

UN ALGORITMO CUASI-NEWTON CON APROXIMACIONES CONSISTENTES PARA PROGRAMACION SEMI-INFINITA

C. Nicolas Baracatt

Universidad Austral de Chile, Valdivia-Chile

y

J. Heskovits N.

Universidad Federal de Rio de Janeiro, Rio de Janeiro-Brasil

Abstract

En este artículo desarrollamos un algoritmo cuasi-Newton de puntos interiores para la resolución de problemas semi-infinitos no lineales. Usamos una estrategia de discretización basada en el concepto de aproximaciones consistentes, la cual genera una sucesión de problemas aproximados que convergen Epigráficamente al problema original. Además se construye una función de optimalidad que resulta natural para métodos de Newton y se demuestra que todo punto de acumulación de la sucesión de puntos estacionarios correspondientes a la sucesión de problemas aproximados es un punto estacionario del problema original.

El uso de esta técnica de aproximaciones consistentes en conjunción con una estrategia de discretización diagonalizante y de filtrado de restricciones es ilustrada con ejemplos numéricos.

1. Introducción.-

Debido a su extraordinaria eficiencia, el método de Newton ha fascinado durante muchas décadas a investigadores, encontrándose abundante literatura sobre este método numérico en aplicaciones a la Programación Matemática (Ref. [3,6,8,11,15,20,24]). Sin embargo la extensión del método de Newton a problemas de Programación Semi-Infinita no Lineal (es decir, problemas de infinitas restricciones) parece estar confinada al uso del método de Newton aplicado bajo la hipótesis que se cumple el Teorema de la Función Implícita a las restricciones activas en un punto viable, lo cual permite tratar el problema Semi Infinito como un problema de Programación de finitas restricciones (Ref.[11, 15]).

Por ejemplo un problema de la forma :

$$\text{Min}\{f(x)/\phi(x, u) \leq 0, \forall u \in [a, b], a < b\} \quad (1.1)$$

se puede convertir en un problema de programación matemática estándar de la forma :

$$\text{Min}\{f(x)/\phi(\bar{x}, u_j(x)) \leq 0, \forall j = 1, \dots, s; x \in V(\bar{x}, \delta); \delta > 0\} \quad (1.2)$$

Si $\bar{x} \in \mathfrak{R}^n$ es solución óptima de (1.2) se tienen exactamente "s" maximizadores locales y $\phi_{uu}(x, u_j(x)) < 0; \forall j = 1, \dots, s$. Sin embargo muchas de estas extensiones son algoritmos conceptuales debido a que los maximizadores locales $u_j(x)$ no pueden ser calculados exactamente.

En este trabajo presentamos un algoritmo modificado de Newton (Quasi-Newton) de puntos interiores que en contrapartida a los algoritmos que se han desarrollado hasta la fecha usa una técnica de discretización secuencial diagonalizante con filtrado de restricciones y aproximaciones consistentes (Ref. [22c,23a]); Esta teoría consiste básicamente en aproximar un problema de Optimización Semi- infinita por una sucesión de problemas de finitas restricciones de tal manera que la sucesión de problemas converga Epigráficamente al problema original. De manera simultánea cada problema aproximado de la sucesión tiene asociada una función de optimalidad cuyos ceros expresan la satisfacción de las condiciones de optimalidad de dicho problema, de tal forma que la sucesión de funciones de optimalidad

converja Hipográficamente a la función de optimalidad asociada al problema original, es decir, los puntos estacionarios de cada problema de la sucesión son aproximados a los puntos estacionarios del problema original.

En la sección 2.0 presentamos los fundamentos preliminares de la teoría de aproximaciones consistentes, así como un algoritmo modelo de aproximaciones consistentes ; En la sección 3.0 se desarrolla una implementación del algoritmo modelo y se demuestran sus propiedades; En la sección 4.0 se realizan algunos experimentos numéricos en base a problemas conocidos de la literatura; En la sección 5.0 se entregan las conclusiones.

2. Fundamentos preliminares

Estamos interesados en el siguiente problema de Optimización Semi Infinita (*OSI - P*) :

$$\text{Min}\{f(x)/x \in X(U)\}$$

donde

$$X(U) = \{x \in X_0 / g_i(x) \leq 0, \phi_j(x, u) \leq 0; i \in I, j \in J, \forall u \in U^j\}$$

$f, g_i : \mathbf{X}_o \rightarrow \mathbf{R}; \phi_j : \mathbf{X}_o \times U^j \rightarrow \mathbf{R}$ funciones ; $U^j \subseteq \mathbf{R}^{m_j}$ conjuntos compactos, $m_j \in \mathbf{N}, |U^j| = \infty, \forall j \in J ; I, J \subseteq \mathbf{N}$ conjuntos de cardinalidad finita; $X_0 \subseteq \mathbf{R}^n$.

La solución numérica del problema de optimización semi infinito siempre involucra alguna forma de discretización. Existen considerables evidencias empíricas que sugieren que la aproximación más eficiente computacionalmente, es hacer discretizaciones gradualmente refinadas, en contrapartida al hecho de hacer una única discretización densa. La explicación heurística del éxito de este tipo de aproximación se debe al hecho de que cuando se está lejos de la solución una discretización poco densa no interfiere en el progreso de la convergencia hacia la solución (Ref [23a]).

La técnica que usamos en la resolución del problema (*OSI - P*) se basa en descomponer dicho problema en una familia de problemas de Programación Matemática cuyas restricciones están en un número finito y que provienen de sucesivas discretizaciones del conjunto de parámetros "U", es decir dada una sucesión de discretizaciones $\{U_k\}, k \in \mathbf{N}$ tal que $U_k \subset U_{k+1} \subseteq U, |U_k| < \infty$ se puede asociar una sucesión de problemas

$\{P(U_k)\}$ $k \in \mathbf{N}$ tal que cuando $U_k \rightarrow U$, $k \rightarrow \infty$, $P(U_k)$ se aproxime a $P(U)$. En estas circunstancias cada problema $P(U_k)$ está definido por:

$$\text{Min}\{f(x)/x \in X(U_k)\}$$

donde

$$X(U_k) = \{x \in X_0/g_i(x) \leq 0, \phi_j(x, u) \leq 0; i \in I, j \in J, \forall u \in U_k^j\}$$

La resolución de los problemas $P(U_k)$ se ejecuta bajo un esquema de Diagonalización ver (Ref.[2,7,23a]), es decir, el algoritmo que resuelve cada uno de los problemas ejecuta un número finito de iteraciones hasta que cierta condición dada a priori es satisfecha, y luego la última iteración es usada como punto de partida para el próximo problema $P(U_{k+1})$.

En este trabajo cada problema de finitas restricciones $P(U_k)$ es resuelto por un algoritmo Quasy-Newton de puntos interiores correspondiente a la familia de algoritmos desarrollados en (Ref.[8]) . Para la generación de cada problema $P(U_k)$ proponemos un algoritmo externo el cual denominamos de Algoritmo Maestro y la resolución numérica de este es ejecutada por un algoritmo Interno que corresponde al antes mencionado.

En la presente sección presentamos el concepto de consistencia asociado al uso de la técnica de Diagonalización en la solución de problemas de programación no lineal. En general se definen dos conceptos de consistencia uno relativo a la convergencia de los problemas aproximados $P(U_k)$ con respecto al problema $P(U)$, y otro relativo a la convergencia de los puntos estacionarios de $P(U_k)$ con respecto a los puntos estacionarios de $P(U)$. Esto último es interpretado a través del concepto de función de optimalidad.

2.1. Aproximaciones Consistentes

A continuación presentamos un resumen de las principales definiciones y resultados relacionados con el concepto de aproximaciones consistentes introducido en (Ref.[22c]).

Sea “ B ” un espacio vectorial normado, con norma $\| \cdot \|$. Consideremos el problema de programación no lineal P):

P)

$$\text{Min}\{f(x)/x \in X\}$$

donde $f : B \rightarrow \mathbf{R}$ es una función semi continua inferior y $X \subseteq B$ es el conjunto de restricciones.

Sea $\{B_n\}$, $n \in \mathbf{N}$ una familia de sub-espacios de “ B ” de dimensión finita tal que $B_n = B$, siempre que B es de dimensión finita, y $B_n \subseteq B_{n+1}$, $\forall n \in \mathbf{N}$ de otra manera. Consideremos la siguiente familia de problemas aproximados al problema P) denominados como P_n) y definidos por:

$$P_n) \quad \text{Min}\{ f_n(x) / x \in X_n, n \in \mathbf{N} \}$$

donde $f_n : B_n \rightarrow \mathbf{R}$ es una función semi continua inferior y $X_n \subseteq B_n$, $\forall n \in \mathbf{N}$.

Considerando los Epígrafes de los problemas P) y P_n) denotados por $E \subseteq \mathbf{R} \times B$ y $E_n \subseteq \mathbf{R} \times B_n$ respectivamente y definidos por:

$$E = \{(x, y) / x \in X, f(x) \leq y\},$$

$$E_n = \{(x, y) / x \in X_n, f_n(x) \leq y\}.$$

Puede observarse que para que el problema P_n) converga Epigráficamente al problema P) el Epígrafe E_n debe converger al Epígrafe E en el sentido de Kuratowski (Ref[1]).

Definición 2.1.1.-

Una sucesión de problemas $\{P_n\}$, $n \in \mathbf{N}$ se dice que converge Epigráficamente a un problema P ($P_n \xrightarrow{Epi} P$) si:

a) $\forall x \in X$ existe una sucesión $\{x_n\}$, $n \in \mathbf{N}$ con $x_n \in X_n$ tal que $x_n \rightarrow x$ y $\limsup\{f_n(x_n)\} \leq f(x)$.

b) Para toda sucesión $\{x_n\}$, $n \in \mathbf{N}$, $K \subseteq \mathbf{N}$ que satisface $x_n \in X_n$, $\forall n \in \mathbf{N}$, y $x_n \rightarrow x$ debe ocurrir que $x \in X$ y $\liminf\{f_n(x_n)\} \geq f(x)$.

La convergencia Epigráfica puede verse como una aproximación de orden cero. En particular de ella se desprende el siguiente resultado.

Teorema 2.1.2.-

Supongamos que $P_n \rightarrow P$ Epigráficamente, entonces:

a) Si $\{x_n^*\}$, $n \in \mathbf{N}$ es una sucesión de mínimos globales de la familia de problemas $\{P_n\}$ y x^* un punto de acumulación, entonces x^* es un minimizador global del problema P .

b) Si $\{x_n^*\}$, $n \in \mathbf{N}$ es una sucesión de mínimos locales uniformemente estrictos de la familia de problemas $\{P_n\}$ y x^* es un punto de acumulación, entonces x^* es un minimizador local del problema P .

Para una demostración de este Teorema ver (Ref. [1,22c]).

El concepto de Epiconvergencia no garantiza que los puntos estacionarios de cada problema P_n convergan a los puntos estacionarios del problema P . Para eliminar la posibilidad de esta patología se impone una segunda condición a través del concepto de función de optimalidad como herramienta para forzar la convergencia de las derivadas (aproximación de primer orden) y restringir las estructuras de los conjuntos viables X y X_n .

Definición 2.1.3.-

a) Sea $D \subseteq B$ y $\theta : D \rightarrow \mathbf{R}$ una función, diremos que θ es una función de optimalidad para el problema P si:

- i) $X \subseteq D$
- ii) $\theta(x)$ es semi continua superior
- iii) $\theta(x) \geq 0 \forall x \in D$.
- iv) Para $x^* \in X$, $\theta(x^*) = 0$ si y sólo si x^* es un punto estacionario de P .

Análogamente, sea $D_n \subseteq B_n$ y $\theta_n : D_n \rightarrow \mathbf{R}$ una función, diremos que θ_n es una función de optimalidad para el problema P_n si:

- i) $X_n \subseteq D_n$
- ii) $\theta_n(x)$ es semi continua superior
- iii) $\theta_n(x) \geq 0, \forall x \in D_n$
- iv) Para $x_n^* \in X_n$, $\theta_n(x_n^*) = 0$ si y sólo si x_n^* es un punto estacionario de P_n .

Definición 2.1.4.-

Consideremos los problemas P y P_n y sean $\theta(x)$ y $\theta_n(x)$, $n \in \mathbf{N}$ las funciones de optimalidad asociadas respectivamente. Se dice que los pares (P_n, θ_n) de la sucesión $\{(P_n, \theta_n)\}$, $n \in \mathbf{N}$ son aproximaciones débilmente consistentes al par (P, θ) si:

- i) $P_n \rightarrow P$ Epigráficamente.
- ii) Para toda sucesión $\{x_n\}$, $n \in K$, $K \subseteq \mathbf{N}$ tal que $x_n \in X_n$ y $x_n \rightarrow x$, debe implicar que $\limsup\{\theta_n(x_n)\} \leq \theta(x)$.

En general para que se obtengan aproximaciones consistentes es necesario adicionar una calificación de restricciones la cual se da en la siguiente definición.

Definición 2.1.5.-

Dadas $\theta(x)$ y $\theta_n(x)$, $n \in \mathbf{N}$ funciones de optimalidad asociadas a los problemas P y P_n respectivamente. Diremos que la sucesión de pares $\{(P_n, \theta_n)\}$, $n \in \mathbf{N}$ son aproximaciones consistentes al par (P, θ) :

- i) Si dichas aproximaciones son débilmente consistentes y
- ii) $\theta(x) > 0, \forall x \notin X; \theta_n(x) > 0, \forall x \notin X_n, n \in \mathbf{N}$.

2.2. Un Algoritmo modelo de aproximaciones consistentes para Programación Matemática.

Consideremos un problema de Programación Matemática no Lineal, definido en forma simple por el siguiente problema P):

$$P) \quad \text{Min}\{ f(x) / \psi(x) \leq 0\}$$

y supongamos que el problema P) se puede aproximar mediante una familia de problemas $\{P_n\}$, $n \in \mathbf{N}$ definida por:

$$P_n) \quad \text{Min}\{ f_n(x) / \psi_n(x) \leq 0\}$$

En vista de las definiciones de la sección anterior, suponemos las siguientes hipótesis

Suposición 2.2.1.-

- i) Las funciones $f, \psi : X \rightarrow \mathbf{R}$ y $f_n, \psi_n : X_n \rightarrow \mathbf{R}; n = 1, 2, 3, \dots$ son continuas.
- ii) Existen funciones de optimalidad continuas $\theta : X \rightarrow \mathbf{R}$ para el problema P) y $\theta_n : X_n \rightarrow \mathbf{R}$ para P_n) $n = 1, 2, 3, \dots; X \subseteq R^n, X_n \subseteq R^n$
- iii) Si $\{x_n\}$, $n \in \mathbf{N}$ es tal que $x_n \in X_n$ y $x_n \rightarrow x^*$ cuando $n \rightarrow \infty$ entonces $\theta_n(x_n) \rightarrow \theta(x^*)$ cuando $n \rightarrow \infty$.
- iv) P_n converge epigráficamente a P .

Las hipótesis contenidas en la Suposición 2.2.1 aseguran que el par (P_n, θ_n) es una aproximación débilmente consistente al par (P, θ) .

Algoritmo Modelo

Dados $n_0 \in \mathbf{N}$, $x_0 \in X_{n_0}$, $\sigma : \mathbf{N} \rightarrow \mathbf{R}^+$ función decreciente, tal que $\sigma(n) \rightarrow 0$ cuando $n \rightarrow \infty$.

0.- Haga $i = 0, n = n_0$

- 1.- Calcular $x_{i+1} \in A(x_i)$
- 2.- Si $\theta_n(x_{i+1}) \leq \sigma(n)$ y $\psi_n(x) \leq \sigma(n)$ haga $x_n^* = x_{i+1}$, $n = n + 1$
- 3.- Haga $i = i + 1$ y vaya a (1).

Observación.-

En el paso (1) la función $A : \mathbf{R}^n \rightarrow P(\mathbf{R}^n)$ es una multifunción que representa en este caso un algoritmo de Programación Matemática aplicado a P_n .

La diferencia sustancial con el Modelo propuesto en [22c], radica en la introducción de la función $\sigma(n)$ la cual puede reflejar una sucesión para métodos de Penalización, o una sucesión de parámetros para métodos con aproximación de funciones o simplemente puede indicar una medida de la magnitud de la malla, para métodos que usan discretización, como es nuestro caso.

Teorema 2.2.2.-

Si la Suposición 2.2.1 es satisfecha, y existe una función decreciente positiva $\sigma(n)$ convergente a cero tal que para $n \geq n_0$, $n_0 \in \mathbf{N}$. Todo punto de acumulación \tilde{x} de la sucesión $\{x_i\}$, $i \in \mathbf{N}$ construida según el Algoritmo $x_{i+1} \in A(x_i)$ satisface $\theta_n(\tilde{x}) = 0$ y $\psi(\tilde{x}) \leq 0$. Consideremos además las sucesiones $\{x_i\}$ y $\{x_n^*\}$, construidas por el Algoritmo Modelo.

i) Si la sucesión $\{x_n^*\}$ es finita, entonces la sucesión $\{x_i\}$ no tiene puntos de acumulación.

ii) Si la sucesión $\{x_n^*\}$ es infinita, entonces todo punto de acumulación \tilde{x} de $\{x_n^*\}$, satisface $\theta(\tilde{x}) = 0$ y $\psi(\tilde{x}) \leq 0$.

3. Implementación.

Basados en el Algoritmo Modelo de la sección anterior, en esta sección implementamos dicho modelo para el problema de Optimización Semi Infinito:

$$P(U)$$

$$\text{Min}\{ f(x) / x \in X(U) \}$$

donde

$$X(U) = \{x \in X_0 / g_i(x) \leq 0, \phi_j(x, u) \leq 0; i \in I, j \in J, \forall u \in U^j\}$$

$f, g_i : \mathbf{X}_0 \rightarrow \mathbf{R}$; $\phi_j : \mathbf{X}_0 \times U^j \rightarrow \mathbf{R}$ funciones ; $U^j \subseteq \mathbf{R}^{m_j}$ conjuntos compactos, $m_j \in \mathbf{N}$, $|U^j| = \infty, \forall j \in J$; $I, J \subseteq \mathbf{N}$ conjuntos de cardinalidad finita; $X_0 \subseteq \mathbf{R}^n$.

Asímismo el problema aproximado está definido por:

$$P(U_k) \quad \text{Min}\{ f(x) / x \in X(U_k) \}$$

donde

$$.X(U_k) = \{x \in X_0 / g_i(x) \leq 0, \phi_j(x, u) \leq 0; i \in I, j \in J, \forall u \in U_k^j\}$$

Debe observarse que respecto a la teoría general no hemos aproximado la función objetivo ni el conjunto de restricciones, es decir, $g_{ki}(x) = g_i(x)$; $\phi_k(x, u) = \phi(x, u) \forall u \in U_k$ y $f_k(x) = f(x)$.

La estrategia para discretizar el conjunto “ U ” está implícita en el algoritmo modelo, en el sentido que describe una manera pre-planificada para conducir la discretización a través de la función decreciente “ $\sigma(k)$ ”. Para el caso específico de Optimización Semi Infinita la función $\sigma(k)$ que se introduce generaliza el concepto de “magnitud de paso de la discretización”, lo cual permite extender el Algoritmo Modelo original a mallas de \mathbf{R}^m , $m > 1$. Es necesario aclarar que el tipo de mallas al que nos referimos son “mallas de tipo regular”.

A manera de comprender mejor cómo se puede construir la función decreciente $\sigma(k)$, supongamos que $U \subset \mathbf{R}^m$ y $\{h_k\} \subset \mathbf{R}^m$, $k \in \mathbf{N}$ es una sucesión de vectores que definen el paso de la discretización $h_k > 0$, $h_k \rightarrow 0$ cuando $k \rightarrow \infty$, tal que para “ k ” fijo $U_k \subset U$, entonces la función $\sigma(k)$ puede definirse en relación a la malla U_k como $\sigma(k) = \| h_k \|$ ó también, $\sigma(k) = \| h_k \|_\infty$.

3.1. Notación Definiciones y Demostraciones.

Sean $I_i = [\alpha_i, \beta_i]$ intervalos finitos $i = 1, \dots, m$ tal que $U \subseteq I_1 \times I_2 \times \dots \times I_m \subseteq \mathbf{R}^m$. El vector de paso que define la discretización del conjunto $\prod(\times I_i)$, $i = 1, \dots, m$ se define por $h \in \mathbf{R}^m$, $h > 0$, $h = (h^1, h^2, \dots, h^m)$ tal que $h^i = \frac{(\beta_i - \alpha_i)}{t}$, $t \in \mathbf{N}$, $t \geq 2$, $i = 1, \dots, m$.

Una partición o malla del conjunto $\prod(\times I_i)$, $i = 1, \dots, m$ está definida por:

$$I[h] = \{u = (u_1, u_2, \dots, u_m) / u_i = \alpha_i + kh_i, k = 0, \dots, t; i = 1, \dots, m\}$$

De esta manera se genera una malla para el conjunto U , definida por:

$$U[h] = I[h] \cap U$$

Sea $U[h_k] \subseteq U$ una discretización o malla del conjunto " U " se define y denota por $d(U[h])$ a la densidad de malla

$$d(U[h_k]) = \max_{u \in U} \{ \min_{u^* \in U[h]} \| u - u^* \| \}.$$

Nótese que cuando $d(U[h_k]) \rightarrow 0$, con $h_k \rightarrow 0$, $\| u - u^* \| \rightarrow 0$, $\Rightarrow U[h_k] \rightarrow U$.

En lo sucesivo denotaremos los problemas $P(U_k)$ por $P(U[h_k])$ para ser consecuentes con la notación anterior en que hemos cambiado la malla U_k por $U[h_k]$ donde $\{h_k\} \subseteq \mathbf{R}^m$, $k \in \mathbf{N}$ es una sucesión de vectores que definen el paso en la discretización tal que $h_k \rightarrow 0$ cuando $k \rightarrow \infty$.

A continuación demostraremos las hipótesis requeridas en la Suposición 3.1 para justificar la validez y aplicabilidad del Algoritmo Modelo y también para definir la implementación en sí.

Suposición 3.1.1-

a) Para todo $i \in I$; $j \in J$, las funciones g_i , ϕ_j son dos veces diferenciables y continuas, respecto al primer argumento, y ϕ_j es continua respecto al segundo argumento.

b) Para todo $i \in I$; $j \in J$, $\nabla g_i(\cdot)$, $\nabla^2 g_i(\cdot)$, $\nabla_x \phi_j(\cdot, \cdot)$, $\nabla_{xx} \phi_j(\cdot, \cdot)$ son funciones continuas.

Teorema 3.1.2.-

El problema $P(U[h_k])$ converge Epigráficamente al problema $P(U)$.

Demostración.-

Por definición hemos tomado $f_k(x) = f(x)$, además como $X(U) \subseteq X(U[h_k])$, $\forall k \in \mathbf{N}$ dado $x \in X(U) \Rightarrow x \in X(U[h_k])$ en consecuencia podemos definir una sucesión $\{x_k\}$, $\forall k \in \mathbf{N}$ por $x_k = x$, $\forall k \in \mathbf{N}$, es claro que $x_k \rightarrow x$ y además como f es continua, $\lim f_k(x_k) = \lim f(x_k) = f(x)$, lo que demuestra la parte (a) de la Definición 2.1.1.

b) Sea $\{x_k\}$, $\forall k \in \mathbf{N}$ una sucesión tal que $x_k \in X(U[h_k])$ y $x_k \rightarrow x$ cuando $k \rightarrow \infty$ como

$g_i(x_k) \leq 0$, y $\phi_j(x_k, u) \leq 0$, $\forall u \in U^j[h_k]$, $i \in I$, $j \in J$ y $U^j[h_k] \rightarrow U^j$ cuando $k \rightarrow \infty$. Por otro lado de la continuidad de g_i , ϕ_j se tiene que

$g_i(x_k) \rightarrow g_i(x) \leq 0, \phi_j(x_k, u) \rightarrow \phi_j(x, u) \leq 0, \forall u \in U, i \in I, j \in J$ lo cual implica $x \in X(U)$, esto demuestra (b) en la Definición 2.1.1. □

Para caracterizar que la aproximación es consistente es necesario definir funciones de optimalidad para los problemas $P(U)$ y $P(U[h_k])$. Aunque en la literatura se encuentran definidas algunas funciones de optimalidad, en esta ocasión definimos funciones de optimalidad que resultan natural del Algoritmo Interno por sus características.

Sea $\{\varepsilon_n\}, n \in N, \varepsilon_n > 0$ una sucesión tal que $\varepsilon_n \rightarrow 0, n \rightarrow \infty$, se definen los siguientes conjuntos:

$$U_{\varepsilon_n}^j(x) = \{u \in U^j[h_n] / -\varepsilon_n \leq \phi_j(x, u) \leq 0\}, x \in X(U[h_n]), \forall j \in J$$

$$I_{\varepsilon_n}(x) = \{i \in I / -\varepsilon_n \leq g_i(x) \leq 0\}$$

$$U_0^j(x) = \{u \in U^j / \phi_j(x, u) = 0\}, x \in X(U), \forall j \in J$$

$$I_0(x) = \{i \in I / g_i(x) = 0\}$$

Los conjuntos $U_{\varepsilon_n}^j, I_{\varepsilon_n}$ son conocidos en la literatura como conjuntos de restricciones ε_n -activas.

Suposición 3.1.3.-

$\forall x \in \mathbf{R}^n, U_0^j(x), \forall j \in J$ es finito.

Definición 3.1.4.-

Sean $\theta, \theta_k : \mathbf{R}^n \rightarrow \mathbf{R}$ funciones asociadas a los problemas $P(U)$ y $P(U[h_k])$ respectivamente y definidas por:

$$\theta(x) = \{1/2 \|\nabla f(x) + \lambda_1^t \nabla_x \phi(x, u) + \lambda_2^t \nabla g(x)\|^2 + 1/2 \|\lambda_1 \phi(x, u) + \lambda_2 g(x)\|^2 / u \in \bigcup_j U_0^j(x), j \in J, i \in I_0(x)\}$$

$$\theta_k(x) = \{1/2 \|\nabla f(x) + \lambda_1^t \nabla_x \phi(x, u) + \lambda_2^t \nabla g(x)\|^2 + 1/2 \|\lambda_1 \phi(x, u) + \lambda_2 g(x)\|^2 / u \in \bigcup_j U_{\varepsilon_k}^j(x), j \in J, i \in I_{\varepsilon_k}(x)\}$$

donde $\phi \in R^{p_\varepsilon}, g \in R^{q_\varepsilon}$ con $p_\varepsilon = |\bigcup_j U_{\varepsilon_k}^j(x)|, q_\varepsilon = |I_{\varepsilon_k}(x)|; \lambda_1 \in R^{p_\varepsilon},$

$\lambda_2 \in R^{q_\varepsilon}$ son solución de la transformación lineal $F : \mathbf{R}^l \rightarrow \mathbf{R}^l$ definida por el sistema:

$$\begin{aligned} Bd + \nabla_x \phi(x, u)^t \lambda_1 + \nabla g(x)^t \lambda_2 &= -\nabla f(x) \\ \Lambda(\nabla \phi(x, u), \nabla g(x))^t d + \text{diag}(\phi(x, u), g(x))^t (\lambda_1, \lambda_2) &= 0 \end{aligned} \quad (*)$$

donde $B \in \mathbf{R}^{n \times n}$ definida positiva; $\Lambda = \text{diag}(\bar{\lambda}_1, \bar{\lambda}_2) \in \mathbf{R}^{p_\varepsilon \times q_\varepsilon}$; $\bar{\lambda}_1, \bar{\lambda}_2$ valores anteriores, λ_1, λ_2 , son los valores actuales, $l = n + p_\varepsilon + q_\varepsilon$.

Observación.-

El sistema (*) resulta de aplicar el método de Newton a las condiciones de optimalidad de primer orden de K.K.T. en que el número de restricciones corresponde a la cardinalidad del conjunto $\bigcup_j U_{\varepsilon_k}^j(x) \cup I_{\varepsilon_k}(x), \forall j \in J$.

En [8] se prueba que F es inyectiva, con lo cual la solución λ_1, λ_2 se obtiene de forma única y continua.

Proposición 3.1.5.-

Dadas las sucesiones $\{x_n\}, n \in \mathbf{N}, x_n \in X(U[h_n]), \{\varepsilon_n\}, n \in \mathbf{N}, \varepsilon_n > 0$ tal que $\varepsilon_n \rightarrow 0, x_n \rightarrow x$, cuando $n \rightarrow \infty$ entonces $\lim U_{\varepsilon_n}^j(x_n) = U_0^j(x), \forall j \in J$.

Demostración.-

Sea $\{u_{n_j}\}, n_j \in K$ con $K \subseteq \mathbf{N}$ tal que $u_{n_j} \rightarrow u_j, u_{n_j} \in U_{\varepsilon_n}^j(x_n)$ demostraremos que $u_j \in U_0^j(x), \forall j \in J$. En efecto $\phi_j(x_n, u_{n_j}) \rightarrow \phi_j(x, u_j)$ por ser continua, luego el conjunto

$$U_{\varepsilon_n}^j(x_n) = \{u \in U^j[h_n] / -\varepsilon_n \leq \phi_j(x_n, u_{n_j}) \leq 0\}, x_n \in X(U[h_n])$$

cuando $n \rightarrow \infty$, converge trivialmente a

$$U_0^j(x) = \{u \in U^j / \phi_j(x, u) = 0\}, x \in X(U), u \in U^j(x).$$

$\forall j \in J, \square$.

Proposición 3.1.6.-

Dadas las sucesiones $\{x_n\}, n \in \mathbf{N}, x_n \in X(U[h_n]), \{\varepsilon_n\}, n \in \mathbf{N}, \varepsilon_n > 0$ tal que $\varepsilon_n \rightarrow 0, x_n \rightarrow x$, cuando $n \rightarrow \infty$ entonces, $\lim I_{\varepsilon_k}(x) = I_0(x)$.

Demostración.-

Por la continuidad de las funciones $g_i : X_0 \rightarrow R, \forall i \in I$ se tiene que $g_i(x_n) \rightarrow g_i(x)$, cuando $n \rightarrow \infty$ como consecuencia de esto y del hecho que la sucesión $\varepsilon_n \rightarrow 0$, se tiene que la sucesión de conjuntos $I_{\varepsilon_n}(x_n) = \{i \in I / -\varepsilon_n \leq g_i(x_n) \leq 0\}$ converge al conjunto $I_0(x) = \{i \in I / g_i(x) = 0\}$ cuando $n \rightarrow \infty$. \square

Proposición 3.1.7.-

Las funciones $\theta, \theta_k : \mathbf{R}^n \rightarrow \mathbf{R}$ son funciones de optimalidad para los problemas $P(U)$ y $P(U[h_k])$ respectivamente.

Demostración.-

i) De la definición resulta claro que $X(U) \subseteq \mathbf{R}^n$ y $X(U[h_k]) \subseteq \mathbf{R}^n$

ii) De la Suposición 3.1.1 a) y b) tenemos que $g, f \in C^2(\mathbf{R}^n), \phi \in C^2(\mathbf{R}^n \times U)$ respecto a "x", además $\lambda_1 \in \mathbf{R}^{p_\epsilon}, \lambda_2 \in \mathbf{R}^q$ son soluciones únicas del sistema lineal en la Definición 3.1.4 como consecuencia se obtiene $\lambda_1(x), \lambda_2(x)$ en forma continua $\forall x \in X(U[h_k])$, de donde se concluye que $\nabla f(x) + \lambda_1^t \nabla_x \phi(x, u) + \lambda_2^t \nabla g(x) \in \mathbf{R}^n, \forall u \in \bigcup_j U_{\epsilon_k^j}(x)$ es continua, tambien es

inmediato que $\lambda_1^t \phi(x, u), \lambda_2^t g(x)$ son continuas. Finalmente como la función $\| \cdot \|_2 : \mathbf{R}^n \rightarrow \mathbf{R}$ es continua se concluye que θ_k es continua, $\forall x \in X(U[h_n])$. Similarmente y teniendo presente que de la Proposición 3.1.5, se concluye que $p_{\epsilon_k} \rightarrow p$ cuando $\epsilon_k \rightarrow 0$, donde $p = |\bigcup_j U_0^j|$ se demuestra que θ es

continua $\forall x \in X(U), u \in (\bigcup_j U_0^j)$.

iii) Obvio de la definición $\theta(x) \geq 0$, y $\theta_k(x) \geq 0 \forall x \in \mathbf{R}^n$.

iv) \Rightarrow) Sea $x_k^* \in X(U[h_k])$ y supongamos que $\theta_k(x_k^*) = 0$, esto implica que $\nabla f(x) + \lambda_1^t \nabla_x \phi(x) + \lambda_2^t \nabla g(x) = 0, \lambda_1^t \phi(x, u) = 0, \lambda_2^t g(x) = 0, \forall u \in \bigcup_j$

$U^j[h_k]$ (esto porque $\lambda_1 = \lambda_2 = 0, \forall u \in \bigcup_j \{U^j[h_k] - U_{0_k^j}(x)\}$, donde $U_{0_k^j}(x)$

es el conjunto de las restricciones activas en el conjunto $U_{\epsilon_k^j}(x_k), \forall j \in J$), luego x_k^* es punto estacionario del problema $P(U[h_k])$. Similarmente si $x^* \in X$ es tal que $\theta(x^*) = 0$ entonces $\nabla f(x) + \lambda_1^t \nabla_x \phi(x) + \lambda_2^t \nabla g(x) = 0, \lambda_1^t \phi(x, u) = 0, \lambda_2^t g(x) = 0, \forall u \in (\bigcup_j U^j)$ (esto porque $\lambda_1 = \lambda_2 = 0, \forall u \in \bigcup_j$

$\{U^j - U_0^j\}, \forall j \in J$ si la Suposición 3.1.3 es verdadera) luego podremos concluir que x^* , es punto estacionario de $P(U)$.

\Leftarrow) La demostración resulta trivial, en efecto, si x_k^* es punto estacionario de $P(U[h_k])$ entonces $\nabla f(x) + \lambda_1^t \nabla_x \phi(x) + \lambda_2^t \nabla g(x) = 0, \lambda_1^t \phi(x, u) = 0, \lambda_2^t g(x) = 0, \forall u \in \bigcup_j U^j[h_k]$, como $\bigcup_j U_{0_k^j}(x) \cup \bigcup_j U_{\epsilon_k^j}(x) \subset \bigcup_j U^j[h_k], \forall j \in J$

y $\bigcup_j U_{\epsilon_k^j}(x)$ contiene las restricciones activas entonces en particular $\lambda_1 =$

$\lambda_2 = 0, \forall u \in \bigcup_j \{U_{\epsilon_k^j}(x) - U_{0_k^j}(x)\}$ luego $\theta_k(x_k^*) = 0$. Análogamente se

prueba para la función θ . \square

Proposición 3.1.8.-

$\lim \theta_k(x_k) = \theta(x)$ cuando $x_k \rightarrow x, k \rightarrow \infty$.

Demostración.-

Como $x_k \rightarrow x, k \rightarrow \infty$, de la Proposición 3.1.5 y 3.1.6 se desprende $\lim \theta_k(x_k) =$
 $\lim_{k \rightarrow \infty} \{1/2 \| \nabla f(x_k) + \lambda_1^t \nabla_x \phi(x_k, u) + \lambda_2^t \nabla g(x_k) \|^2 + 1/2 \| \lambda_1 \phi(x_k, u) +$
 $\lambda_2 g(x_k) \|^2 /$
 $u \in \bigcup_j U_{\varepsilon_k}^j(x_k), j \in J, i \in I_{\varepsilon_k}(x_k) \}$
 $\{1/2 \| \nabla f(x) + \lambda_1^t \nabla_x \phi(x, u) + \lambda_2^t \nabla g(x) \|^2 + 1/2 \| \lambda_1 \phi(x, u) + \lambda_2 g(x) \|^2 /$
 $u \in \bigcup_j U_0^j(x), j \in J, i \in I_0(x) \} = \theta(x). \quad \square$

De las definiciones de $\theta(x)$, y $\theta_k(x)$ el Teorema 3.1.2, Proposición 3.1.6 y 3.1.7 se desprende que el par $P(U[h_k]), \theta_k$ es una aproximación débilmente consistente al par $(P(U), \theta)$.

Suposición 3.1.9.-

$\forall x \notin X(U), \theta(x) > 0$ y $\forall x \notin X(U[h_n]), \theta_n(x) > 0$.

Si la Suposición 3.1.8 es verdadera entonces el par $(P(U[h_k]), \theta_k)$ es una aproximación consistente al par $(P(U), \theta)$. En nuestro caso tal suposición es verdadera por definición de las funciones θ y θ_n .

El conjunto de Teoremas, Proposiciones, Definiciones, y Suposiciones en los numerales 3.1.2 hasta 3.1.8 constituyen la formalización y demostración de las hipótesis requeridas en la Suposición 2.2.1 para garantizar la aplicación del Algoritmo Modelo y obtener un algoritmo con aproximaciones consistentes.

Sea $\{\varepsilon_n\}, n \in N, \varepsilon_n > 0$ una sucesión tal que $\varepsilon_n \rightarrow 0, n \rightarrow \infty$, consideremos el conjunto

$$U_{\varepsilon_n}^j(x) = \{u \in U^j[h_n] / -\varepsilon_n \leq \phi_j(x, u) \leq 0\}, x \in X(U[h_n]), \forall j \in J$$

y su forma asimétrica

$$A(U_{\varepsilon_n}^j(x)) = \{u \in U^j[h_n] / \phi_j(x, u) \geq \varepsilon_n\}, x \in R^n, \forall j \in J$$

El conjunto $U_{\varepsilon_n}^j$ es conocido en la literatura como el conjunto de las restricciones ε_n -activas interiores

Algoritmo Maestro

Datos. - Seleccione sucesiones $\{h_k\} \subseteq \mathbf{R}^m, k \in \mathbf{N} \cup \{0\}, h_k > 0, h_k \rightarrow 0, k \rightarrow \infty; \{\epsilon_k\} \subset \mathbf{R}, k \in \mathbf{N} \cup \{0\}, \epsilon_k > 0, \epsilon_k \rightarrow 0, k \rightarrow \infty, \sigma : \mathbf{N} \rightarrow \mathbf{R}^+$ función decreciente, tal que $\sigma(k) \rightarrow 0$ cuando $k \rightarrow \infty. S = \emptyset, R = \emptyset, x_0 \in \mathbf{R}^n, r \in [0, 1).$

0.- $k = 0; n = 1$

1.- Si $\sigma(k) \leq r$ pare. Dado $U^j[h_k] \subseteq U^j, j \in J$ haga $S = S \cup \bigcup_j$

$A(U_{\epsilon_k^j}(x_k)), R = \bigcup_j U_{\epsilon_k^j}(x_k)$

2.- Determine $D[h_k] \subseteq R \cup S$ si $D[h_k] = \emptyset$ haga $D[h_k] = \bigcup_j U^j[h_k]$

3.- Encuentre una solución x_{k+1} de $P(D[h_k])$ tal que $\theta_n(x_{k+1}) \leq \sigma(k)$ existen dos posibilidades:

i) Si $\{\max_u \max_j \phi_j(x_{k+1}, u) / u \in \bigcup_j U^j[h_k]\} \leq \sigma(k)$ haga $x_n^* = x_{k+1};$

$k = k + 1, n = n + 1$ y vaya a (1)

ii) Caso contrario vaya a (4)

4.- Haga $D[h_k] = D[h_k] \cup \{u^*\}, S = S \cup \{u^*\}$ donde

$$\phi(x_{k+1}, u^*) = \max_u \max_j \{\phi_j(x_{k+1}, u) / u \in \bigcup_j U^j[h_k]\}$$

y vaya a (3).

Teorema 3.1.10.-

El Algoritmo Maestro resuelve el problema $P(U[h_k])$ en finitas iteraciones.

Demostración.-

Es suficiente probar que dado “k”, $k \in \mathbf{N}$, un número finito de pasos (3), (4) conduce a la solución de $P(U[h_k])$. En efecto esto se consigue del hecho que en el paso (4) $D[h_k] \subseteq \bigcup_j U^j[h_k]$ es aumentado en un punto $u^* \in \bigcup_j U^j[h_k]$ y $|\bigcup_j U^j[h_k]| < \infty. \square$

Observación.-

Cuando $\bigcup_j U^j[h_k], j \in J$ es reemplazado por $D[h_k]$ la solución del problema $P(D[h_k])$ tal que $\theta_n(x_{k+1}) \leq \sigma(k)$ no necesariamente resuelve el problema $P(U[h_k])$ en el sentido que se cumpla la condición $\{\max_u \max_j \phi_j(x_{k+1}, u)/u \in U[h_k]\} \leq \sigma(k).$

En el paso (2) el conjunto $D[h_k]$ se define de la siguiente manera:

$$\text{Si } R \cup S \neq \emptyset, D[h_k] = R \cup S - \{u \in S / \phi(x, u) < -\varepsilon_k\}.$$

3.2. Algoritmo Interno

Anteriormente se mencionó que cada problema de finitas restricciones $P(U[h_k])$ es resuelto por un algoritmo debido a Heskovits (Ref. [8]) que hemos denominado de algoritmo interno. La idea principal en que se basa este algoritmo consiste en resolver las ecuaciones K.K.T. mediante el método de Newton para definir la dirección de búsqueda. El algoritmo genera una secuencia de puntos interiores $\{x_n\}$, $n \in \mathbf{N}$ convergentes a un punto estacionario de K.K.T.

Si consideramos el problema $P(U[h_k])$ la búsqueda de un punto de K.K.T. se reduce a la solución de un sistema no lineal de “ $n + m + s$ ” incógnitas $(x, \lambda_1, \lambda_2)$ y “ $n + m + s$ ” ecuaciones donde $|I| = s \mid \bigcup_j U^j[h_k] \mid = m$.

$$\begin{aligned} \text{i) } & \nabla f(x) + \lambda_1^t \nabla_x \phi(x, u) + \lambda_2^t \nabla g(x) = 0 \\ \text{ii) } & \lambda_1^t \phi(x, u) = 0 \quad ; \lambda_2^t g(x) = 0, \forall u \in \bigcup_j U^j[h_k] \\ \text{iii) } & \phi(x, u) \leq 0 \quad , g(x) \leq 0, \forall u \in \bigcup_j U^j[h_k] \\ \text{iv) } & \lambda_1, \lambda_2 \geq 0, \lambda_1 \in \mathbf{R}^m, \lambda_2 \in \mathbf{R}^s \end{aligned} \quad (1)$$

El método resuelve el sistema (i) y (ii) linealizándolo por el método de Newton encontrando soluciones $(x, \lambda_1, \lambda_2)$ que satisfacen las inecuaciones (iii) y (iv). Una iteración de Newton en la solución de (i) y (ii) parte de un punto $((x_k, \lambda_{1k}, \lambda_{2k}))$ linealizando (i) y (ii) en una vecindad de dicho punto y definiendo una nueva estimación $(x_{k+1}, \lambda_{1k+1}, \lambda_{2k+1})$ solución del sistema lineal

$$\begin{aligned} & \begin{bmatrix} B & (\nabla_x \phi(x, u), \nabla g(x)) \\ \Lambda(\nabla_x \phi(x, u)^t, \nabla g(x)) & G(x, u) \end{bmatrix} \begin{bmatrix} x_{k+1} - x_k \\ \lambda_{1(k+1)} - \lambda_{1(k)} \\ \lambda_{2(k+1)} - \lambda_{2(k)} \end{bmatrix} = \\ & = - \begin{bmatrix} \nabla f(x) + \lambda_1^t \nabla_x \phi(x, u) + \lambda_2^t \nabla g(x) \\ G(x, u)(\lambda_1, \lambda_2) \end{bmatrix} \end{aligned} \quad (2)$$

$$G(x, u) = \text{diag}\{(\phi(x, u), g(x))/u \in \bigcup_j U^j[h_k]\}; \Lambda = \{(\lambda_1, \lambda_2) \in \mathbf{R}^m \times \mathbf{R}^s\}.$$

La matriz $B \in \mathbf{R}^{n \times n}$ es aproximada por el método BFGS lo cual hace que B sea simétrica definida positiva. Si en la ecuación matricial (2) permutamos términos, simplificando y substituyendo $d_k = x_{k+1} - x_k$, obtenemos el siguiente sistema:

$$\begin{aligned} Bd + \nabla_x \phi(x, u)^t \lambda_1 + \nabla g(x)^t \lambda_2 &= -\nabla f(x) \\ \Lambda(\nabla \phi(x, u), \nabla g(x))^t d + \text{diag}(\phi(x, u), g(x))^t (\lambda_1, \lambda_2) &= 0 \end{aligned} \tag{3}$$

La solución “ d ” define la dirección de búsqueda del método. Obsérvese que en los puntos estacionarios el sistema es homogéneo, en consecuencia la dirección d es cero. Sin embargo d no es útil como dirección de búsqueda debido a que no siempre ella constituye un campo de direcciones viables. Esto se debe al hecho de que cuando muchas restricciones son activas ($\phi(x, u) = 0, u \in U[h_n]; g(x)$) el término $G(x, u) \approx 0$ se aproxima a cero forzando d a la dirección tangente del conjunto viable. Para evitar este problema se hace una pequeña deflexión en la dirección d de forma que la nueva dirección defina una dirección hacia el interior del conjunto viable. La deflexión se consigue perturbando la segunda ecuación de (3) resultando el siguiente sistema.

$$\begin{aligned} Bd^1 + \nabla_x \phi(x, u)^t \lambda_1^1 + \nabla g(x)^t \lambda_2^1 &= -\nabla f(x) \\ \Lambda(\nabla \phi(x, u), \nabla g(x))^t d^1 + \text{diag}(\phi(x, u), g(x))^t (\lambda_1^1, \lambda_2^1) &= \\ = -\Lambda \omega; \omega > 0, \omega \in \mathbf{R}^{m+s} & \end{aligned} \tag{4}$$

Teniendo en vista lo anterior la dirección de búsqueda del algoritmo interno se define por:

$$d^2 = d + \rho d^1 \quad \text{donde} \quad \rho \in \mathbf{R}, \rho > 0$$

ALGORITMO Parámetros. $\alpha, \eta, \nu \in (0, 1), \varphi > 0$.

Datos. $x \in X(U_n), B \in \mathbf{R}^{n \times n}$ simétrica definida positiva, $\omega, \lambda \in \mathbf{R}^m, \omega > 0, \lambda > \theta$.

0.- $k = 0$

1.- Determinación de la dirección de búsqueda.

(i) Calcule $(d, \lambda_1, \lambda_2)$ resolviendo el sistema lineal.

$$\begin{aligned} Bd + \nabla_x \phi(x, u)^t \lambda_1 + \nabla g(x)^t \lambda_2 &= -\nabla f(x) \\ \Lambda(\nabla \phi(x, u), \nabla g(x))^t d + \text{diag}(\phi(x, u), g(x))^t (\lambda_1, \lambda_2) &= 0 \end{aligned}$$

Si $d = 0$, pare.

(ii) Calcule $(d^1, \lambda_1^1, \lambda_2^1)$ resolviendo el sistema lineal.

$$Bd^1 + \nabla_x \phi(x, u)^t \lambda_1^1 + \nabla g(x)^t \lambda_2^1 = -\nabla f(x)$$

$$\Lambda(\nabla \phi(x, u), \nabla g(x))^t d^1 + \text{diag}(\phi(x, u), g(x))^t (\lambda_1^1, \lambda_2^1) = -\Lambda \omega$$

(iii) Si $(d^1) \nabla f(x) > 0$, haga

$$\rho = \inf\{\varphi \|d\|^2, (\alpha - 1)(d) \nabla f(x) / (d^1) \nabla f(x)\}$$

De otra manera

$$\rho = \varphi \|d\|^2$$

(iv) Calcule la dirección de búsqueda

$$d^2 = d + \rho d^1; \lambda^2 = \lambda + \rho \lambda^1$$

2. - Búsqueda lineal

Calcule t , el primer número de la sucesión $\{1, \nu, \nu^2, \nu^3, \dots\}$ que satisfaga

$$(i) f(x + td^2) \leq f(x) + t\eta \nabla f(x) d^2 \quad y$$

$$(ii) \phi(x + td^2, u) < 0 \text{ si } \lambda_1^2 \geq 0$$

$$g(x + td^2) < 0 \text{ si } \lambda_2^2 \geq 0$$

ó

$$(iii) \phi(x + td^2, u) \leq \phi(x, u)$$

$$g(x + td^2) \leq g(x) \text{ de otra manera}$$

3. - Actualizaciones

(i) haga

$$x = x + td^2$$

y defina un nuevo valor para

$$\omega > 0; \lambda > 0; B(d.p.)$$

(ii) haga $k = k + 1$ y vaya al paso (1).

El algoritmo en el paso (2) ejecuta una búsqueda lineal inexacta basada en el método de Armijo para optimización irrestricta, junto con hacer prevalecer la condición de Goldstein para conseguir convergencia global.

En condiciones habituales de regularidad para las funciones “ f ” y “ ϕ ” la conducta asintótica del algoritmo se torna interesante ya que se puede obtener tasa de convergencia superlineal o cuadrática.

4. Resultados Numéricos.

Presentamos algunos resultados numéricos de problemas que se encuentran en la literatura (Ref [23, 27, 28, 29]). El Algoritmo Maestro se programó en Matlab y fue rodado en un Micro Computador 486.

Parámetros en el Algoritmo Maestro

A) Problemas uni - parámetros, $u \in [a_1, b_1] \subset R$
vector de paso

$$h_k = (b_1 - a_1)/2^k, \quad k = 2, 3, 4, \dots$$

$$\sigma(k) = \varepsilon_k = h_k, \quad k = 2, 3, \dots$$

Condición de parada : $\sigma(k) \leq 10^{-3}$

B) Problemas bi-parámetros, $u \in [a_1, b_1] \times [a_2, b_2] \subseteq R^2$
vector de paso :

$$h_k = (h_k^1, h_k^2); \quad h_k^i = (b_i - a_i)/2^k, \quad k = 2, 3, 4, \dots; \quad i = 1, 2$$

$$\sigma(k) = \varepsilon_k = \| h_k \|_\infty, \quad k = 2, 3, 4, \dots$$

Condición de parada : $\sigma(k) \leq 10^{-2}$

Parámetros en el Algoritmo Interno

$$\alpha = 0.8, \quad \nu = 0.8, \quad \rho = 1, \quad \varepsilon = 0.1$$

Los siguientes problemas fueron seleccionados :

Problema 1.-

$$U = [0, 1]$$

$$f(x) = 1/3x_1^2 + x_2^2 + 1/2x_1$$

$$\phi(x, u) = (1 - u^2x_1^2)^2 - u^2x_1 - x_2^2 + x_2$$

Punto inicial : $x_0 = (-1, -3)$ viable

$$\text{Solución} \quad : \quad x^* = (-0.74991, -0.618058)$$

$$f(x^*) = 0.194496$$

Punto inicial : $x_0 = (0, 0)$ no viable

$$\begin{aligned}\text{Solución} & : x^* = (-0.750002, -0.618909) \\ & f(x^*) = 0.19454\end{aligned}$$

Problema 2.-

$$U = [0, 1]$$

$$f(x) = x_1^2 + x_2^2 + x_3^2$$

$$\phi(x, u) = x_1 + x_2 e^{ux_3} + e^{2u} - 2\text{sen}(4u)$$

Punto inicial : $x_0 = (1, 1, 1)$ no viable

$$\begin{aligned}\text{Solución} & : x^* = (-0.2126, -1.3608, 1.8541) \\ & f(x^*) = 5.3347\end{aligned}$$

Punto inicial : $x_0 = (-9, 1/2, -5)$ viable

$$\begin{aligned}\text{Solución} & : x^* = (-0.209925, -1.37087, 1.85278) \\ & f(x^*) = 5.34007\end{aligned}$$

Problema 3.-

$$U = [0, 1]$$

$$f(x) = \sum_{i=1}^n e^{x_i}$$

$$\phi(x, u) = 1/(1 + u^2) - \sum_{i=1}^n x_i u^{i-1}$$

Para $n = 3$

Punto inicial : $x_0 = (1, 0.5, 0)$ viable

$$\begin{aligned}\text{Solución} & : x^* = (1.00533, -0.105815, -0.39817) \\ & f(x^*) = 4.30382\end{aligned}$$

Punto inicial : $x_0 = (-1, 5, 3)$ no viable

$$\begin{aligned}\text{Solución} & : x^* = (1.01084, -0.147015, -0.322512) \\ & f(x^*) = 4.310058\end{aligned}$$

Problema 4.-

$$U = [0, 1] \times [0, 1]$$

$$f(x) = x_1^2 + x_2^2 + x_3^2$$

$$\phi(x, u) = x_1(u_1 + u_2^2 + 1) + x_2(u_1u_2 - u_2^2) + x_3(u_1u_2 + u_2^2 + u_2) + 1$$

Punto inicial : $x_0 = (2, -1, 1)$ viable

Solución : $x^* = (-1.00028, 0.00298 - E004, -0.181 - E007)$

$$f(x^*) = 1.0000189$$

Punto inicial : $x_0 = (0, 1, -1)$ no viable

Solución : $x^* = (-1.00049, 1.55429 - E006, -1.55429 - E006)$

$$f(x^*) = 1.000097$$

En la tabla 1 se presenta un informe respecto a la evaluación del número de funciones y el número de gradientes de las funciones que ejecuta el Algoritmo, para ello usamos un índice compuesto referido en (Ref. [22c]), el que esta dado por la siguiente expresión:

$$NT = NF + nNG$$

donde

NF = número total de evaluación de funciones

NG = número total de evaluación de gradientes

n = número de variables

Además se incluyen los siguientes parámetros:

NR = número máximo de restricciones con que resuelve el Algoritmo

$iter$ = número de iteraciones

Tabla1

problema	n	iter	NR	NF	NG	NT
.....1.....	2	18	.2..	198	162	522
.....2....	3	26	14	286	234	988
.....3....	3	43	99	473	387	1634
.....4...	3	18	21	198	162	684

Observación

1) Debido a la falta de memoria del computador, se debió fijar tolerancias que permiten error de truncamiento no muy apropiado para alcanzar una solución mas precisa.

2) Puede observarse que para un mismo problema no se obtiene exactamente los mismos resultados partiendo de dos puntos diferentes, la explicación de esto quizás se deba al hecho de que por la naturaleza de ser el Algoritmo Interno un algoritmo de puntos interiores; Continuamente se utilizó un algoritmo Fase-I dado por el problema de programación no lineal:

$$\begin{aligned} & \text{Min } w/w \in R \\ & \text{s.a } \phi(x, u) - w \leq 0 \end{aligned}$$

Dicho problema no considera la función objetivo, lo cual hace que muchas veces los puntos no viables entren en la región viable con un crecimiento exagerado de función objetivo, de ahí la explicación de la diferencia obtenida.

5. Conclusiones

En este artículo hemos desarrollado un Algoritmo Cuasi-Newton de puntos interiores con el uso de aproximaciones consistentes para Programación Semi infinita, introduciendo una función de optimalidad que resulta natural en general para métodos Cuasi-Newton. Adicionalmente incorporamos una técnica de filtrado de restricciones bajo una estrategia diagonalizante que permitió disminuir drásticamente el número de restricciones y hacer mas eficiente el Algoritmo. Finalmente la robustez del Algoritmo Interno, unida a sus propiedades de convergencia superlineal (Ref. [8]) han permitido en el esquema de aproximaciones consistentes conservar dicha tasa de convergencia en cada uno de los de los problemas de la sucesión generada ; como consecuencia de ello se presenta altamente competitivo en relación a los mejores algoritmos testados en (Ref. [27]).

Una futura investigación es extender el uso de aproximaciones consistentes para métodos

Cuasi-Newton en general.

6. Referencias

- [1] H. Attouch, *Variational Convergence for Functions and Operators* Pitman, London, (1984).
- [2] J.C. Dunn, *Diagonally modified conditional Gradient methods for input constrained Optimal Control Problems*. Siam J. Control and Optimization, Vol 24, No 6 (1986).
- [3] K. Glashoff and Sven Ake Gustafson, *Linear Optimization and approximation*. Springer Verlag (1983).
- [4] C. Gonzaga and E. Polak, *On constraint dropping scheme and optimality functions for a class of outer approximations algorithms*. Siam J. Control and Optimizations, Vol 17, No 4 (1979).
- [5] C. Gonzaga, E. Polak and Trahan, *An improved algorithm for Optimization Problems with functional Inequality constraints*. IEEE Trans. Automat. control AC-25 (1980).
- [6] S. A. Gustafson, *A tree phase algorithm for Semi Infinite Programs*. Semi Infinite Programming and applications. Lectures notes in Economics and Mathematical Systems 215. Springer Verlag (1983).
- [7] L. He and E. Polak, *Effective diagonalization strategies for the solution of a class of Optimal Design Problems*. IEEE Trans. on automatic control Vol 35, No 3 (1990) J. Herskovits An interior point technique for Nonlinear Optimization. INRIA, modulef proj B.P.105, 78153 Le Chesnay Cedex (1993).
- [8] R. Hettich,
 - a) *A comparison of some numerical methods for Semi Infinite Programming*. *Semi Infinite Programming*, proceeding of a workshop Springer Verlag (1979).
 - b) *A review of numerical methods for Semi Infinite Optimization*. *Semi Infinite Programming and application proceeding of an international. Symposium*. Springer Verlag (1983).
 - c) *An implementation of a discretization method for Semi Infinite Programming*. *Mathematical Programming* 34 (1986).

- [9] R. Hettich and G. Gramlich, *A note on a implementation of a method for quadratic Semi Infinite Programming*. Mathematical programming. 46 (1990).
- [10] R. Hettich and W.van Honsted, *On quadratically convergent methods for Semi Infinite Programming*. In [9b].
- [11] R. Hettich and Jongen, *Semi Infinite Programming, condition of optimality and applications. Optimization techniques, part 2*. Lectures notes in control and information sciences. Springer Verlag 7 (1978).
- [12] R. Hettich and K.O Kortanek, *Semi Infinite Programming*. Theory methods and applications. Siam Review, Vol 35, No3 (1993).
- [13] L. He and Polak, *Effective diagonalization strategies for the solution of a class of Optimal Design Problems*. IEEE Trans. on automatic control Vol 35, No3 (1990).
- [14] W. van Honstede, *An approximation method for Semi Infinite Problems*. In [9b].
- [15] H. Hu, *A one phase algorithm for Semi Infinite Linear Programming*. Mathematical Programming 46 (1990).
- [16] H.T. Jongen, F. Twilt and G.W Weber, *Semi Infinite Optimization struture and stability of the feasible set*. JOTA Vol 72, No 3 (1992).
- [17] R.Klessig and E.Polak, *An adaptative precision Gradient method for Optimal Control*. Siam J. Control, Vol 11, No 1 (1973).
- [18] D.Q Mayne and E. Polak, *A Quadratic convergent algorithm for solving Infinite dimensional Inequalities*. Appl. Matematics and Optimization (1982).
- [19] M. Minoux, *Mathematical Programming*. Theory and Algorithms. Ed. John Wiley and Sons (1986).
- [20] E. Panier and A. Tits, *A globally convergent algorithm with adaptively refined discretization for Semi Infinite Optimization Problems arising in Engineering Design*. System Research Center and Electrical Engineering department. Univers. of Maryland. (1988).
- [21] E. Polak,

- a) *Computational methods in Optimization*. Academic Press -New York (1971).
 - b) *On the Mathematical foundations of Nondifferentiable Optimization in Engineering Design*. *Siam Review*, Vol 29, No1 (1987).
 - c) *On the use of consistent approximations in the solution of Semi Infinite Optimization and Optimal Control Problems*. *Mathematical Programming* 62, No 2 (1993).
- [22] E. Polak and L. He,
- a) *Rate preserving discretization strategies for Semi Infinite Programming and Optimal Control*. *Siam J. Control and Optimization*, Vol 30, No 3 (1992).
 - b) *A unified Phase-I Phase-II method of feasible directions for Semi-Infinite Optimization*. M No UCB/ERL M89/7, University of California, Berkeley (1989).
- [23] E. Polak, D.Q. Mayne, and J. E. Higgins, *On the extension of Newton method to Semi Infinite Minimax problems*. *Siam J. Control and Optimization*, Vol 30, No2 (1992).
- [24] E. Polak and A. Tits, *A recursive quadratic programming algorithm for Semi Infinite Optimization Problems*. *Appl. Math. Optimization* (1982).
- [25] R. Reemtsen, *Discretization methods for the solution of Semi Infinite Programming Problems*. *JOTA* Vol 71, No1 (1991).
- [26] Y. Tanaka, M. Fukushima, T. Ibaraki, *A comparative study of several Semi-Infinite nonlinear Programming Algorithms*. *European Journal of Operational Research* 36 (1988).
- [27] G.A. Watson, *Numerical experiments with globally convergent methods for Semi Infinite Programming Problems*. In [9b]
- [28] G.A. Watson and I.D. Coope, *A projected Lagrangian algorithm for Semi Infinite Programming*. *Mathematical Programming* 32 (1985).

Received : February 24, 1997.

C. Nicolas Baracatt

Instituto de Matemáticas
Universidad Austral de Chile
Casilla 567
Valdivia
Chile

and

J. Heskovits N.

Departamento de Ingeniería Mecánica
Universidad Federal de Río de Janeiro
Caixa Postal 56018
22290 Río de Janeiro, R. J.
Río de Janeiro
Brasil