Hessiano en minimización ---	252	-	63
Número de minimizaciones ---	11.25	5	3
Número de globales ---	1	0	0

Tabla A18.- Ejemplo 10
 NOTA: MRS falla después del número de minimizaciones indicadas. (Pág. 58)

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	157.83	10.672	10.375
Evaluaciones totales de la función ---	36977	1262	1259
Evaluaciones totales del gradiente ---	3708	460	447
Tiempo en minimización ---	78.94	10.660	10.363
Evaluaciones de la función en minimización ---	16488	1262	1257
Evaluaciones del gradiente en minimización ---	1641	460	447
Evaluaciones del Hessiano en minimización ---	1624	457	455
Número de minimizaciones ---	15	3	2
Número de globales ---	1	1	1

Tabla A19.- Ejemplo 11

CONCEPTO	METODO		
	TUNELIZACION	MRS	MMRS
t _{av} ---	230.381	312.0	230.040
Evaluaciones totales de la función ---	41703	-	92208
Evaluaciones totales del gradiente ---	3170	-	161
Tiempo en minimización ---	208.570	-	10.304
Evaluaciones de la función en minimización ---	36741	-	1324
Evaluaciones del gradiente en minimización ---	2686	-	161
Evaluaciones del Hessiano en minimización ---	2678	-	153
Número de minimizaciones ---	6.75	1	2
Número de globales ---	1	0	0

Tabla A20.- Ejemplo 12

NOTA: MRS falla después del número de minimizaciones indicadas. (Pág.58)

<u>Ejemplo</u>	<u>Método</u>	<u>t_{av}</u>	<u>t_{min}</u>	<u>Número de minimizaciones</u>	<u>Número de globales</u>
3	G. Ord.	87.045	2.113	17.	17.
	G. Con.	105.474	4.283	19.75	17.25
	Newton	128.656	3.210	18.	17.
4	G. Ord.	8.478	1.054	3.	1.
	G. Con.	4.696	0.064	6.75	1
	Newton	8.173	2.157	7.	1.
5	G. Ord.	5.984	1.409	3.	1.
	G. Con.	5.944	0.446	4.75	1.
	Newton	10.466	1.699	7.	1.
6	G. Ord.	1.984	0.033	2.	2.
	G. Con.	0.586	0.07	2.	2.
	Newton	0.43	0.061	3.5	2.
7	G. Ord.	3.283	0.947	2.75	1.
	G. Con.	1.052	0.091	3.75	1.
	Newton	0.327	0.11	2.125	1.
8	G. Ord.	12.915	4.435	3.25	1.
	G. Con.	7.218	0.187	3.	1.
	Newton	5.863	0.377	2.75	1.
9	G. Ord.	20.45	1.91	4.25	1.
	G. Con.	11.782	0.323	3.125	1.
	Newton	5.29	3.48	2.	1.
10	G. Ord.	11.885	9.32	2.	1.
	G. Con.	15.850	1.593	5.25	1.
	Newton	46.195	6.829	11.25	1.
11	G. Ord.	45.474	35.644	2.5	1.
	G. Con.	24.773	6.001	4.75	1.
	Newton	157.83	78.941	15.	1.
12	G. Ord.	68.22	67.383	2.5	1.
	G. Con.	118.737	7.890	11.25	1.
	Newton	230.381	208.57	6.75	1.

Tabla A21. - Comparación de los métodos de minimización local.

APENDICE B

METODOS DE MINIMIZACION LOCAL

En este apéndice se presentan los algoritmos de minimización local en la forma en que fueron implementados en los diferentes programas empleados en esta tesis.

B1.- Gradiente Ordinario.

Sea

$$f = f(x) \quad (B.1)$$

$$a \leq x \leq b$$

la función escalar que se desea minimizar, donde x es un vector de dimensión n y, a y b forman la región de interés donde se debe localizar el mínimo de $f(x)$. Si $f(x)$ se expande en su serie de Taylor alrededor de un punto cualquiera x , se obtiene

$$f(x+\Delta x) = f(x) + f_x(x)\Delta x + \Delta x^T f_{xx}(x)\Delta x/2 + \dots \quad (B.2)$$

Si se toma Δx tal que $\Delta x \rightarrow 0$, $f(x + \Delta x)$ se puede aproximar como

$$f(x + \Delta x) \approx f(x) + f_x(x)\Delta x \quad (B.3)$$

y si el desplazamiento Δx se toma en la dirección de máximo descenso, es decir,

$$\Delta x = -\alpha f_x(x) \quad (B.4)$$

donde α es un escalar mayor que cero, se obtiene

$$f(x + \Delta x) - f(x) = -\alpha f_x^T(x) f_x(x) \quad (B.5)$$

y ya que $f_x^T(x) f_x(x) > 0$, entonces

$$f(x + \Delta x) - f(x) < 0 \quad (B.6)$$

para α suficientemente pequeña, es decir, se obtiene la propiedad de descenso local del algoritmo, tal y como la requiere el método de tunnelización. Obviamente si $f_x(x) = 0$, entonces $\Delta x = 0$, para toda α , y la función $f(x)$ ha alcanzado un punto estacionario.

Los pasos requeridos para implementar este algoritmo son los siguientes:

- 1.- Estimar x , la localización de un punto extremo
- 2.- Calcular $f(x)$ y $f_x(x)$
- 3.- Si $f_x^T(x)f_x(x) \leq \epsilon$, se ha alcanzado un punto extremo de la función.
- 4.- $\alpha = 1$
- 5.- $\Delta x = -\alpha f_x(x)$
- 6.- $\bar{x} = x + \Delta x$
- 7.- Verificar que \bar{x} está en la región de interés, procediendo como se describió en la sección 4.1
- 8.- Si $f(\bar{x}) < f(x)$, se acepta \bar{x} como correcto, se hacen $x = \bar{x}$ y $f(x) = f(\bar{x})$, calculándose $f_x(x)$ en el nuevo punto. Se procede nuevamente desde el paso 3.
- 9.- Si $f(\bar{x}) \geq f(x)$, se bisecta el valor del tamaño de paso α y se repite el procedimiento desde el paso 5. Los pasos 5-9 se repiten hasta un máximo de N_b veces.

Es fácil demostrar que cuando en el paso 7 se modifican a cero algunos de los componentes del desplazamiento Δx , la propiedad de descenso local del método, ecuación (B.6), no se pierde.

B2.- Gradiente Conjugado

La versión implementada de este algoritmo, corresponde a la de Fletcher-Reeves (Computer J., 7, No.149, 1964). Este método genera una secuencia de direcciones de búsqueda s , que son combinación lineal de la dirección de máximo descenso en el último punto calculado, $-f_x(x^k)$, y de las direcciones de búsqueda empleadas en iteraciones anteriores, $s^0, s^1, s^2, \dots, s^{(k-1)}$, escogiéndose un factor de peso tal que las direcciones de búsqueda sean conjugadas

La idea detrás del método es la siguiente. Sea la dirección inicial de búsqueda $s^0 = -f_x(x^0)$. El siguiente punto, definido como en el paso 6 del método de gradiente ordinario, es decir,

$x^1 = x^0 + \alpha s^0$, se escoge tal que

$$\frac{df(x^0 + \alpha x^0)}{d\alpha} = 0 \quad (\text{B.7})$$

lo que implica que se está encontrando el mínimo de $f(x)$ a lo largo de la dirección de búsqueda s^0 .

La siguiente dirección de búsqueda a partir de x^1 , se escoge como

$$s^1 = -f_x(x^1) + \omega_1 s^0 \quad (\text{B.8})$$

donde el escalar ω_1 es tal que si $f(x)$ es cuadrática, entonces

$$(s^0)^T f_{xx} s^1 = 0 \quad (\text{B.9})$$

Si lo anterior es cierto, el valor de ω_1 está dado por

$$\omega_1 = \frac{f_x^T(x^1) f_x(x^1)}{f_x^T(x^0) f_x(x^0)} \quad (\text{B.10})$$

La dirección de búsqueda s^2 se toma como una combinación lineal de $-f_x(x^2)$, s^1 y s^0 , y si se toma en cuenta que $(s^1)^T f_x(x^2) = 0$, se obtiene

$$\omega_2 = \frac{f_x^T(x^2) f_x(x^2)}{f_x^T(x^1) f_x(x^1)} \quad (\text{B.11})$$

y generalizando se llega a

$$\omega_k = \frac{f_x^T(x^k) f_x(x^k)}{f_x^T(x^{k-1}) f_x(x^{k-1})} \quad (\text{B.12})$$

Los pasos involucrados en este algoritmo son los siguientes:

- 1.- Estimar x^0
- 2.- Se toma $s^0 = -f_x(x^0)$ como dirección de búsqueda
- 3.- $k = 0$
- 4.- Determinése α tal que la condición (B.7) se satisfaga dentro

de una cierta tolerancia ϵ . Si el nuevo punto $x^{k+1} = x^k + \alpha s^k$ satisface que $a \leq x^{k+1} \leq b$, continúe con el paso 6.

- 5.- Si x^{k+1} no está en la región de interés, se procede como en el método de gradiente ordinario y se reinicia el proceso desde el paso 2, tomando como x^0 el último valor que se calcule en éstas etapas de gradiente ordinario.
- 6.- Si se han efectuado $(n + 1)$ iteraciones ($k = n$), se reinicia el proceso desde el paso 2, tomando como x^0 el último valor calculado de x .
- 7.- Se evalúa $f_x(x^{k+1})$
- 8.- Si $f_x^T(x^{k+1})f_x(x^{k+1})$, se ha alcanzado un punto estacionario de $f(x)$
- 9.- La siguiente dirección de búsqueda, s^{k+1} , se determina de acuerdo a

$$s^{k+1} = -f_x(x^{k+1}) + \frac{f_x^T(x^{k+1})f_x(x^{k+1})}{f_x^T(x^k)f_x(x^k)} s^k$$

10.- $k = k + 1$

11.- El proceso anterior se repite desde el paso 4.

Para determinar el valor óptimo del tamaño de paso α , tal y como se requiere en el paso 4, se usó el algoritmo de Davis, Swann y Campey para identificar en que rango del parámetro se encuentra el mínimo de $f(x)$; una vez que se ha localizado dicho rango, para determinar la localización exacta del tamaño de paso óptimo, se empleó el algoritmo de Powell (Himmelblau, D.M., "Applied Nonlinear Programming", Mc Graw-Hill Book Co., 1972)

La primera parte del método, con la cual se localiza el rango del parámetro α en el cual está el mínimo, incluye los siguientes pasos (Fig. B1)

- 1.- Evalúe $f(x)$ en el punto inicial. Si $f(x^0 + \Delta x) < f(x^0)$ continuar con el paso 2. Si $f(x^0 + \Delta x) > f(x^0)$, hacer $\Delta x = -\Delta x$ y continuar con el paso 2
- 2.- $x^{k+1} = x^k + \Delta x$

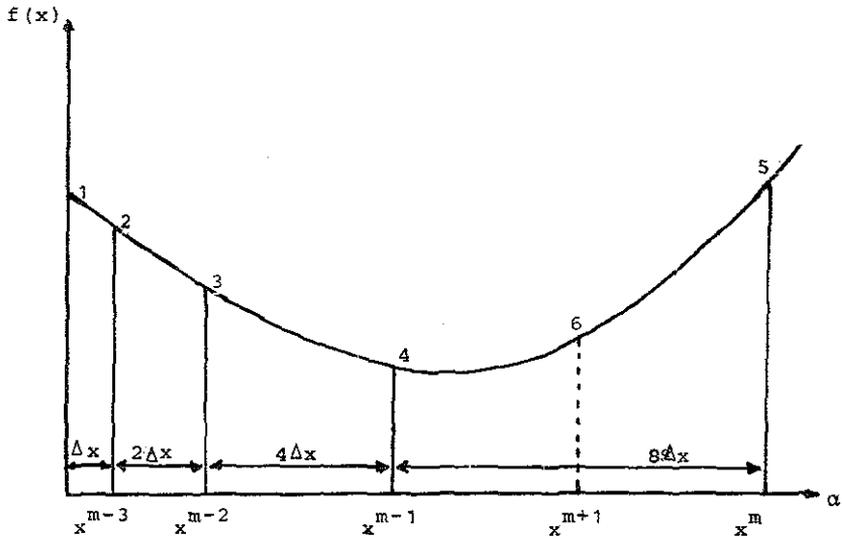


Figura B.1.- Minimización unidireccional

- 3.- Evalúese $f(x^{k+1})$
- 4.- Si $f(x^{k+1}) < f(x^k)$, se duplica Δx y se regresa al paso 2 con $k = k + 1$. Si $f(x^{k+1}) > f(x^k)$, sean $x^m = x^{k+1}$, $x^{m-1} = x^k$, etc. Redúzcase Δx a la mitad y regrese a los pasos 2 y 3, únicamente una vez más.
- 5.- De los cuatro valores de x igualmente espaciados en el conjunto $\{x^{m+1}, x^m, x^{m-1}, x^{m-2}\}$, se elimina ya sea x^m o x^{m-2} , dependiendo de cual está mas lejos de la x que produjo el menor valor de la función dentro del conjunto. Los tres puntos restantes se nombran, ahora, x^a , x^b y x^c , donde $x^a = x^b - \Delta x$ y $x^c = x^b + \Delta x$.
- 6.- Se hace una estimación de x^* , mediante la interpolación cuadrática

$$x^* \approx \bar{x}^* = x^b + \frac{\Delta \bar{x} \{f(x^a) - f(x^c)\}}{2\{f(x^a) - 2f(x^b) + f(x^c)\}}$$

Hasta este punto, ya se ha logrado determinar entre que valores de α está el mínimo de $f(x)$. Lo que sigue es la determinación exacta de x^* , lo cual se realiza usando interpolaciones cuadráticas en forma iterativa.

- 7.- Si $f(\bar{x}^*)$ y $\min\{f(x^a), f(x^b), f(x^c)\}$ difieren en una cantidad no mayor que una tolerancia predeterminada, la búsqueda del óptimo de $f(x)$ ha terminado, con $x^* = \bar{x}^*$. De lo contrario, se elimina del conjunto $\{x^a, x^b, x^c\}$ el punto que produzca el mayor valor de la función, teniendo cuidado de no perder el rango de valores de α donde se encuentra el mínimo.
- 8.- El paso 7 se repite tantas veces como sea necesario.

B3.- Newton Modificado

Los dos métodos anteriores se basan en la reducción del valor de la función $f(x)$ de una iteración a otra, hasta que la condición de paro es satisfecha. Si bien el fundamento del método de Newton es otro, se puede adaptar para que se comporte de manera análoga a los dos métodos anteriores.

El fundamento del método de Newton es el siguiente. Sea

$$g(x) = f_x^T(x) f_x(x) \quad (B.13)$$

una medida del error en la condición de optimalidad de primer orden, es decir, si $g(x) = 0$ se está en un punto extremo de $f(x)$, mientras que si $g(x) \neq 0$, se está fuera de un punto extremo (máximo, mínimo o punto silla). En el método de Newton original, se intenta reducir de una iteración a otra el valor de $g(x)$, lo cual es equivalente a resolver el sistema de ecuaciones no lineales $f_x(x) = 0$. Si $f_x(x)$ se expande alrededor de cualquier punto, en su serie de Taylor, se obtiene

$$f_x(x + \Delta x) = f_x(x) + f_{xx}(x)\Delta x + \dots \quad (B.14)$$

la cual es una aproximación de primer orden de $f_x(x)$. Sea x un punto tal que $f_x(x) \neq 0$ y se pretende que en la siguiente iteración, i.e., en $x + \Delta x$, $f_x(x + \Delta x) = 0$, obteniéndose de la ecuación (B.14)

$$\Delta x = -f_{xx}^{-1}(x) f_x(x) \quad (B.15)$$

Los pasos que se dan en el método de Newton original, se toman multiplicando el lado derecho de la ecuación (B.15) por un escalar α , como

$$\Delta x = -\alpha f_{xx}^{-1}(x) f_x(x) \quad (B.16)$$

debiéndose encontrar el valor de α tal que

$$g(x + \Delta x) < g(x) \quad (B.17)$$

Es fácil observar que la condición anterior no implica de manera alguna, que $f(x + \Delta x) < f(x)$, por lo que el método de Newton no se puede emplear en la forma en que ha sido descrito.

Para lograr que el método de Newton posea la propiedad de descenso local que se requiere, la ecuación (B.16) se modifica de acuerdo a

$$\Delta x = -\alpha p f_{xx}^{-1}(x) f_x(x) \quad (B.18)$$

donde

$$\rho = \begin{cases} 1 & \text{si } \delta f = -f_x^T (f_{xx})^{-1} f_x < 0 \\ -1 & \text{si } \delta f = -f_x^T (f_{xx})^{-1} f_x > 0 \end{cases} \quad (\text{B.19})$$

La razón de introducir el control de dirección ρ , se puede justificar de la siguiente forma. Si en un punto x se tiene que $\delta f > 0$ el método de Newton convergería a un máximo o a un punto silla si no se invierte el sentido de búsqueda, por lo que al hacer esto último, el método convergerá a un punto silla o bien a un mínimo. Si para el punto x antes mencionado resulta que $\delta f < 0$, no es necesario cambiar el sentido de búsqueda, ya que el método convergerá a un punto silla o bien a un mínimo.

Cuando se aplica el método de Newton con esta serie de cambios, el problema consiste ahora en encontrar el valor de α tal que $f(x + \Delta x) < f(x)$. Este proceso se puede empezar con $\alpha = 1$, y permitiendo hasta N_b bisecciones sobre el valor de α .

Además del cambio en la dirección de búsqueda señalado anteriormente, es necesario prever los dos siguientes problemas.

(a) Si en algún paso del algoritmo resulta que $\delta f > 0$, existe la posibilidad de que al aplicar la ecuación (B.18), en el siguiente punto generado por el algoritmo, $f_{xx}(x)$ se haga casi-singular, lo que provocará inestabilidades en el método. Cuando esta situación se presenta, se hace un cambio al método de gradiente ordinario hasta que $f_{xx}(x)$ se convierte, nuevamente, en una matriz bien comportada.

(b) Si para un punto x sucede que la condición de descenso local no se satisface, después de N_b bisecciones, se invierte el signo de ρ , se hace nuevamente $\alpha = 1$ y se reinicia el proceso de búsqueda del tamaño de paso que produzca el descenso deseado sobre la función. Este comportamiento se puede presentar si para α suficientemente pequeña, la variación de segundo orden de la función no puede ser despreciada al lado de la de primer orden.

Incorporando todos los cambios anteriores, el método de Newton Modificado se aplica de acuerdo a los siguientes pasos:

- 1.- Se estima x

- 2.- Se calculan $f(x)$, $f_x(x)$ y $f_{xx}(x)$
- 3.- Si $f_x^T f_x \leq \epsilon_1$, se ha alcanzado un punto extremo de $f(x)$
- 4.- Si $|\det(f_{xx})| \geq \epsilon_2$, se continua con el paso 6
- 5.- Si la condición anterior no se cumple, se avanza con el método de Gradiente Ordinario hasta que la condición anterior se satisface, reiniciándose el proceso desde el paso 2.
- 6.- Calcule f_{xx}^{-1}
- 7.- Si $f_x^T f_{xx}^{-1} f_x > 0$ fijar $\rho = 1$; de lo contrario, $\rho = -1$.
- 8.- Fijar $\alpha = 1$ y $k = 0$
- 9.- $\Delta x = -\alpha \rho f_{xx}^{-1} f_x$
- 10.- $\bar{x} = x + \Delta x$
- 11.- Verificar que \bar{x} se encuentra en la región de interés, de acuerdo a lo descrito en la sección 4.1
- 12.- Si $f(\bar{x}) < f(x)$ continúe con el paso 15
- 13.- Si $f(\bar{x}) \geq f(x)$ se hace $\alpha = \alpha/2$ y se repite desde el paso 9 hasta un máximo de N_b bisecciones sobre α .
- 14.- Si se alcanzó el límite de bisecciones y $k = 0$, se hacen $\alpha = 1$ y $k = 1$, repitiéndose el procedimiento desde el paso 9, invirtiendo el signo de ρ . Si se alcanzó el límite de bisecciones y $k = 1$, esto se toma como una falla del método de Newton y el proceso se dá por terminado.
- 15.- Si se ha satisfecho la condición de descenso local, i.e., $f(\bar{x}) < f(x)$, se toma $x = \bar{x}$ y se repite el proceso desde el paso 2.

NOTA: Para calcular el determinante del Hessiano, $f_{xx}(x)$, así como su inversa, se usó el método de Gauss-Jordan (Isaacson E., and Keller, H.B., "Analysis of Numerical Methods", J. Wiley & Sons, 1966)

APENDICE C.

ANALISIS DEL METODO DE GRADIENTE CONJUGADO

Como se mencionó en el apéndice A, el método de gradiente conjugado presentó una serie de dificultades para su implementación, por lo que se procedió a analizarlas en detalle.

Para analizar el comportamiento del método, se escogió la función del problema 7, observándose lo siguiente. Para un punto de arranque cualquiera, la primera dificultad que apareció fue que el método en general no encontraba mínimo alguno. Después de una serie de pruebas, se encontró que si se reducía el tamaño del paso que usa para buscar el valor óptimo del desplazamiento (Apéndice B), el método tendía a estabilizarse. Finalmente se vió que, para la función escogida, el método funcionaba razonablemente bien con un límite en el tamaño de paso de 0.1

Una vez determinado el valor del máximo tamaño de paso que se puede dar, se procedió a analizar el comportamiento de un ciclo de gradiente conjugado, observándose lo siguiente (Se reportan resultados típicos): Al escoger como punto de arranque $x = -6.0, y = 8.8$, el método de interpolación cuadrática empleado para localizar el valor óptimo de α , genera los resultados que se muestran en la Tabla C1, cuando se usa como dirección de búsqueda

$$s^0 = -f_x(x^0) = (97.968\dots, 2275.200\dots)$$

<u>α</u>	<u>$f(\alpha)$</u>
0.0	438.45
0.1	87089.2
-0.1	78912.1
-0.002476	302.869
-0.004418	553.415
-0.001896	914.252
-0.003310	567.310
-0.002729	861.586
	.
-0.00254931	96.6834 (Optimo)

Tabla C1.- Resultados de la interpolación

Al efectuar la búsqueda señalada, se observa que, debido a la magnitud del gradiente, al usar $\alpha = 0.1$, $f(0.1) > f(0)$, por lo que el método busca en dirección contraria a la prescrita en s^0 , encontrando que $f(-0.1) > f(0)$, procediéndose entonces a empezar las interpolaciones con los tres datos disponibles, obteniéndose como solución, la que se muestra en la Tabla C.1. (En la Fig C1 se muestra una parte de la variación de $f(x)$, en la dirección del gradiente, observándose ahí mismo el caracter multimodal de la función. En la Tabla C.2, se indican las localizaciones de los mínimos unidireccionales, a lo largo de la dirección del gradiente). Este tipo de comportamiento provoca una incertidumbre en cuanto a cual de todos los mínimos locales se va a localizar.

<u>x</u>	<u>y</u>	<u>f(x,y)</u>
-6.4643	-1.9844	119.10
-6.4212	-0.9833	109.818
-6.3781	0.0177	103.190
-6.3350	1.0188	95.340
-6.2919	2.0199	96.683
-6.2488	3.0209	103.033
-6.2057	4.0220	105.521
-6.1645	4.9770	113.286
-6.1215	5.9787	124.324
-6.0783	6.9798	138.957
-6.0352	7.9809	157.276
-5.9921	8.9820	179.334
-5.9490	9.9831	205.144

Tabla C.2.- Localización de los mínimos locales de $f(x)$, a lo largo de la dirección de búsqueda del primer paso de un ciclo.

El resultado anterior provoca un nuevo problema en la determinación de la dirección de búsqueda del segundo paso dentro del mismo ciclo (s^1), ya que esta se determina de acuerdo a



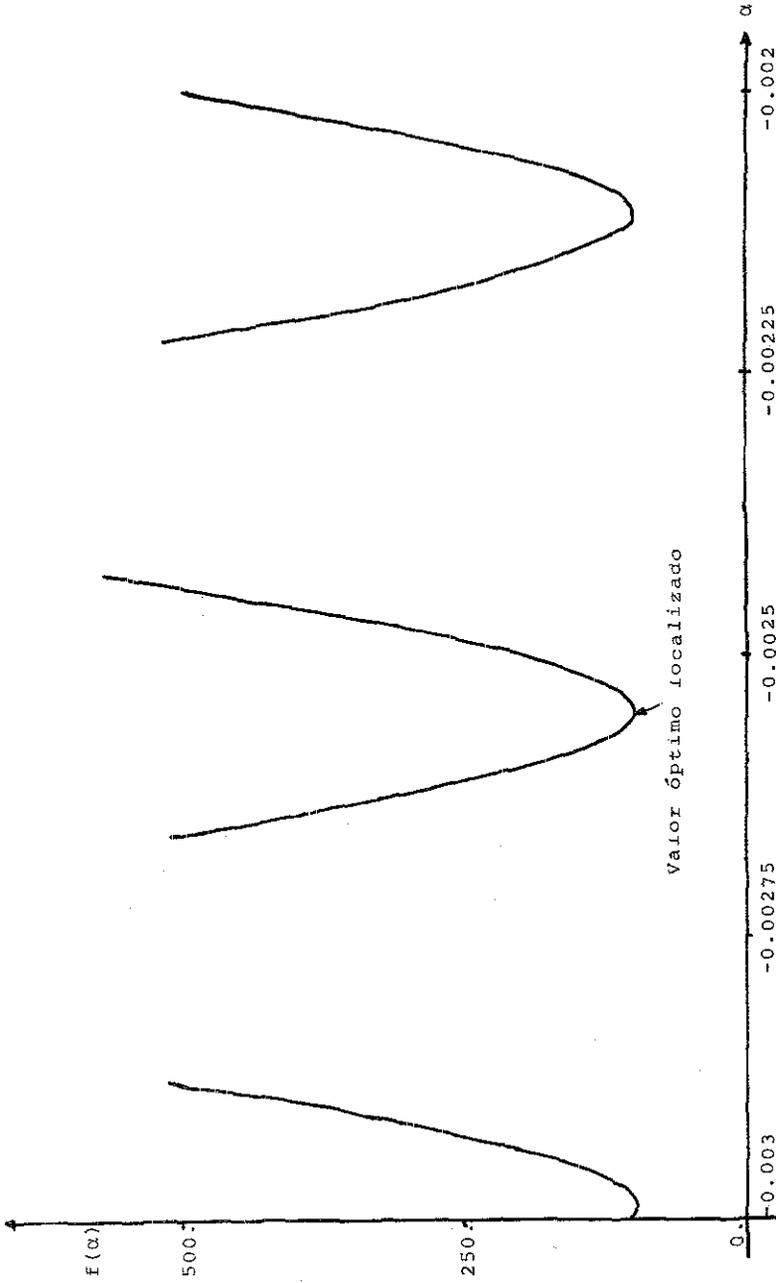


Figura C.1.- Comportamiento de $f(\alpha)$ en la vecindad del óptimo localizado.

TESIS CON
FALLA DE ORIGEN

$$s^1 = -f'_x(\bar{x}) + \omega s^0$$

donde

$$\bar{x} = x^0 + \alpha_{\text{opt}} s^0$$

$$x^0 = (-6.0, 8.8)$$

$$\alpha_{\text{opt}} = -0.0025491$$

$$\omega = \frac{f''_x(\bar{x}) f'_x(\bar{x})}{f''_x(x) f'_x(x)}$$

Debido a que x y \bar{x} no están localizados en el mismo valle de la función, la nueva dirección de búsqueda se torna incierta por no haber mayor relación entre los puntos señalados, desde el punto de vista de gradiente conjugado.

Al efectuarse el segundo paso dentro del mismo ciclo, se obtuvo como dirección de búsqueda $s^1 = (72.222\dots, -1.0025\dots)$. Los resultados que genera el método de interpolación cuadrática se muestran en la Tabla C.3, presentándose nuevamente la característica de varios mínimos locales a lo largo de la dirección de búsqueda, lo que se observa claramente en la Figura C.2.

α	$f(\alpha)$
0.0	96.6833
0.1	5.7874
0.3	2491.91
0.06022	25.3357
0.08466	11.8421
0.09565	18.1709
0.11686	15.6736
.	.
0.100385	5.66565 (Optimo)

Tabla C.3.- Resultados de la interpolación

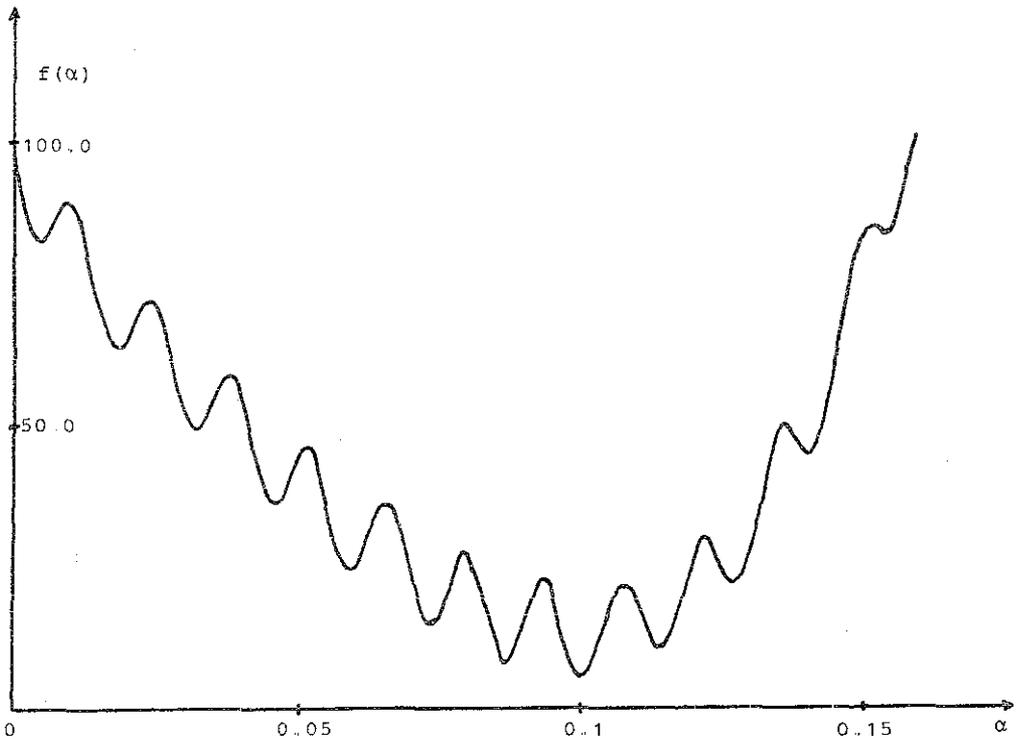


Figura C.2.- Comportamiento de $f(\alpha)$ a lo largo de la dirección de búsqueda s^1

En éste apéndice se ha analizado una de las situaciones típicas que ocurren al usar el método de gradiente conjugado, y sin duda alguna existen algunas otras situaciones que pudieran ser más complejas. La solución inmediata al problema analizado consiste en limitar fuertemente el tamaño máximo del paso que puede ser dado y así intentar garantizar que el método no se salte de un valle a otro; esto con el propósito que todos los pasos de un ciclo tengan relación entre sí.

Sin embargo, la "solución" presentada en el párrafo anterior puede provocar, y seguramente sucederá, una gran lentitud en el avance del método, como lo que le sucede al método de Newton-modificado (ver apéndice B).

Tomando en cuenta lo expuesto en los dos apéndices anteriores y lo analizado en éste, podría concluirse que no es conveniente usar métodos de minimización local que avanzan con la información generada durante etapas anteriores del mismo, como sucede con los métodos de gradiente conjugado, Fletcher-Powell, y en general todos los métodos derivados del de Newton que no requieren de segundas derivadas, (métodos de tipo cuasi-Newton) cuando se está resolviendo un problema de minimización global.

APENDICE D

CONVERGENCIA GLOBAL DEL ALGORITMO DE TUNELIZACION

Para demostrar la convergencia global del algoritmo de tunelización hacia el mínimo global de $f(x)$, se procederá de la siguiente forma:

a) Fase de minimización.- Para esta fase se demostrará que, dado cualquier nominal x^0 se puede emplear un algoritmo $A(x)$ el cual genera una secuencia de iterandos $\{x^k\}$ que converge globalmente a un punto estacionario de $f(x)$.

b) Fase de tunelización.- Para esta fase se demostrará que, dado un nominal x^* , que es solución de la fase de minimización, se puede construir una función auxiliar $T(x, \lambda)$ tal que si para alguna $x^0 \neq x^*$, $T(x^0, \lambda) = 0$, el método de Newton converge monotónicamente a este punto, el cual también es solución de $f(x)$, i.e., $f(x^0) = 0$.

c) Algoritmo de tunelización.- Habiendo demostrado la convergencia de las dos fases que componen el algoritmo de tunelización, se demostrará que, después de un número finito de iteraciones, el mapeo compuesto entre las dos fases ya mencionadas genera un punto x_G^* el cual corresponde a la localización del mínimo global de $f(x)$.

d) Comentarios finales.- En esta sección se eliminan algunas de las restricciones impuestas en las demostraciones que se hacen en los tres incisos anteriores con el propósito de dejar la demostración presentada lo mas general posible.

D.1.- Convergencia de la fase de minimización.

Para ilustrar la convergencia global de esta fase se hará uso del algoritmo de máximo descenso con búsqueda óptima del tamaño de paso en cada iteración y para esto se define:

Definición 1.- El algoritmo $S : E^{2n} \rightarrow E^n$, de búsqueda en una línea, se define como

$$S(x, d) = \{y \mid y = x + \alpha d \text{ para alguna } \alpha \geq 0, \quad (d1)$$
$$f(y) = \min_{0 \leq \alpha < \infty} f(x + \alpha d)\}$$

Además de la definición anterior, se hará uso de la siguiente proposición y corolarios sobre mapeos compuestos.

Proposición 1.- Sea $A : X \rightarrow Y$ y $B : Y \rightarrow Z$ dos mapeos punto-a-punto. Sea A cerrado en x y B cerrado en $A(x)$. Suponga además que si $x_k \rightarrow x$ y $y_k \in A(x_k)$, existe una y tal que para alguna subsecuencia $\{y_{k_i}\}$, $y_{k_i} \rightarrow y$. Entonces, el mapeo compuesto $C = BA$ es cerrado en x . (La demostración se puede encontrar en Luenberger, Op.Cit..)

Corolario A.- Sean $A : X \rightarrow Y$ y $B : Y \rightarrow Z$ dos mapeos punto-a-conjunto. Si A es cerrado en x , B cerrado en $A(x)$ y Y es compacto, entonces el mapeo $C = BA$ es cerrado en X .

Corolario B.- Sean $A : X \rightarrow Y$ y $B : Y \rightarrow Z$ mapeos punto-a-punto. Si A es continuo en x y B es cerrado en $A(x)$, entonces el mapeo compuesto $C = BA$ es cerrado en X .

Teorema.- Sea $f(x)$ una función continua en E^n . Entonces el mapeo definido por (d1) es cerrado en x si $d \neq 0$. (La demostración se puede encontrar en Luenberger, Op.Cit..)

Definición 2.- El método del máximo descenso está definido por el algoritmo

$$x_{k+1} = x_k - \alpha_k f_x(x_k) \quad (S1)$$

donde α_k es un escalar no negativo que minimiza $f(x_k - \alpha f_x(x_k))$.

El algoritmo de máximo descenso $A : E^n \rightarrow E^n$, el cual genera $x_{k+1} \in A(x_k)$, se puede descomponer en la forma $A = SG$. En esta última expresión, $G : E^n \rightarrow E^{2n}$ se define por $G(x) : (x, -f_x(x))$, dando el punto inicial y la dirección de búsqueda. Este mapeo G es seguido por la búsqueda en una línea $S : E^{2n} \rightarrow E^n$, definido al principio de este inciso.

Convergencia Global.- En el teorema anterior se demuestra que el algoritmo S es cerrado si $f_x(x) \neq 0$, y claramente G es continuo. Se sigue que, por el corolario B, $A = SG$ es cerrado.

En este caso el conjunto solución se define como los puntos donde $f_x(x) = 0$ y obviamente $f(x)$ se puede tomar como al función de descenso para A , ya que si $f_x(x) \neq 0$

$$\min_{0 \leq \alpha < \infty} f(x - \alpha f_x(x)) < f(x)$$

Finalmente, y ya que la secuencia $\{x_k\}$ está acotada, ésta tendrá puntos límite y cada uno de ellos será una solución.

Con lo anterior se ha demostrado la convergencia global, hacia un mínimo de $f(x)$, del método de máximo descenso.

D.2.- Convergencia de la Fase de Tunnelización.

Antes de proceder a demostrar la convergencia de esta fase, es conveniente hacer las siguientes aclaraciones:

Suponga que el problema original es el de encontrar el mínimo global de $g(u)$, $u \in U = \{u_i \mid a_i \leq u_i \leq b_i, i=1,2,\dots,n\}$, es decir, U corresponde a un hiperparalelepípedo de n -dimensiones. Suponga además que en la fase de minimización se localizó un mínimo de $g(u)$ en $u^* \in U$ y cuyo valor es $g^* = g(u^*)$.

Sea

$$d = \sup_{u \in U} \{ \|u^* - u\| \}$$

y defínanse las variables escaladas

$$x = (u - u^*)d^{-1}$$

con la consiguiente contracción de U en X , i.e.,

$$X = \{x_i \mid x_i = (u_i - u_i^*)d^{-1}, i=1,2,\dots,n\}$$

es decir que toda $x \in X$ satisface que $\|x\| \leq 1$, y que $g(u(0)) = g^*$ es el valor del mínimo de $g(u)$ en $u^* = u(0)$.

Sea

$$f(x) = g(u(x)) - g^*, \quad x \in X$$

una función que, debido a su definición, exhibe el mismo número de mínimos que $g(u(x))$ y además con la misma localización relativa.

Ya que en la fase de tunnelización se pretende encontrar un punto $u^0 \neq u^*$, $u^0 \in U$, tal que $g(u^0) = g(u^*)$, este problema se puede reformular como sigue:

Si $x^* = 0$ corresponde a la localización de un mínimo de $f(x)$ y además $f(0) = 0$, encontrar un punto $x^0 \neq x^*$, $x^0 \in X$ tal que $f(x^0) = 0$.

Como los dos problemas que se acaban de explicar son equiva-



lentes, en lo que sigue se tratará únicamente el planteado en el párrafo anterior. Además de esto lo que resta de este inciso es válido para problemas en una dimensión, quedando abierto el campo para la demostración del problema n-dimensional.

En este inciso se demostrará el siguiente teorema:

Teorema. - Bajo las hipótesis

- i) $f(x) \in C^2$, $f : \mathbb{R} \rightarrow \mathbb{R}$ y $x \in X = \{x \mid ||x|| \leq 1\}$,
- ii) en $x^* = 0$ existe un mínimo de $f(x)$ que satisface $f(x^*) = f_x(x^*) = 0$ y $f_{xx}(x^*) > 0$ (convexidad local),
- iii) en $\bar{x} \neq x^*$, $\bar{x} \in X$, existe un cero de $f(x)$, i.e., $f(\bar{x}) = 0$.

Entonces, existen $\lambda^* \in \mathbb{N}$, $0 < \epsilon < 1$ y una función $T(x, \lambda)$ para la cual se satisface que:

- a) $T(\bar{x}, \lambda) = 0$, $\bar{x} \in I = \{x \mid \epsilon \leq ||x|| \leq ||\bar{x}||\} \subset X$
 $\lambda \in \mathbb{R}$
- b) Para alguno de los puntos $x^0 = \epsilon$ o bien $x^0 = -\epsilon$, $x^0 \in I$ los iterandos $\{x^k\}$ generados mediante el método de Newton, definido como

$$x^{k+1} = x^k - \frac{T(x^k, \lambda)}{T_x(x^k, \lambda)} \quad k = 0, 1, 2, \dots$$

$$x^k \in I$$

$$\lambda \in \Lambda = \{\lambda \mid \lambda \geq \lambda^* \in \mathbb{N}\}$$

convergen monotónicamente al cero de $T(x, \lambda)$, es decir, al cero de $f(x)$ localizado en \bar{x} .

Demostración.

Sea

$$T(x, \lambda) = \frac{f^2(x)}{x^{2\lambda}} \quad (1)$$

Observe que si en $\bar{x} \neq x^*$ se tiene que $f(\bar{x}) = 0$, esto implica que $T(\bar{x}, \lambda) = 0$, para toda $\lambda \in \mathbb{R}$, y que mientras $f(x) \neq 0$, $x \in X$, $T(x, \lambda) > 0$, también para toda $\lambda \in \mathbb{R}$. Con esta definición de $T(x, \lambda)$ queda demostrada la primera conclusión del teorema.

Para demostrar la segunda conclusión del teorema, se exhibirá que las condiciones del siguiente teorema pueden ser satisfechas usando el valor adecuado del parámetro λ en la ecuación (1).

Teorema de Newton (monótono) (*)

Sea $F : D \subset \mathbb{R} \rightarrow \mathbb{R}$ y suponga que existen $x^\circ, y^\circ \in D$ tales que

$$F(x^\circ) \leq 0 \leq F(y^\circ)$$

Suponga que $F(x)$ es continua, diferenciable y convexa en $|x^\circ, y^\circ|$ y que para cada $x \in |x^\circ, y^\circ|$ $F'_x(x)^{-1}$ existe.

Entonces los iterandos del método de Newton

$$y^{k+1} = y^k - F'_x(y^k)^{-1}F(y^k) \quad k = 0, 1, 2, \dots \quad (2)$$

satisfacen que $\lim_{k \rightarrow \infty} y^k = y^* \in |x^\circ, y^\circ|$ y además $F(y^*) = 0$.

(La demostración se detalla en Ortega y Rheinboldt, Op. Cit.)

1.- De la hipótesis del teorema que se está demostrando, se sabe que existen $\bar{x} \in I \subset X$, para la cual $f(\bar{x}) = T(\bar{x}, \lambda) = 0$ así como $x \in B_\epsilon(x^* = 0)$, $x \neq x^*$ para la cual $f(x) > 0$ y $T(x, \lambda) > 0$, ya que $f(x)$ exhibe un mínimo en x^* . Por otro lado, si $\bar{x} < 0$, entonces el intervalo $|x^\circ, y^\circ|$ queda definido por $|\bar{x}, -\epsilon|$, mientras que si $\bar{x} > 0$ el intervalo estará dado por $|\epsilon, \bar{x}|$. Con esto queda demostrado que existe un intervalo $|x^\circ, y^\circ|$ para el cual se satisface que

$$T(x^\circ, \lambda) \leq 0 \leq T(y^\circ, \lambda) \quad \lambda \in \mathbb{R}$$

y en particular $|x^\circ, y^\circ| \subset I \subset X$. (Note que dependiendo de donde está localizado \bar{x} , uno de los signos de esta doble desigualdad se convertirá en estricta igualdad)

2.- De la definición de $T(x, \lambda)$ se observa que para cada $0 < \epsilon < 1$ tanto $T(x, \lambda)$ como $T'_x(x, \lambda)$ son continuas para cada $x \in I$ y $\lambda \in \mathbb{R}$, ya que

(*). Ortega, J.M., and W.C. Rheinboldt. "Iterative Solution of Non-linear Equations in Several Variables". Academic Press (1970)

$$T_x(x, \lambda) = \frac{2f(x)(xf'_x(x) - \lambda f(x))}{x^{2\lambda+1}} \quad (3)$$

y $x \neq x^* = 0$. Con esto queda demostrada la continuidad y diferenciabilidad de $T(x, \lambda)$, $x \in I$, $\lambda \in R$

3.- Para demostrar la convexidad de $T(x, \lambda)$, respecto a x , se procede de la siguiente forma.

Proposición 1.- Existe $\lambda_1 \in N$ para la cual $T_x(x, \lambda_1) \neq 0$, $x \in I - \partial I$ y $T_x(x, \lambda) = 0$ solo si $x = \bar{x}$.

Demostración.- Note que si $\bar{x} > 0$, entonces $f_x(\bar{x}) \leq 0$ y que si $\bar{x} < 0$, entonces $f_x(\bar{x}) \geq 0$, es decir, $\bar{x}f_x(\bar{x}) \leq 0$. Además, si $x \in B_\varepsilon(x^*=0)$, se satisface que $xf'_x(x) \geq 0$.

Ahora bien, si para cada $x \in I - \partial I$, $\lambda = \lambda(x)$ se escoge como

$$\lambda(x) > \frac{xf'_x(x)}{f(x)} \quad (4)$$

entonces $T_x(x, \lambda(x)) \neq 0$. En particular $T_x(x, \lambda(x)) > 0$ si $x < 0$ y $T_x(x, \lambda(x)) < 0$ si $x > 0$.

Tomando en cuenta la nota del primer párrafo, así como el teorema de Bolzano-Weierstrass, se concluye que el problema

$$w = \max \left\{ \frac{xf'_x(x)}{f(x)} : x \in I \right\} > 0 \quad (5)$$

tiene solución para cada ε , $0 < \varepsilon < ||\bar{x}||$.

Finalmente, si $\lambda > \lambda_1 = w$, se demuestra que

$$T_x(x, \lambda) \neq 0, \quad x \in I, \quad x \neq x^*$$

$$y \quad T_x(\bar{x}, \lambda) = 0 \quad \bar{x} \in I$$

Proposición 2.- Existe $\lambda_2 \in N$ para la cual $T(x, \lambda_2)$ es convexa para todo x , $x \in I$.

Demostración.- Note que en una vecindad de \bar{x} $T(x, \lambda)$ es convexa ya que de la definición de $T(x, \lambda)$, esta función exhibe un mínimo en \bar{x} . (Lo anterior no implica necesariamente que $T_{xx}(x, \lambda) > 0$, ya que

por ejemplo, puede suceder que $T(\bar{x}, \lambda) = T_x(\bar{x}, \lambda) = T_{xx}(\bar{x}, \lambda) = T_{xxx}(\bar{x}, \lambda) = 0$, y $T_{xxxx}(\bar{x}, \lambda) > 0$, i.e., $f(\bar{x}) = f_x(\bar{x}) = 0$, y $f_{xx}(\bar{x}) > 0$

De la definición de $T(x, \lambda)$, se obtiene facilmente que

$$T_{xx}(x, \lambda) = \frac{H(x, \lambda)}{x^{2\lambda+2}}$$

donde

$$H(x, \lambda) = 4\lambda^2 f^2(x) + \lambda(2f^2(x) - 8xf_x(x)) + 2x^2(f(x)f_{xx}(x) + f_x^2(x)) \quad (6)$$

Paso 1.- Si x se fija y $x \in \Gamma$, la función $H(x, \lambda)$, es casi-convexa en λ , $\lambda \in R_+$.

Demostración.-

$$\frac{\partial H(x, \lambda)}{\partial \lambda} = 8\lambda f^2(x) + (2f^2(x) - 8xf_x(x)) \quad (7)$$

$$\frac{\partial^2 H(x, \lambda)}{\partial \lambda^2} = \begin{cases} 8f(x) > 0 & \varepsilon \leq ||x|| < ||\bar{x}|| \\ & \lambda \in R_+ \\ 0 & x = \bar{x} \\ & \lambda \in R \end{cases} \quad (8)$$

Además, el valor de λ que minimiza a $H(x, \lambda)$ está dado por

$$\underline{\lambda}(x) = \frac{xf_x(x)}{f^2(x)} - \frac{1}{4} \quad (9)$$

Paso 2.- Existe un escalar λ_2 tal que

$$H(x, \lambda) > 0, \quad \lambda > \lambda_2$$

para toda x , $x \in I$.

Demostración.- Caso a. Si $H(x, \underline{\lambda}(x)) > 0$ para $x \in I$, es claro que se puede tomar cualquier valor de λ . En particular, se tomará $\lambda_2 > \underline{\lambda}_m$, donde

$$\underline{\lambda}_m = \max \left\{ \frac{xf_x(x)}{f^2(x)} - \frac{1}{4} : x \in I \right\} > 0 \quad (10)$$

problema análogo al de la proposición 1 y el cual también tiene solución.

Caso b. - Si $H(x, \lambda(x)) < 0$, para algún valor de x , $x \in I$, el valor de λ_2 se puede tomar como cualquier escalar mayor que el mayor de los escalares que satisface $H(x, \lambda) = 0$. Este escalar está acotado inferiormente por

$$\lambda_2 > \frac{\lambda_m}{m} + \sqrt{(\frac{\lambda_m}{m})^2 - \gamma/2} \quad (11)$$

donde

$$\gamma = \inf\{(x/f(x))^2 (f'_x(x)^2 + f_{xx}(x)f(x)); x \in I\} < 0 \quad (12)$$

Note que el valor de γ debe ser negativo ya que, por el teorema del valor medio, $f(x)$ exhibe al menos un máximo entre $x = x^*$ y $x = \bar{x}$, ocurriendo en ese lugar que $f_{xx}(x) < 0$. Lo anterior descarta la posibilidad de no solución de este problema ya que en \bar{x}

$$\lim_{\substack{x \rightarrow \bar{x} \\ x \in I}} \{(x/f(x))^2 (f'_x(x)^2 + f(x)f_{xx}(x))\} \geq 0 \quad (13)$$

Usando la mayor λ_2 de los casos (a) y (b), queda demostrado este segundo paso.

Corolario. - Si $\lambda^* = \max\{\lambda_1, \lambda_2\}$, entonces

$$i) \quad T_x(x, \lambda) \begin{cases} \neq 0, & x \in I, \quad x \neq \bar{x}, \quad \lambda \in \Lambda \\ = 0, & x = \bar{x}, \quad \lambda \in R \end{cases} \quad (14)$$

ii) $T(x, \lambda)$ es convexa con respecto a x , para toda $(x, \lambda) \in I \times \Lambda$.

Este corolario se sigue fácilmente debido a la monotonicidad de $H(x, \lambda)$ y de $T_x(x, \lambda)$, con respecto a λ , para $\lambda > \lambda^*$.

Con esto último se ha demostrado que $T(x, \lambda)$ es convexa en $(x, \lambda) \in I \times \Lambda$, con lo cual se satisface la última condición del teorema de Newton (monótono) y por ende, la segunda conclusión del teorema que se está demostrando en este inciso también es válida.

Del teorema que se acaba de demostrar se puede obtener inmediatamente el siguiente resultado:

Lema. Bajo las hipótesis

- i) $f(x) \in C^2$, $f : R \rightarrow R$, $x \in X = \{x \mid \|x\| \leq 1\}$,
- ii) en $x^* = 0$ existe un mínimo de $f(x)$ que satisface $f(x^*) = f_x(x^*) = 0$ y $f_{xx}(x) > 0$,
- iii) Existen $\lambda^* \in N$, $0 < \epsilon < 1$, y una función $T(x, \lambda)$ que satisfacen el teorema de este inciso en cuanto a convexidad y monotonicidad con respecto a x ,
- iv) $T(x, \lambda)$ exhibe únicamente dos mínimos locales en $\|x\| = 1$.
Entonces, el mínimo global de $f(x)$ se encuentra localizado en $x^* = 0$.

Demostración. - Este teorema es resultado inmediato del teorema que se acaba de demostrar y del teorema de Luenberger para mínimos de funciones monótonas y convexas. (Op.Cit.)

D.3.- Convergencia del algoritmo compuesto.

Considérense los mapeos definidos en los incisos D.1 y D.2 como sigue:

- a) MP : $R \rightarrow R$, el cual, dado un nominal x_k^o , genera un mínimo local de $f(x)$ en x_k^* y cuyo valor es $f(x_k^*)$.
- b) TP : $R - B_{0+}(x_k^*) \rightarrow R - B_{0+}(x_k^*)$, el cual, dado un nominal x_k^* encuentra un cero (mínimo) de $T(x, \lambda)$, de acuerdo al teorema del inciso D.2, cero localizado en x_{k+1}^o . (Nota: De la definición de $T(x, \lambda)$, $x_{k+1}^o \neq B_{0+}(x_k^*)$, i.e., $x_{k+1}^o \neq x_k^*$)
- c) El mapeo compuesto
TA : $R \rightarrow R$, definido como

$$TA(x) = TP \{ MP(x) \}$$

y para el cual

$$x_{k+1}^o = TA(x_k^o)$$

Teorema.- Bajo las hipótesis

- i) $f(x) \in C^2$, $f : R \rightarrow R$, $x \in X = \{x \mid a \leq x \leq b\}$
- ii) $f(x)$, $x \in X$, exhibe un número finito de mínimos, todos a diferente nivel, p.ej., N mínimos locales y solo uno de ellos global.

Entonces, el algoritmo de tunelización converge monotónicamente, en N iteraciones o menos, al mínimo global de $f(x)$.

Demostración.- Suponga que se da un nominal x_0^o , el cual cuando es usado como punto de arranque de la fase de minimización, genera un punto x_1^* que corresponde a un mínimo de $f(x)$, i.e.,

$$x_1^* = MP(x_0^o), \quad x_0^o, x_1^* \in X$$

y además $f_x(x_1^*) = 0$, $f_{xx}(x_1^*) > 0$ (Teorema inciso D.2). Por construcción de esta fase,

$$f(x_1^*) \leq f(x_0^o)$$

Con los valores de x_1^* y $f(x_1^*)$, y si el mínimo localizado no corresponde al global de $f(x)$, la fase de tunelización generará un nuevo punto x_1^o , $x_1^o \in X$, para el cual se satisface que:

$$T(x_1^o, \lambda) = 0 \iff f(x_1^o) = f(x_1^*), \quad x_1^o \neq x_1^*$$

en cuyo caso se tiene la garantía, por la hipótesis (ii), de que al menos existe otro mínimo de $f(x)$ a un nivel inferior a $f(x_1^*)$. Al ocurrir esto último, con la localización de x_1^o se puede reiniciar otra fase de minimización, generándose entonces una secuencia de mínimos locales $f(x_i^*)$, y de sus respectivos ceros $f(x_i^o)$, los cuales satisfacen

$$f(x_0^o) \geq f(x_1^*) = f(x_1^o) > f(x_2^*) = f(x_2^o) > f(x_3^*) \dots$$

o bien, eliminando los ceros,

$$f(x_1^*) > f(x_2^*) > f(x_3^*)$$

Esta última propiedad garantiza el descenso monotónico de $f(x)$ hacia el mínimo global, conforme el algoritmo va generando nue-

vos puntos.

Al contrario del caso explicado en el párrafo anterior, suponga ahora que, después de K fases de minimización, sucede que en la fase de tunelización no se encuentran ceros de $T(x, \lambda)$. Si se hace uso del lema dado en el inciso anterior, se concluye inmediatamente que el mínimo de $f(x)$ encontrado en x_K^* , corresponde al mínimo global de $f(x)$, $x \in X$.

Finalmente, supóngase que el primer mínimo que se encontró, en x_1^* con un valor $f(x_1^*)$, es el mayor de los N mínimos que exhibe la función $f(x)$. Resulta por demás obvio que si este fuera el caso, cuando mucho se requerirán N aplicaciones del algoritmo de tunelización para poder alcanzar el mínimo de $f(x)$, i.e., $K < N$.

D.4.- Comentarios finales.

Por simplicidad, en las demostraciones presentadas en los tres incisos anteriores se hicieron una serie de suposiciones sobre las características que $f(x)$ debería de exhibir en los mínimos, así sobre la clase de funciones a la cual debería de pertenecer. En lo que a continuación se discute, se removerán las suposiciones hechas.

D.4.1.- Teorema en el inciso D.2.

$$f(x) \in C^2, \quad f_x(x^*) = 0 \quad \text{y} \quad f_{xx}(x^*) > 0.$$

De la teoría clásica de minimización de funciones, se dice que una función exhibe un mínimo local en x^* si

$$\begin{aligned} f^{(k)}(x^*) &= 0, \quad k = 1, 2, \dots, n \quad (\text{un impar}) \quad (*) \\ \text{y} \quad f^{(n+1)}(x^*) &> 0 \end{aligned}$$

En particular, esto se verifica para el caso de $n = 1$, caso que corresponde a las hipótesis (i) e (ii) del teorema. Resulta pues obvio que las hipótesis hechas al respecto, se pueden modificar de la siguiente manera, sin que los resultados obtenidos se alteren.

(*) $f^{(i)}(x)$ significa $\frac{d^i f(x)}{dx^i}$

- i) $f(x) \in C^{2m}$, $f : \mathbb{R} \rightarrow \mathbb{R}$, $x \in X = \{x \mid ||x|| \leq 1\}$
 m un entero
- ii) sea $x^* = 0$ un minimizador de $f(x)$ tal que $f(x^*) = 0$ y
 $f^{(i)}(x^*) = 0$, $i = 1, 2, \dots, k < 2m$, k un impar y
 $f^{(k+1)}(x^*) > 0$.

D.4.2.- Teorema en el inciso D.3. Hipótesis (i) e (ii).-

La hipótesis (i) se puede cambiar de acuerdo a lo mencionado en D.4.1.

- ii) $f(x)$, $x \in X$, exhibe un número finito de mínimos locales, todos a diferente nivel, p.ej., N mínimos locales.

Usando la hipótesis (i) modificada, se tiene que en la vecindad del mínimo de $f(x)$, en x^* , $f(x)$ se puede escribir como

$$f(x) = \frac{f^{(2i)}(x^*)}{(2i)!} (x - x^*)^{2i} + \frac{f^{(2i+1)}(x^*)}{(2i+1)!} (x - x^*)^{(2i+1)}$$

$f^{(2i)}(x^*) > 0$, para alguna $i < m$. De esta última expresión se deduce que

$$0 < \lim_{x \rightarrow x^*} \left| \frac{f(x) - f(x^*)}{(x - x^*)^{2i}} \right| < \infty$$

Con lo anterior, sean x_1^* y x_2^* localizaciones de mínimos de $f(x)$ tales que $f(x_1^*) = f(x_2^*) = f^*$ (los subíndices indican el orden en que los mínimos fueron encontrados por el algoritmo de tunelización). Del teorema en D.3, se sigue que al usar como función de tunelización la que se definió en ese teorema, es decir,

$$T(x, \lambda) = \frac{(f(x) - f(x_2^*))^2}{(x - x_2^*)^{2\lambda}}$$

existe la posibilidad de localizar a x_1^* como cero de $T(x, \lambda)$, lo que resulta improductivo en la búsqueda del mínimo global de $f(x)$.

Para evitar el problema mencionado en el párrafo anterior, suponga que se han encontrado p mínimos en x_i^* , $i = 1, 2, \dots, p$ tales

que $f(x_i^*) = f^*$, $i = 1, 2, \dots, p$. Sea η_i , $i = 1, 2, \dots, (p-1)$, η_i un entero, un número tal que

$$\begin{aligned} f^{(j)}(x_i^*) &= 0, \quad j = 1, 2, \dots, 2\eta_i - 1 \\ f^{(2\eta_i)}(x_i^*) &> 0 \\ i &= 1, 2, \dots, (p-1) \end{aligned}$$

donde para cada η_i se satisface que

$$0 < \lim_{x \rightarrow x_i^*} \left| \frac{f(x) - f(x_i^*)}{(x - x_i^*)^{2\eta_i}} \right| < \infty$$

Ya que a partir del último mínimo encontrado, localizado en x_p^* , se desea encontrar un nuevo punto $x^\circ \neq x_i^*$, $i = 1, 2, \dots, p$ tal que $f(x^\circ) = f(x_i^*) = f^*$, la función de tunelización que se debe usar queda definida como

$$T(x, \lambda) = \frac{(f(x) - f^*)^2}{(x - x_p^*)^{2\lambda} \prod_{i=1}^{p-1} (x - x_i^*)^{4\eta_i}}$$

$$x \in X - B_{0+}(x_p^*)$$

función para la cual se satisface que

$$0 < T(x_i^*, \lambda) < \infty \quad i = 1, 2, \dots, (p-1)$$

Incorporando esta modificación de la función de tunelización, la hipótesis (ii) del teorema en D.3 se puede reformular como

- ii) $f(x)$, $x \in X$, exhibe un número finito de mínimos, p.ej., N mínimos locales.

Bajo esta nueva hipótesis, la conclusión del teorema cambia a:

Entonces, el algoritmo de tunelización converge monotónicamente, en N iteraciones o menos, al mínimo global de $f(x)$, aún en el caso de que la función $f(x)$ exhiba varios mínimos locales al mis-

mo nivel y/o varios mínimos globales, en cuyo caso una vez que el algoritmo encuentra el primero de ellos, permanecerá en el nivel f^* , hasta localizarlos todos.

Finalmente, debe añadirse que el teorema del inciso D.3 se puede extender fácilmente al caso de n dimensiones, aunque hay que hacer notar que para esta situación el teorema es válido para una clase muy restringida de funciones, $f(x)$. Actualmente, se están extendiendo los resultados del teorema que aquí se ha presentado, con el objeto de que éste sea lo mas general posible.