



MÉTODOS COMPUTACIONALES DE OPTIMIZACIÓN
SIN DERIVADAS PARA MINIMIZACIÓN
CON RESTRICCIONES

TESIS DOCTORAL

MARÍA BELÉN AROUXÉT

DIRECTOR: DR. ELVIO A. PILOTTA

CODIRECTOR: LIC. NÉLIDA ECHEBEST

UNIVERSIDAD NACIONAL DE LA PLATA

FACULTAD DE CIENCIAS EXACTAS

DEPARTAMENTO DE MATEMÁTICA

2013

Índice general

Capítulo 1. Introducción	1
1.1. Estado del arte	3
Capítulo 2. Métodos de región de confianza basados en modelos de interpolación	7
2.1. Ideas principales de métodos de región de confianza	7
2.2. Interpolación polinomial lineal o cuadrática	15
2.3. Métodos de optimización sin derivadas	21
Capítulo 3. Método de restricciones activas en el Método de Powell	31
3.1. Conjuntos activos para la resolución del subproblema cuadrático en una caja	31
3.2. Detalles de implementación	34
3.3. Experimentación numérica	35
Capítulo 4. Restauración Inexacta para optimización sin derivadas	47
4.1. Método de Restauración Inexacta (IR)	49
4.2. Hipótesis generales y resultados preliminares para optimización sin derivadas	52
4.3. Algoritmo de Restauración Inexacta sin derivadas	61
4.4. Convergencia a la factibilidad	69
4.5. Convergencia a la optimalidad	70
4.6. Detalles de implementación	82
4.7. Experimentación numérica	84
Capítulo 5. Conclusiones y perspectivas	91
Bibliografía	95

Capítulo 1

Introducción

Consideraremos problemas de optimización con restricciones los cuales serán resueltos sin hacer uso de las derivadas de la funciones involucradas. Formalmente,

$$(1.1) \quad \text{minimizar } f(x) \text{ s.a. } x \in \Omega, C(x) = 0,$$

donde $\Omega = \{x \in \mathbb{R}^n : L \leq x \leq U\}$, con L y U vectores fijos en \mathbb{R}^n , y $f : \mathbb{R}^n \rightarrow \mathbb{R}$ y $C : \mathbb{R}^n \rightarrow \mathbb{R}^m$ son funciones continuamente diferenciables.

Observar que si las restricciones $C(x)$, lineales o no lineales fueran de desigualdad, podríamos transformar el problema a uno de restricciones de igualdad agregando variables de holgura.

Nos restringimos a problemas con una cantidad moderada de variables y asumimos que el costo de evaluar la función objetivo y las restricciones es mayor que el álgebra lineal requerida en la iteración de optimización.

Supondremos que los problemas son diferenciables, pero sus derivadas no se pueden calcular ya sea porque el valor $f(x)$ (o $C(x)$), dado un vector x , es el resultado de una medición física, química o más comunmente, porque es el resultado de un cálculo computacional posiblemente muy largo y complejo, para el cual el código fuente no está disponible o no es modificable.

En la literatura se pueden encontrar varios métodos para resolver problemas diferenciables sin calcular derivadas tales como diferencias finitas, búsqueda por patrones y métodos de región de confianza basados en modelos de interpolación, entre otros. Debido al costo computacional, correspondiente a la gran cantidad de evaluaciones necesarias de la función objetivo, estos métodos están restringidos a resolver problemas de pocas variables.

Los métodos de región de confianza basados en interpolación polinomial forman un modelo lineal o cuadrático de la función objetivo y utilizan la región de confianza para lograr la convergencia.

En el caso en que la función objetivo sea diferenciable, la idea de usar métodos de región de confianza basados en interpolación polinomial resulta muy atractivo ya que se puede reemplazar una evaluación de la función objetivo, la cual resulta muy costosa, por una evaluación del modelo

aproximante, cuyo costo es mucho menor. En especial, para problemas complejos, se logra un progreso considerable obteniendo mejores soluciones a un costo moderado.

En el Capítulo 2 describiremos los métodos de región de confianza basados en modelos de interpolación y, luego, algunos de los métodos más utilizados para optimización sin derivadas que utilizan esta metodología.

En este trabajo presentaremos dos métodos de optimización sin derivadas basados en modelos de interpolación y región de confianza.

El primer algoritmo desarrollado para este trabajo es el algoritmo TRB-Powell [12]. TRB-Powell fue propuesto para resolver un problema de optimización irrestricto o con restricciones de caja sin derivadas basado en el método de Powell, NEWUOA [90, 91], para optimización (sin restricciones) sin derivadas. El método NEWUOA, en cada iteración, construye un modelo de interpolación cuadrática de la función objetivo alrededor del iterado actual y este modelo es minimizado para obtener un nuevo punto de prueba. Todo el proceso está inmerso en un marco de región de confianza usando la norma Euclídea. En nuestro método, TRB-Powell, usamos norma infinito en vez de norma Euclídea y resolvemos el subproblema cuadrático usando una estrategia de conjuntos activos para explorar las caras de la caja. Inicialmente hemos considerado el problema irrestricto y luego extendimos el algoritmo, de manera natural, para la resolución de un problema con restricciones de caja. En el Capítulo 3 presentaremos el Algoritmo TRB-Powell y mostraremos los resultados obtenidos al compararlo con NEWUOA y BOBYQA para problemas irrestrictos y con restricciones en las variables, respectivamente.

El segundo algoritmo es el IR-DFO, el cual se propone para resolver problemas de programación no lineal general sin hacer uso de derivadas y está basado en el método de Restauración Inexacta (IR), el cual fue introducido por Martínez y Pilotta (2000) [77] y posteriormente analizado en el año 2005 [78]. Estos métodos proceden en dos fases y generan una sucesión de puntos, posiblemente no factibles respecto de las restricciones de igualdad, con iteraciones intermedias que consisten en puntos inexactamente restaurados. En nuestro método, todos los cálculos de derivadas del método IR han sido adaptados adecuadamente para resolver el problema sin el uso de éstas. La importancia de este nuevo método sin derivadas, IR-DFO, es que se ha demostrado la convergencia a puntos factibles. En el Capítulo 4 describiremos el Algoritmo IR-DFO y expondremos los resultados de convergencia y experimentación numérica.

Finalmente, en el Capítulo 5 presentaremos algunas conclusiones y daremos perspectivas del trabajo realizado.

1.1. Estado del arte

En la literatura existen varios métodos para resolver problemas de optimización sin derivadas. Ver [35]. Un grupo lo conforman los métodos que calculan las derivadas de la función objetivo, ya sea por aproximación como por ejemplo usando diferencias finitas (ver Gill et al [48, 49], Dennis y Schnabel [64] y Nocedal y Wright [83]), o por procedimientos de diferenciación automática (ver Griewank y Corliss [59], Gilbert [47] y Griewank [57, 58]) o calculando el gradiente basándose en la resolución de ecuaciones diferenciales, si el problema depende de la solución de un sistema de ecuaciones diferenciales (ver [26, 50, 99]). En general, dado el costo de evaluar la función objetivo, evaluar su Hessiano o gradiente por diferencias finitas o diferenciación automática puede ser mucho más costoso pero, cuando los recursos están disponibles o las evaluaciones no son irracionalmente caras, este tipo de métodos resulta muy útil y eficiente.

Otro grupo de métodos de optimización que no utilizan derivadas lo forman los métodos de búsqueda directa o métodos de búsqueda por patrones. Éstos están basados en explorar el espacio de variables evaluando la función objetivo en puntos muestrales dados por un patrón geométrico predeterminado. Es el caso de los métodos en donde las muestras son guiadas por un conjunto de direcciones adecuadas [41, 98] y en base a simplexes y operaciones sobre los mismos, tales como el algoritmo de Nelder-Mead [82]. Estos métodos no explotan la suavidad de la función objetivo y, por lo tanto, requieren un gran número de evaluaciones funcionales. Son útiles para problemas no suaves. Un amplio estudio de estos métodos ha sido realizado por Kolda, Lewis y Torczon [68]. En esta clase se encuentra el método GSS [69, 72] el cual es utilizado cuando las restricciones del problema (1.1) son lineales. El método GSS abarca muchos algoritmos, incluyendo Generalized Pattern Search (Búsqueda por patrón generalizado) [72]. Otra generalización de los métodos de búsqueda por patrones es el reciente desarrollo de la adaptación de la malla en los métodos de búsqueda directa (ver Audet y Dennis [14, 15], Abramson y Audet [3], Audet y Orban [16], Audet, Bécharde y Le Digabel [13], Abramson, Audet, Dennis y Le Digabel [4]). Más aún, dentro de la clase de los métodos de búsqueda directa se encuentran unos métodos híbridos que han sido desarrollados e implementados para mejorar la eficiencia de los mismos. Algunos de ellos son el SID-PSM de Custodio, Rocha y Vicente [37, 39], el cual es una implementación en MATLAB para problemas con restricciones lineales de desigualdad, el NOMADm de Abramson [2] y el APPSPACK [56, 61, 63, 67, 70, 84].

Un tercer grupo de métodos se basa en el modelado de la función objetivo mediante interpolación multivariada en combinación con técnicas de región de confianza. Estos métodos se introdujeron por Winfield [100, 101]. Un modelo polinomial es construido con el fin de interpolar la función objetivo en los puntos donde los valores funcionales son conocidos. El modelo es entonces minimizado en una región de confianza y un nuevo punto es calculado. La función objetivo se evalúa en este nuevo punto y por lo tanto, posiblemente, se amplía el conjunto de interpolación. En este nuevo punto se verifica si la función objetivo ha disminuido su valor funcional y todo el proceso se repite hasta que se logre la convergencia. Así, la geometría de los puntos de interpolación de ajuste y la minimización del modelo son las claves para un buen desempeño de los algoritmos.

Actualmente existen variadas implementaciones, de algoritmos basados en modelos de interpolación, para problemas de optimización irrestrictos y con restricciones de caja sin derivadas tales como Powell [86, 88, 89], Wild et al [95], Fasano, Nocedal y Morales [44], Scheinberg y Toint [97], Conn, Scheinberg y Toint [31, 32] y Gratton, Toint y Tröltzsch [55]. La mayoría de estos métodos se diferencian en la forma de adaptar el conjunto de interpolación. Para el caso de problemas con restricciones lineales existe un nuevo método de Powell [94].

Algunos autores han desarrollado métodos para tomar ventaja de la estructura especial del problema con el fin de resolver problemas más grandes (ver Colson y Toint [28, 29], Griewank y Toint [60]).

Los métodos basados en interpolación polinomial son ampliamente usados en la práctica. Algunos de ellos son el DFO de Conn, Scheinberg y Toint [31, 32, 33], la implementaciones de Powell UOBYQA [87], NEWUOA [90, 91] y BOBYQA [92], el software Wedge de Marazzi y Nocedal [74] y el código creado por Berghen y Bersini [20], llamado CONDOR, el cual incluye una versión en paralelo basada en NEWUOA.

El problema principal de estos métodos es que, si bien tienen buen desempeño en la práctica, no se ha obtenido buenos resultados teóricos de convergencia. Es por esto que varios grupos de investigación se han dedicado a buscar resultados de convergencia. En el caso de problemas irrestrictos existen varios estudios tales como los realizados por Scheinberg y Toint [97], Powell [93] y Nocedal y Marazzi [74] (en el caso de interpolación lineal).

Con el fin de demostrar convergencia teórica, además de buenos resultados en la práctica, es que surge un cuarto grupo de métodos para resolver el problema (1.1) el cual contiene a los métodos que son híbridos como por ejemplo aquéllos que, sobre el esquema de métodos que han

demostrado tener buenas propiedades de convergencia en el caso con derivadas, utilizan cualesquiera de los métodos ya reconocidos sin derivadas para resolver subproblemas internos. Estos métodos son transformados a métodos sin derivadas con la ventaja que heredan las propiedades de convergencia. Por ejemplo, Restauración Inexacta de Bueno, Friedlander, Martínez y Sobral [27], el cual es parcialmente sin derivadas ya que no calculan derivadas de la función objetivo pero calculan las derivadas de las restricciones, Lagrangiano Aumentado de Diniz-Ehrhardt, Martínez y Pedroso [42] o el algoritmo de Conejo, Karas, Ribeiro, Pedroso y Sachine [30]. En [27] los autores proponen resolver la fase de optimalidad de un algoritmo de Restauración Inexacta mediante el algoritmo GSS [69, 72] evitando así el cálculo de las derivadas de la función objetivo. En [42] cada subproblema se puede resolver utilizando cualquier método adecuado sin derivadas. El algoritmo de región de confianza presentado en [30] resuelve un problema de minimización de una función en un dominio convexo para el cual, en cada iteración, genera un modelo cuadrático y busca minimizadores del mismo en una región de confianza intersecada por el dominio convexo, por cualquier método adecuado pero satisfaciendo ciertas hipótesis. Bajo dichas hipótesis, los autores demuestran convergencia global del algoritmo.

Métodos de región de confianza basados en modelos de interpolación

Existen muchos artículos de optimización sin derivadas en los que se han considerado estrategias de región de confianza con interpolación polinomial de la función objetivo [31, 32, 33, 89, 90, 91]. Los métodos propuestos para resolver problemas de optimización irrestricta o restringida con cotas en las variables sirven de base para resolver problemas más generales de optimización no lineal.

En este capítulo daremos una breve descripción de los conceptos básicos utilizados en los métodos de región de confianza con interpolación polinomial lineal o cuadrática. También describimos los conceptos de interpolación polinomial lineal o cuadrática de una función diferenciable.

2.1. Ideas principales de métodos de región de confianza

En el marco de un método iterativo para resolver el problema

$$\min_{x \in \mathbb{R}^n} f(x),$$

donde f es una función a valores reales con $\nabla f(x)$ y $\nabla^2 f(x)$ continuos, en cada iteración, en lugar de minimizar la función f , se minimiza un modelo aproximado de ella.

En los métodos de región de confianza para resolver problemas de optimización, la información de la función objetivo se utiliza para formar un modelo m_k el cual, cerca del punto actual x_k , tenga en lo posible un comportamiento similar al de la función objetivo. Como el modelo puede no ser una buena aproximación de la función considerada, cuando se toman puntos x alejados de x_k , se debe restringir la búsqueda del minimizador de m_k a puntos que están en un entorno de x_k . Tal región restringida se denomina “región de confianza”.

En cada iteración de esta metodología, se resuelve el subproblema

$$(2.1) \quad \begin{aligned} \min_s m_k(x_k + s) &= c_k + g_k^t s + \frac{1}{2} s^t G_k s \\ \text{sujeto a } \|s\| &\leq \Delta_k, \end{aligned}$$

donde $c_k \in \mathbb{R}$, $g_k \in \mathbb{R}^n$ y $G_k \in \mathbb{R}^{n \times n}$ es una matriz simétrica.

Usualmente las regiones de confianza son esferas definidas por la norma 2 centradas en x_k , con radio Δ_k llamado radio de la región de confianza ($\|s\|_2 \leq \Delta_k$). También se usan regiones de confianza elípticas o en forma de cajas usando la norma infinito ($\|s\|_\infty \leq \Delta_k$).

Cuando se utilizan derivadas, si m_k es la aproximación de Taylor de primer orden entonces $c_k = f(x_k)$, $g_k = \nabla f(x_k)$ y G_k es la matriz nula. Luego, si m_k es la aproximación de Taylor de segundo orden, $c_k = f(x_k)$, $g_k = \nabla f(x_k)$ y $G_k = \nabla^2 f(x_k)$.

La solución exacta del modelo m_k es más o menos difícil según sea el modelo cuadrático o lineal y según como sea la región de confianza. Si la región de confianza está definida por la norma 2, y si el modelo es lineal, la solución se encuentra en el borde de la región y en la dirección de $-g_k$. Con igual tipo de región, si el modelo es cuadrático la solución, s^* , cumple:

TEOREMA 1. El vector s^* es una solución global del problema de región de confianza

$$(2.2) \quad \begin{aligned} \min_{s \in \mathbb{R}^n} m(s) &= c + g^t s + \frac{1}{2} s^t G s \\ \text{sujeto a } \|s\|_2 &\leq \Delta, \end{aligned}$$

si y sólo si s^* es factible y existe un escalar $\lambda \geq 0$ tal que se satisfacen las siguientes condiciones

$$(2.3) \quad (G + \lambda I)s^* = -g$$

$$(2.4) \quad \lambda(\Delta - \|s^*\|_2) = 0$$

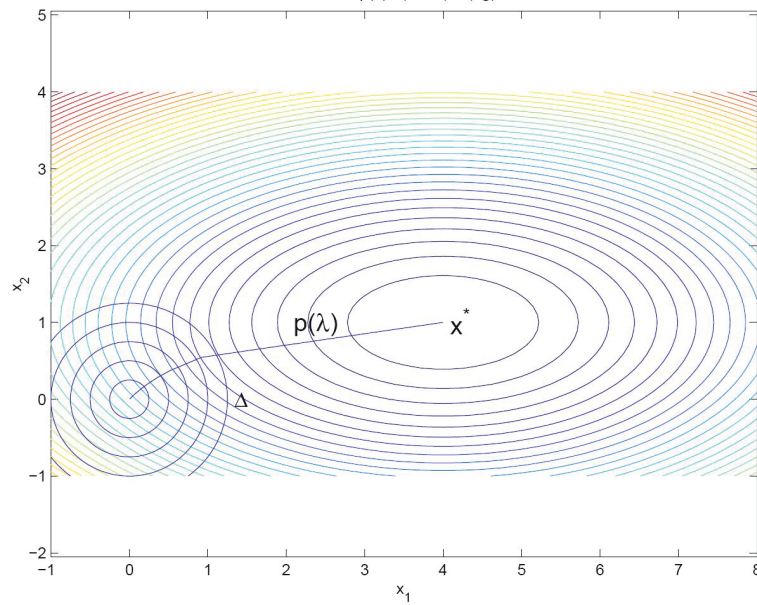
$$(2.5) \quad G + \lambda I \quad \text{es semidefinida positiva.}$$

Observar que, si $\lambda \neq 0$, la solución s^* es colineal con $-\nabla m(s^*)$ pues, de (2.3), se deduce que

$$\lambda s^* = -(g + Gs^*) = -\nabla m(s^*),$$

y normal a las curvas de nivel. En este caso, $\|s^*\|_2 = \Delta$ o sea, s^* se encuentra en el borde de la región, ya sea porque el mínimo de la cuadrática es exterior a la región o porque hay curvaturas nulas o negativas. Cuando la matriz G es definida positiva, la solución s^* está sobre una curva $p(\lambda) = (G + \lambda I)^{-1}(-g)$, tal que para $\|p(\lambda)\| = \Delta$ da la solución en la esfera de radio Δ , si el óptimo global es exterior a la esfera de radio Δ . Cuando el mínimo de la función cuadrática es interior a la esfera de radio Δ , la solución del subproblema es $s^* = (G)^{-1}(-g)$, la dirección de Newton, correspondiente a $\lambda = 0$. En la Figura 1 se ilustra lo anterior.

Para definir el iterado x_{k+1} , un punto posible es un punto $x_k + s_k^*$, donde s_k^* es el mínimo del modelo m_k , en la región de confianza o una aproximación del mismo.

FIGURA 1. Curva $p(\lambda) = (G + \lambda I)^{-1}(-g)$

Así, en x_k se considera el subproblema cuadrático

$$(2.6) \quad \min_{s \in \Omega_k} m_k(x_k + s),$$

donde $\Omega_k = \{s : s \in \mathbb{R}^n : \|s\| \leq \Delta_k\}$.

En particular, cuando se usan derivadas, si la región de confianza es “pequeña” y el modelo aproximado es “bueno”, al resolver el problema (2.6) podríamos perder la oportunidad de acercarnos lo más posible al mínimo real de la función objetivo. Si la región es “grande” y la aproximación “no es muy buena”, el mínimo del modelo puede estar lejos del mínimo de la función objetivo en la región de confianza (podríamos estar en el caso en el cual el modelo no es una buena aproximación de f en toda la región). Esto motiva a que los radios de las regiones de confianza puedan modificarse al cambiar de iteración, dependiendo del tamaño de la región y del análisis comparativo del comportamiento del decrecimiento de m_k y de la función objetivo.

Para elegir el radio de confianza de la región en la iteración $k + 1$, dado el paso s_k^* obtenido al resolver el problema (2.6) en la iteración anterior, se define

$$\rho_k = \frac{f(x_k) - f(x_k + s_k^*)}{m_k(x_k) - m_k(x_k + s_k^*)}.$$

El numerador indica la variación de la función objetivo, se denomina reducción actual, y el denominador mide la variación del modelo y recibe el nombre de predictor. Este último es siempre no negativo pues proviene de la minimización, exacta o aproximada, del modelo m_k

en la región de confianza. Cuando el punto x_k es un punto estacionario del problema original, $s_k^* = 0$, es un caso particular.

Si ρ_k es negativo, la reducción actual de f es negativa, es decir $f(x_k) < f(x_k + s_k^*)$, por tanto la función objetivo crece. Esto motiva el rechazo del paso s_k^* , la definición de $x_{k+1} = x_k$, y la reducción del radio Δ_{k+1} cuando se tienen modelos definidos mediante las derivadas de la función original.

Por otro lado, si ρ_k es cercano a 1, el modelo se comporta de manera similar a la función en ese punto. Se propone entonces para la próxima iteración aumentar el radio. Si ρ_k es no negativo pero no cercano a uno, en general no se altera el tamaño de la región y si ρ_k es no negativo pero cercano a cero, se disminuye el tamaño de la región.

Un esquema general del método de Región de Confianza (RC) es el que se enuncia a continuación [83].

ALGORITMO 1. Algoritmo (RC)

Dados $x_0, \epsilon > 0, 0 < \eta_1 < \eta_2 < 1, \gamma_1 \in (0, 1), \gamma_2 > 1, \Delta_c > 0, \Delta_0 \in (0, \Delta_c)$.

$k \leftarrow 0$

Paso Iterativo: Dados $x_k, f(x_k), g_k, G_k, \Delta_k$

- Test para finalizar.
- (1) Obtener s_k resolviendo aproximadamente el problema (2.6)
- Calcular ρ_k
 - Si $\rho_k < \eta_1$ definir $\Delta_k = \gamma_1 \Delta_k$ y volver a (1)
 - Si $\rho_k \geq \eta_1$ definir $x_{k+1} = x_k + s_k$
 - Si $\rho_k \geq \eta_2$ definir $\Delta_{k+1} = \gamma_2 \Delta_k$
 - Si no $\Delta_{k+1} = \Delta_k$.
- $k \leftarrow k + 1$

Fin

Es ilustrativo describir algunos resultados sobre esta metodología cuando se usan derivadas, que sirven de base para justificar que se pueden obtener buenos resultados de convergencia resolviendo los subproblemas en forma aproximada. Esos resultados también ayudan al desarrollo de resultados teóricos y prácticos cuando las derivadas de la función objetivo no están disponibles.

En particular para nuestro trabajo, cuando las derivadas del problema no están disponibles, motivan a considerar modelos m_k tal que su gradiente g_k logre una buena aproximación asintótica al $\nabla f(x_k)$.

2.1.1. Punto de Cauchy y propiedades de convergencia. En esta sección se analizan las condiciones que debe cumplir el paso s_k para obtener buenas propiedades de convergencia del método de Región de Confianza, cuando se usan derivadas.

Se dice que un algoritmo iterativo es globalmente convergente si

$$\|g_k\| \rightarrow 0 \text{ cuando } k \rightarrow \infty.$$

Se ve que, para demostrar la convergencia global del método de Región de Confianza, es suficiente encontrar una solución aproximada s_k del problema (2.6) que proporcione una reducción “suficiente” en el modelo m_k .

La reducción requerida puede ser cuantificada en términos de un punto particular que se denomina punto de Cauchy y será denotado por s_k^c .

Cuando la región de confianza se define usando la norma 2, el punto de Cauchy es el mínimo del modelo $m_k(x_k + s) = c_k + g_k^t s + s^T G_k s$, restringido a la dirección de $-g_k$, dentro de la región de confianza.

El mínimo buscado s_k^c es de la forma $s_k^c = \tau (-g_k)$ con τ tal que $\|s_k^c\| \leq \Delta_k$.

Si $g_k^t G_k g_k \leq 0$, como m_k decrece monótonamente a lo largo de $-g_k$, entonces τ satisface $\tau = \frac{\Delta_k}{\|g_k\|}$ y $\|s_k^c\| = \Delta_k$.

Si $g_k^t G_k g_k > 0$, es decir G_k es definida positiva a lo largo de $-g_k$, el mínimo en esa dirección puede estar dentro o fuera de la región de confianza.

Si el mínimo estuviese en el exterior, el punto buscado s_k^c está en la frontera de la región y como en el caso anterior $\tau = \frac{\Delta_k}{\|g_k\|}$.

Si el mínimo de m_k en la dirección de $-g_k$, que se conoce es de la forma $\frac{\|g_k\|^2}{g_k^t G_k g_k} (-g_k)$, estuviese dentro de la región de confianza, se obtiene $\tau = \frac{\|g_k\|^2}{g_k^t G_k g_k}$.

Así, el paso de Cauchy puede escribirse de la siguiente manera $s_k^c = \sigma \frac{\Delta_k}{\|g_k\|} (-g_k)$ donde

$$\sigma = \begin{cases} 1 & \text{si } g_k^t G_k g_k \leq 0 \\ \min\{1, \frac{\|g_k\|^3}{\Delta_k g_k^t G_k g_k}\} & \text{si } g_k^t G_k g_k > 0 \end{cases}$$

El punto de Cauchy $x_k + s_k^c$ cumple el siguiente resultado, del cual se desprende la convergencia global del método de Región de Confianza [83].

PROPOSICIÓN 2. El punto de Cauchy satisface

$$m_k(x_k) - m_k(x_k + s_k^c) \geq \frac{1}{2} \|g_k\| \min \left\{ \Delta_k, \frac{\|g_k\|}{\|G_k\|} \right\}.$$

Como consecuencia inmediata de la proposición anterior se tiene el siguiente teorema.

TEOREMA 3. Sea $x_k + s_k$ cualquier punto tal que $\|s_k\|_2 \leq \Delta_k$, cumpliendo que

$$m_k(x_k) - m_k(x_k + s_k) \geq c [m_k(x_k) - m_k(x_k + s_k^c)],$$

siendo $c > 0$. Entonces s_k satisface

$$m_k(x_k) - m_k(x_k + s_k) \geq \frac{c}{2} \|g_k\| \min \left\{ \Delta_k, \frac{\|g_k\|}{\|G_k\|} \right\}.$$

En particular, la solución exacta del problema (2.6) satisface la desigualdad anterior con $c = 1$.

Una demostración de los resultados anteriores puede encontrarse en J. Nocedal, S. Wright [83].

Para demostrar la convergencia global del método de Región de Confianza, usando derivadas, se asumen las siguientes hipótesis generales:

- i)* f es una función a valores reales con $\nabla f(x)$ y $\nabla^2 f(x)$ continuos;
- ii)* el conjunto de nivel $A = \{x : f(x) \leq f(x_0)\}$ es acotado y f está acotada inferiormente sobre A ;
- iii)* las matrices G_k están uniformemente acotadas en norma, es decir $\|G_k\| \leq M \forall k \geq 0$.

También se le permitirá a la solución aproximada s_k del problema (2.6) que exceda la frontera de la región de confianza pero asegurando que quede dentro de un múltiplo fijo del radio de confianza en cada iteración, es decir $\|s_k\| \leq \gamma \Delta_k$ para alguna constante $\gamma \geq 1$.

Se obtiene entonces el siguiente resultado:

TEOREMA 4. Sea $\eta_1 \in (0, \frac{1}{4})$ en el Algoritmo **RC** de región de confianza. Supongamos que se verifican *i)*, *ii)* y *iii)* y que las soluciones aproximadas s_k del problema (2.6) satisfacen las siguientes desigualdades

$$(2.7) \quad m_k(x_k) - m_k(x_k + s_k) \geq c \|g_k\| \min \left\{ \Delta_k, \frac{\|g_k\|}{\|G_k\|} \right\}$$

$$(2.8) \quad \|s_k\| \leq \gamma \Delta_k,$$

donde $c > 0$ y $\gamma \geq 1$. Entonces $\|g_k\| \rightarrow 0$ cuando $k \rightarrow \infty$.

Se desprende de este teorema, cuya demostración puede encontrarse en [83], que no es necesario resolver de manera exacta el problema (2.6) para obtener la convergencia global del método. Alcanza con encontrar soluciones aproximadas que verifiquen las condiciones (2.7) y (2.8).

Como el punto de Cauchy satisface la condición (2.7) con $c = \frac{1}{2}$, los puntos s_k en Ω_k , tales que su reducción en el modelo cuadrático sea mayor que la reducción obtenida en el punto de Cauchy, también verifican la misma desigualdad. Por lo tanto, aceptando tales puntos como soluciones aproximadas del problema (2.6), se garantiza la convergencia global del método de Región de Confianza.

En la siguiente sección se describirá un método que puede ser utilizado para resolver en forma aproximada el problema (2.6).

2.1.2. Método de Steihaug. Este es un método que está basado en el algoritmo de Gradientes Conjugados [83] y su ventaja más importante es que, en la solución s_k^* , el modelo m_k obtiene una reducción mayor o igual al del punto de Cauchy. Esto demuestra que el método de Región de Confianza usando este algoritmo de Steihaug para resolver el problema (2.6) es globalmente convergente.

El algoritmo de Steihaug aplica el método de Gradientes Conjugados a partir del punto x_k aunque el modelo no sea convexo. En este caso especial, si obtiene una dirección de curvatura negativa, puede quedarse con un punto de la frontera de la región que se encuentra en tal dirección.

Suprimimos los índices k para simplificar la notación, recordando que, en cada iteración k del método de Región de Confianza, el problema (2.6) se resuelve en forma aproximada usando el método de Gradientes Conjugados. Un algoritmo estándar es el siguiente:

ALGORITMO 2. Algoritmo (ST)

Dados $\epsilon_{gc} > 0$, $\Delta > 0$. Sean $p_0 = 0$, $r_0 = g$, $d_0 = -g$.

$j \leftarrow 0$

Iteración j -ésima: Para obtener d_{j+1} se hacen los siguientes pasos:

Paso 1: Si $d_j^t G d_j \leq 0$,

buscar τ tal que $p = p_j + \tau d_j$ y $\|p\| = \Delta$;

definir $s^* = p$.

Paso 2: Calcular $\alpha_j = \|r_j\|^2 / (d_j^t G d_j)$.

Definir $p_{j+1} = p_j + \alpha_j d_j$.

Si $\|p_{j+1}\| > \Delta$,

buscar τ tal que $p = p_j + \tau d_j$ y $\|p\| = \Delta$;

y definir $s^* = p$.

Paso 3: Calcular $r_{j+1} = r_j + \alpha_j G d_j$,

Si $\|r_{j+1}\| < \epsilon_{gc} \|r_0\|$ definir $s^* = p_{j+1}$.

Paso 4: Calcular $\beta_j = \|r_{j+1}\|^2 / \|r_j\|^2$.

Definir $d_{j+1} = -r_{j+1} + \beta_j d_j$.

$j \leftarrow j + 1$

Fin

OBSERVACIÓN 2.1. El punto p_1 definido en la primera iteración es exactamente el punto de Cauchy. Por tanto, como cada iteración del método de Gradientes Conjugados produce una reducción en el modelo m , el punto s^* obtenido por este algoritmo cumplirá la condición (2.7) necesaria para la convergencia global.

Otra propiedad importante de este método es que el iterado p_j es más grande en norma que el iterado p_{j-1} . Su principal consecuencia es que se puede finalizar la iteración tan pronto como la frontera es alcanzada, pues los iterados posteriores quedarán afuera de la región y los anteriores dan una reducción menor en el modelo. Se enuncia este resultado en el siguiente teorema (la demostración puede hallarse en [83]).

TEOREMA 5. La sucesión de puntos generados por el algoritmo de Steihaug (ST) satisface

$$0 = \|p_0\| < \dots < \|p_j\| < \|p_{j+1}\| < \dots < \|p^*\| \leq \Delta.$$

OBSERVACIÓN 2.2. En el caso en que no se puedan calcular las derivadas de la función objetivo se consideran modelos aproximados [35], donde $g_k \neq \nabla f(x_k)$ y $G_k \neq \nabla^2 f(x_k)$, y, en ausencia de interpolación, $c_k \neq f(x_k)$. En este caso sin derivadas es necesario asegurar que el radio de la región de confianza sea acotado inferiormente, esto es, que no sea muy pequeño.

Los modelos basados en interpolación polinomial no necesariamente mejoran la aproximación a la función objetivo a medida que se reduce el radio de la región de confianza. Esta aclaración se hace porque, cuando los modelos se basan en aproximaciones con polinomios de Taylor, el modelo representa mejor a la función a medida que el radio de confianza se hace más pequeño. El radio de la región de confianza se asegura que está acotado inferiormente, siempre que no se

haya alcanzado un punto estacionario.

Así cuando no se dispone de las derivadas, es necesario asegurar que el radio de la región de confianza se reduce sólo si se asegura que la reducción de la función objetivo no es satisfactoria debido a que el radio es muy grande y no porque el modelo no sea suficientemente bueno.

Con esta aclaración, se puede demostrar que cuando el entorno se hace pequeño, la predicción se hace buena y luego el radio de la región de confianza permanece acotado inferiormente con resultados similares a los obtenidos para el caso con derivadas. Ver para más detalles [35].

Notar que, en el caso en que se tengan $p = n + 1$ puntos de interpolación, el modelo a construir es un modelo lineal tomándose $G_k \equiv 0$ y, si se usa la norma 2, la solución se encuentra en el borde de la región y en la dirección de $-g_k$.

También en el caso de norma 2, si $p > n + 1$ entonces la solución s^* para el problema de región de confianza (2.1) satisface

$$(G_k + \lambda I)s^* = -g_k$$

para algún $\lambda \geq 0$ cumpliendo propiedades adicionales que se explicitaron en el Teorema 1.

2.2. Interpolación polinomial lineal o cuadrática

Para definir el modelo m_k como en (2.1) se necesita obtener el vector g_k y la matriz simétrica G_k . Los coeficientes son determinados de tal manera que el modelo m_k , interpola a la función f en un conjunto de puntos $Y = \{y^1, y^2, \dots, y^p\}$ que contiene al iterado actual x_k

$$(2.9) \quad f(y^i) = m_k(y^i) \quad \text{para todo } y^i \in Y,$$

donde p es un entero dado que es independiente de k , pero las posiciones de los puntos de interpolación son generadas automáticamente y deben satisfacer las siguientes propiedades:

- (1) El modelo cuadrático m_k debe satisfacer las condiciones de interpolación (2.9) cualesquiera sean los valores funcionales en los puntos de interpolación.
- (2) Los puntos de interpolación no deben estar en un mismo hiperplano $\{x : \pi(x) = 0\}$ donde π es un polinomio lineal.

Observar que, según la cantidad de puntos que se tengan para interpolar y la posición de éstos, va a ser la forma del modelo.

La condición (2) implica que $p \geq n+1$. Si se tiene $p = n+1$ puntos, entonces el modelo a utilizarse va a ser lineal, y se obtiene solución única del subproblema, como ya se dijo anteriormente. Si

$p = n + 2$ puntos, se utiliza la poca información disponible de la curvatura de la función objetivo para formar un modelo de interpolación cuadrático con $\nabla^2 m_k = \alpha_k I$, $\alpha_k \in \mathbb{R}$. Esto hace que la cantidad de iteraciones, para resolver el subproblema de región de confianza, sea muy grande ya que la región de confianza es pequeña y por lo tanto el modelo interpolante no se pega muy bien con la función objetivo. En cambio, si $p = 2n + 1$ entonces existe más información sobre la curvatura pues se podría obtener por ejemplo, $\nabla^2 m_k = \text{diag}(\alpha_k^1, \dots, \alpha_k^n)$ con $\alpha_k^i \in \mathbb{R}$ para $i = 1, \dots, n$ y así, en el caso sin derivadas, se tiene la misma cantidad de restricciones que cuando se tienen derivadas.

La condición (1) implica que $p \leq \frac{(n+1)(n+2)}{2}$. De tomarse $p = \frac{(n+1)(n+2)}{2}$ entonces no hay grados de libertad por lo que el modelo se obtiene de las condiciones de interpolación.

Por tanto, se debe tomar $p \in [n + 2, \frac{(n+1)(n+2)}{2}]$ para que haya algún cambio en las derivadas segundas del modelo cuadrático en cada iteración.

En algoritmos iterativos de minimización irrestricta sin derivadas, las aproximaciones cuadráticas a la función objetivo producen buenas estimaciones de las primeras y segundas derivadas. Por tanto, quisiéramos saber cómo obtener buenas aproximaciones cuadráticas interpolando valores funcionales de la función objetivo.

En cada iteración, la función objetivo es evaluada en el punto que minimiza o en un punto aproximado al minimizador del modelo en una región de confianza acotada, y éste valor de f aporta información importante para adaptar el modelo cuadrático, con el agregado de algunos pocos valores extras que se necesitan para evitar degeneración.

La cantidad de puntos de interpolación y sus posiciones pueden ser controlados adecuadamente extrayendo un punto de los actuales para ser reemplazado por uno nuevo.

En cada iteración, se resuelve el problema (2.1) obteniendo un nuevo punto $x_{k+1} = x_k + s_k$ por lo que en la siguiente iteración tendremos el nuevo problema:

$$\begin{aligned} \min_{s \in \mathbb{R}^n} m_{k+1}(x_{k+1} + s) &= c_{k+1} + g_{k+1}^t s + \frac{1}{2} s^t G_{k+1} s \\ \text{s.a. } \|s\| &\leq \Delta_{k+1}. \end{aligned}$$

Luego, al finalizar la k -ésima iteración, se debe definir g_{k+1} y G_{k+1} . Por ello, es necesario decir cómo se define el modelo en la nueva iteración en base al modelo usado en la iteración previa.

Es ilustrativo recordar cómo se trabaja en optimización cuando se conocen las derivadas primeras.

Así, supongamos que se tienen las derivadas primeras de la función objetivo, por lo tanto

$$g_{k+1} = \nabla f_{k+1} = \nabla f(x_{k+1}).$$

Luego, para adaptar la matriz G de una iteración a otra, se pone como condición para la nueva matriz que satisfaga la condición secante, esto es

$$G_{k+1}s_k = z_k,$$

con $z_k = \nabla f_{k+1} - \nabla f_k$. Observar que la condición secante siempre se satisface cuando se cumple $s_k^t z_k > 0$ (pues $s_k^t G_{k+1} s_k > 0$). Tal condición se denomina “condición de curvatura”.

Cuando se satisface la condición de curvatura, el sistema definido por la ecuación secante siempre tiene solución G_{k+1} . De hecho, admite infinitas soluciones ya que hay $n(n+1)/2$ grados de libertad en la matriz simétrica y la ecuación secante sólo representa a n restricciones. Cuando se agrega la condición de positividad para los autovalores de la matriz, representando n restricciones más¹, aún restan $n(n-1)/2$ grados de libertad.

Para determinar G_{k+1} unívocamente, deben imponerse condiciones adicionales sobre todas las matrices simétricas que satisfacen la ecuación secante y que, en algún sentido, herede propiedades y/o esté cerca de G_k . En otras palabras, se resuelve el problema

$$\begin{aligned} \text{mín} \quad & \|G - G_k\| \\ \text{s.a.} \quad & G = G^t \quad \text{y} \quad Gs_k = z_k, \end{aligned}$$

donde s_k y z_k satisfacen la condición de curvatura y G_k es simétrica, y también se puede requerir que sea definida positiva.

Se pueden usar distintas normas de matrices y cada una define un método Quasi-Newton diferente. Powell (ver [64]) propone el uso de la norma Frobenius, en cambio, los métodos BFGS y DFP proponen el uso de la norma Frobenius pesada². En estos métodos, G_{k+1} difiere de G_k en una matriz de rango 2. También se puede adaptar con una matriz simétrica de rango 1 utilizando el método SR1 pero, dada G_k simétrica definida positiva, no se puede garantizar que G_{k+1} sea definida positiva.

Esto sucede cuando se dispone de las primeras derivadas de la función objetivo, pero no es nuestro caso. Sin embargo, se puede observar que, si $p = 2n + 1$ se tiene igual cantidad de

¹los menores principales deben ser definidos positivos

² $\|A\|_W = \|W^{1/2}AW^{1/2}\|_F$ con el peso W elegido tal que $Wy_k = s_k$.

restricciones que en el caso de los modelos Quasi-Newton descriptos cuando se tienen derivadas. Por tanto, se podría pensar en usar alguna estrategia similar que se adapte al caso sin derivadas y que permita adaptar la matriz en cada iteración usando los restantes $n(n-1)/2$ grados de libertad.

2.2.1. Modelos de interpolación. Sea P_n^d el espacio de los polinomios de grado $\leq d$, en \mathbb{R}^n . Una *base de polinomios* $\Phi = \{\phi_1(x), \phi_2(x), \dots, \phi_p(x)\}$ de P_n^d es un conjunto de p polinomios de grado menor o igual a d que generan P_n^d como por ejemplo, la base de monomios, en el caso de interpolación lineal, o las bases de polinomios de Lagrange o Newton, en el caso cuadrático.

Así, todo polinomio $m(x)$ en P_n^d puede escribirse como

$$m(x) = \sum_{j=1}^p \alpha_j \phi_j(x),$$

donde los coeficientes α_j son reales.

Dado un conjunto de interpolación Y , los coeficientes de un modelo $m(x)$ de grado d que interpole a f en tales puntos, se pueden obtener resolviendo:

$$M(\Phi, Y)\alpha = f(Y),$$

siendo $\alpha = (\alpha_1, \dots, \alpha_p)^t$, $f(Y) = (f(y^1), \dots, f(y^p))^t$, y

$$M(\Phi, Y) = \begin{pmatrix} \phi_1(y^1) & \dots & \phi_p(y^1) \\ \vdots & \vdots & \vdots \\ \phi_1(y^p) & \dots & \phi_p(y^p) \end{pmatrix}.$$

Considerando lo anterior, se dice que el conjunto de puntos Y es “*equilibrado*” (o *poised*) si la matriz $M(\Phi, Y)$ es no singular, para alguna base Φ en P_n^d .

En el libro [35] hay un extenso tratamiento sobre la definición de conjuntos de interpolación apropiados y las propiedades correspondientes.

Modelos de interpolación lineal

En el caso de interpolación lineal, $p = n + 1$ y sea $Y = \{y^0, y^1, \dots, y^n\}$ un conjunto de puntos de interpolación. Sean los monomios de grado 1, $\Phi = \{1, x_1, \dots, x_n\}$, una base para P_n^1 .

Luego,

$$M(\Phi, Y) = \begin{pmatrix} 1 & y_1^0 & \cdots & y_n^0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & y_1^n & \cdots & y_n^n \end{pmatrix}.$$

Siguiendo [35], definimos la *cápsula afín* del conjunto $Y \in \mathbb{R}^n$ como el conjunto afín más pequeño que contiene a Y . La cápsula afín de Y está unívocamente determinada y consiste de todas las combinaciones lineales de los elementos de Y cuyos escalares suman hasta 1. También decimos que Y es *afinmente independiente* si su cápsula afín tiene dimensión n . Así, equilibrado es sinónimo de independencia afín.

La cápsula convexa de un conjunto de $n + 1$ puntos afinmente independientes o conjunto equilibrado, $Y = \{y^0, y^1, \dots, y^n\}$ se llama *simplex*. Los $n + 1$ puntos se llaman *vértices del simplex*. Como el conjunto Y es equilibrado, la matriz

$$S = [y^1 - y^0, \dots, y^n - y^0]^t$$

es no singular y el *gradiente simplex* en y^0 está definido en la literatura de optimización (ver [35, 66]) mediante

$$(2.10) \quad \nabla_s f(y^0) = S^{-1} \delta(f, S),$$

en donde $\delta(f, S) = [f(y^1) - f(y^0), \dots, f(y^n) - f(y^0)]^t$.

El gradiente simplex está íntimamente relacionado a los modelos lineales de interpolación polinomial.

El cálculo del gradiente simplex requiere la selección de un conjunto de puntos. Las propiedades geométricas del conjunto de puntos seleccionado determina la calidad del correspondiente gradiente simplex en relación a la bondad de la aproximación al gradiente verdadero de la función objetivo.

Luego, observar que, en el caso de interpolación lineal, el gradiente simplex coincide con el gradiente del modelo $m(y^0 + s) = f(y^0) + g^t s$, denotándose:

$$\nabla_s f(y^0) = g.$$

Si Y está contenido en la bola $B(y^0, \Delta(Y))$ de radio $\Delta = \Delta(Y) = \max_{1 \leq i \leq n} \|y^i - y^0\|$ y f y ∇f son Lipchitz continuos en un abierto que contiene a $B(y^0, \Delta)$ con constante ν , entonces, para todo $y \in B(y^0, \Delta)$ ([35])

$$(2.11) \quad \|\nabla f(y) - \nabla m(y)\| \leq \kappa_{eg} \Delta$$

y

$$(2.12) \quad |f(y) - m(y)| \leq \kappa_{ef} \Delta^2,$$

donde κ_{eg} y κ_{ef} son números que dependen del número de condición de la matriz $M(\Phi, Y)$, n y de ν .

Modelos de interpolación cuadrática

Las bases que usualmente se usan son las bases de monomios, polinomios de Newton o de Lagrange, siendo la última la más utilizada por poseer ciertas propiedades para mantener el conjunto de interpolación equilibrado.

Dados un conjunto $Y = \{y^1, \dots, y^p\}$, $f(Y) = \{f(y^1), \dots, f(y^p)\}$, la función objetivo evaluada en Y , y una base de polinomios de Lagrange $\{l_i(x)\}_{i=1}^p$ (ver [[35], pag. 39]), el modelo de interpolación queda unívocamente determinado por

$$m(x) = \sum_{i=1}^p f(y^i) l_i(x).$$

La calidad del conjunto Y en relación al “buen equilibrio” (well-poised) (ver [35]) puede medirse usando los elementos siguientes:

Dado un conjunto B en \mathbb{R}^n y una base en P_n^d , un conjunto equilibrado $Y = \{y^1, y^2, \dots, y^p\}$ se denomina Λ -equilibrado en B , para algún $\Lambda > 0$, si y sólo si para la base de Lagrange asociada con Y se cumple:

$$\Lambda \geq \max_{1 \leq i \leq p} \left\{ \max_{x \in B} |l_i(x)| \right\}.$$

La acotación del error entre los valores funcionales de f y $m(x)$, y entre los gradientes de las mismas, considerando $B_2(x, \Delta) = \{v \in \mathbb{R}^n : \|v - x\|_2 \leq \sqrt{n} \Delta\}$, el conjunto de interpolación equilibrado $Y \subset B_2(x, \Delta)$ y la base de polinomios de Lagrange asociada $\{l_i(x)\}$, se obtiene para $y \in B_2(x, \Delta)$:

$$(2.13) \quad |f(y) - m(y)| \leq K_{ef} \sum_{i=1}^p (\|y^i - y\|^2 |l_i(y)|),$$

$$(2.14) \quad \|\nabla f(y) - \nabla m(y)\| \leq K_{eg} \Lambda \Delta,$$

donde $\Lambda = \max_{1 \leq i \leq p} \{\max_{z \in B_2(x, \Delta)} |l_i(z)|\}$, siendo K_{ef} y K_{eg} constantes positivas.

Otra forma alternativa para analizar el “equilibrio” puede realizarse analizando el número de condición de una matriz cuando se utiliza una base de monomios, $\bar{\Phi}$, y es una versión escalada de las direcciones determinadas por Y , \bar{S} . Con tal objetivo se considera el centro del conjunto de interpolación Y , generalmente un iterado corriente x_k , representado por ejemplo por y^1 en Y , y los incrementos: $s^i = y^i - y^1$, para $i = 2, \dots, p$. La región considerada es $B_2(0, \Delta(Y))$ siendo el radio $\Delta = \Delta(Y) = \max_{2 \leq i \leq p} \|y^i - y^1\|_2$. El conjunto escalado: $\bar{S} = \{s^i/\Delta\}_2^p$, está contenido en una esfera de radio 1, centrada en el origen.

Notación: Denominamos $\hat{M} = M(\bar{\Phi}, \bar{S}) = \begin{pmatrix} s^2/\Delta \\ \vdots \\ s^p/\Delta \end{pmatrix}$.

El siguiente resultado de [35](pag.5) se transcribe a continuación:

TEOREMA 6. Si \hat{M} es no singular y $\|\hat{M}^{-1}\|_2 \leq \Lambda$, entonces el conjunto \bar{S} es $\sqrt{p}\Lambda$ -equilibrado en $B(0, 1)$. Recíprocamente, si \bar{S} es Λ -equilibrado en $B(0, 1)$, entonces el número de condición $\kappa_2(\hat{M}) \leq \theta p^2 \Lambda$, donde $\theta > 0$ es dependiente de n y d es independiente de \bar{S} y Λ .

Este resultado indica que el número de condición de \hat{M} puede usarse para analizar la geometría del conjunto de puntos de interpolación sin calcular los polinomios de Lagrange y Λ . También se puede concluir que si el conjunto \bar{S} está bien equilibrado entonces los cálculos numéricos a partir de usar \hat{M} no tendrán errores numéricos significativos.

En el contexto del uso de modelos de interpolación, diferentes autores han propuesto distintas estrategias para mantener a lo largo del proceso iterativo un conjunto “equilibrado” [55, 74].

2.3. Métodos de optimización sin derivadas

Los métodos basados en métodos de región de confianza explotan la suavidad de la función objetivo y pretenden conservar las propiedades de convergencia de los métodos basados en gradientes. Un modelo m_k es creado para aproximar a f alrededor del iterado actual x_k . El modelo requiere que se interpole a f en un conjunto Y de puntos que contiene al iterado actual x_k y a $(p - 1)$ puntos muestrales a los cuales llamaremos puntos satélites. Se pueden escribir estas condiciones de interpolación como un sistema lineal de ecuaciones, como en (2.9), cuyas incógnitas son los coeficientes del modelo. Para que este sistema lineal esté bien definido, se debe asegurar que los puntos muestrales sean tales que las columnas formadas por las ecuaciones

(2.9) sean linealmente independientes; a esto lo llamamos *condición geométrica*, y si se cumple decimos que el conjunto de puntos muestrales es no degenerado o equilibrado.

En el método descrito por Conn, Scheinberg y Toint en [33], se obtiene un paso s_k minimizando $m_k(x_k + s)$ sujeto a la región de confianza $\|s\|_2 \leq \Delta_k$, donde el radio Δ_k se ajusta automáticamente de acuerdo a reglas establecidas. Después de reemplazar uno de los puntos muestrales por el nuevo punto $x_+ = x_k + s_k$ la condición geométrica puede, de todas formas, no satisfacerse. Para evitar este inconveniente se utilizan dos tipos de iteraciones para generar un nuevo punto x_+ :

1. iteraciones de “minimización” para reducir f ;
2. iteraciones de diseños “simplex” para definir un modelo que aproxime a f adecuadamente.

Si el punto de prueba x_+ no reduce el valor funcional de f , entonces se considera que el modelo no es muy bueno como aproximación local de f y se toma uno de los siguientes caminos:

- Si el conjunto de puntos muestrales es casi degenerado, o se considera que alguno de los puntos interpolados por m_k está muy lejos de x_k , entonces se requiere de un nuevo conjunto muestral y se hace una iteración de simplex. Esta devuelve un punto que es cercano a x_k y que aumenta, en alguna medida, la bondad de la geometría del simplex (por ejemplo, el determinante del simplex inducido por (2.9)).
- Si no, al nuevo punto x_+ se lo considera lejos de x_k como para que m_k sea una buena aproximación de f . Entonces, el radio de la región de confianza se reduce y se hace otra iteración de minimización.

2.3.1. Método de cuñas.

En contraste con el método de Conn, Scheinberg y Toint, el método de Marazzi y Nocedal [74] sólo hace un tipo de iteración. En vez de resolver un subproblema de región de confianza estándar y prestar atención si el nuevo punto $x_+ = x_k + s_k$ satisface o no las propiedades geométricas, impone una condición geométrica implícitamente en el cálculo, garantizando que el nuevo conjunto de puntos defina al modelo adecuadamente. Esto, junto con un mecanismo que asegura que el modelo m_k aproxima bien a f , son las claves de este método, el cual tiene dos versiones, dependiendo de si se usa un modelo m_k lineal o cuadrático.

Supongamos que se comienza la iteración actual con un conjunto de puntos Y no degenerado. Antes de calcular el nuevo punto de prueba usando el modelo m_k , se identifica al punto de Y que

está más lejos de x_k , llamémoslo y^{out} , el cual será extraído de Y cuando se obtenga un nuevo punto. Esta elección hace que el conjunto de puntos muestrales sea tal que dan información de f alrededor de x_k . Luego, se define una *región tabú* en \mathbb{R}^n , T_k , que contiene a los puntos $x_k + s$ que, de ser incluídos en el conjunto de interpolación en lugar de y^{out} , daría como resultado un conjunto de puntos muestrales degenerado. También se define un conjunto W_k , llamado “*cuña*”, que contiene a T_k , y que es diseñado de tal forma que evade los puntos cercanos a T_k .

Una vez que la “*cuña*” W_k es determinada, se calcula el paso de prueba s_k resolviendo de forma aproximada

$$(2.15) \quad \min_s m_k(x_k + s) = f(x_k) + g_k^t s + \frac{1}{2} s^t G_k s$$

$$(2.16) \quad \text{s.a. } \|s\| \leq \Delta_k,$$

$$(2.17) \quad s \notin W_k,$$

y se define $x_+ = x_k + s_k$. Si el nuevo punto x_+ reduce el valor funcional de f , entonces x_+ es el nuevo iterado, y x_k se convierte en un punto satélite. Si, por el contrario, x_+ no reduce el valor funcional de f , x_k permanece como el iterado actual, y x_+ puede o no ser descartado, dependiendo de cuan lejos está de x_k comparado con y^{out} .

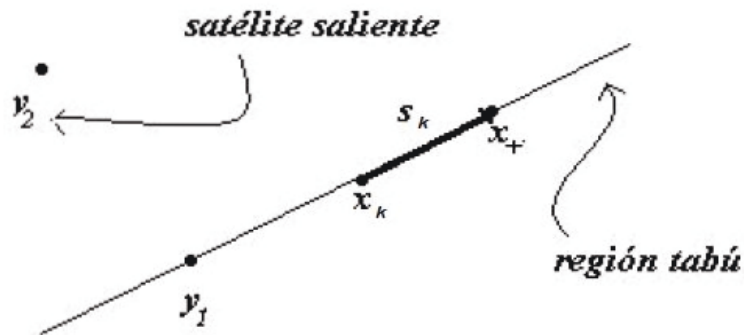


FIGURA 2. Ejemplo para $n = 2$ con modelo de interpolación lineal. El conjunto de puntos muestrales en la iteración actual es $Y = \{x_k, y_1, y_2\}$. Como y_2 es el punto más alejado de x_k , es el candidato a ser removido del conjunto muestral. Si el paso de prueba s_k está sobre la *región tabú*, el nuevo conjunto muestral, $Y = \{x_+, x_k, y_1\}$, será degenerado.

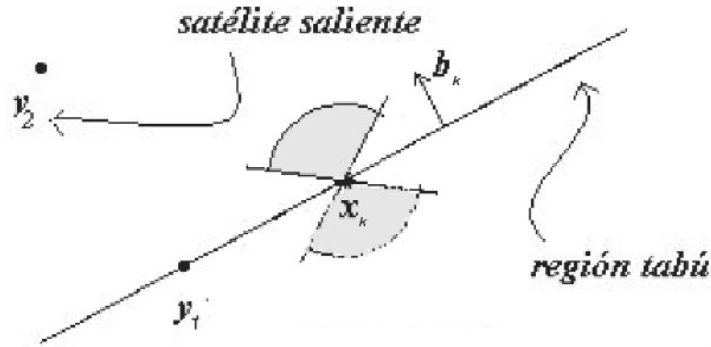


FIGURA 3. Ejemplo para $n = 2$ con modelo de interpolación cuadrático. Los pasos s que caigan en el área sombreada deben satisfacer (2.17) y no deben pertenecer a la región tabú T_k . El círculo representa la región de confianza y el vector b_k es perpendicular a las direcciones que definen la región tabú.

La región tabú T_k , que es la región que se pretende evitar al calcular el nuevo punto, para que el conjunto de puntos muestrales sea no degenerado, está definido por un subespacio $(n-1)$ -dimensional generado por los vectores de desplazamiento

$$(2.18) \quad \{s^l : l = 1, \dots, n, \quad l \neq l_{out}\},$$

correspondientes a los satélites que permanecen en el conjunto muestral, en el caso de interpolación lineal y de manera similar, luego de algunas cuentas, para el caso de interpolación cuadrática. Una representación más conveniente es

$$T_k = \{s \in \mathbb{R}^n : b_k^t s = 0\},$$

donde $b_k \in \mathbb{R}^n$ es normal a los vectores de desplazamiento (2.18).

2.3.2. El algoritmo NEWUOA.

NEWUOA es un algoritmo propuesto por Powell en [90] basado en sus artículos [86, 87, 88, 89]. Este método presenta una sofisticada estrategia con el fin de manejar el radio de la región de confianza y el radio del conjunto de interpolación. El más chico de los dos radios es usado para forzar que los puntos de interpolación se mantengan suficientemente alejados para evitar la influencia de ruido en las evaluaciones funcionales. Luego, el paso de adaptación de la región de confianza en NEWUOA es más complicado que el paso de un método clásico de

región de confianza [35].

Las principales características de NEWUOA son las siguientes:

- (i) Usa aproximaciones cuadráticas de la función objetivo con el fin de obtener una convergencia más rápida en algoritmos iterativos para optimización irrestricta. De todas maneras, cada modelo cuadrático tiene $\frac{1}{2}(n+1)(n+2)$ coeficientes independientes a ser determinados, y este número puede ser demasiado grande en problemas con n grande. Por lo que, NEWUOA construye modelos cuadráticos adecuados con una menor cantidad de datos. Cada conjunto de interpolación tiene p puntos donde $n+2 \leq p \leq \frac{1}{2}(n+1)(n+2)$. El valor por defecto en NEWUOA es $p = 2n + 1$.

Como p puede tomar un valor menor que $\frac{1}{2}(n+1)(n+2)$, el conjunto de interpolación Y puede no ser completo. Los grados de libertad restantes son calculados minimizando la norma Frobenius de la diferencia de dos modelos Hessianos consecutivos [89]. Este procedimiento define al modelo de manera única, siempre que las restricciones (2.9) sean consistentes y $p \geq n + 2$ pues, la norma Frobenius es estrictamente convexa. Las actualizaciones a lo largo de las iteraciones aprovechan la suposición de que cada actualización de los puntos de interpolación es la sustitución de un solo punto por uno nuevo. Powell justifica en [89] que, usando polinomios de Lagrange en los puntos de interpolación, cuando un punto x_+ es reemplazado por uno de los puntos de Y , es posible mantener la independencia lineal de las condiciones de interpolación (2.9). Se ha demostrado que estas condiciones son heredadas por los nuevos puntos de interpolación, cuando x_+ reemplaza a y^t en Y , cada vez que se eligen de manera que el polinomio de Lagrange $l_t(x_+)$ es no nulo. Más aún, la preservación de la independencia lineal del sistema lineal (2.9) en las iteraciones sucesivas, en presencia de errores de redondeo computacionales, puede ser más estable si $|l_t(x_+)|$ es relativamente grande.

- (ii) Resuelve el subproblema cuadrático de minimización en una región de confianza usando un método de gradientes conjugados truncados. Adopta la norma Euclídea para definir la región de confianza.
- (iii) Adapta el conjunto de puntos de interpolación siguiendo los siguientes pasos:
 - (a) Si la minimización en la región de confianza en la k -ésima iteración produce un paso s el cual es demasiado corto comparado con la distancia máxima entre los puntos muestrales y el iterado actual, entonces la función f es evaluada en $x_k + s$ y se lo acepta como nuevo iterado, x_{k+1} , siempre que la reducción de f sea suficiente. Más aún, si

el nuevo punto $x_k + s$ es aceptado como nuevo iterado, éste es incluido en el conjunto de puntos de interpolación Y , eliminando el punto y^t cuya distancia $\|x_k - y^t\|_2$ y el valor $|l_t(x_k + s)|$ sea lo más grande posible. La compensación entre estos dos objetivos se alcanza mediante la maximización del valor absoluto ponderado $\omega_t |l_t(x_k + s)|$, donde ω_t refleja la distancia $\|x_k - y^t\|_2$.

(b) Si el paso s es rechazado, el nuevo punto $x_k + s$ podría ser aceptado en Y , eliminando el punto y^t que maximice al valor absoluto $\omega_t |l_t(x_k + s)|$, siempre y cuando $|l_t(x_k + s)| > 1$ o $\|x_k - y^t\|_2 > r\Delta_k$ se satisfaga para un $r \geq 1$ dado.

(c) Si la mejora en la función objetivo no es suficiente, y se considera que el modelo debe ser mejorado, entonces el algoritmo elige un punto en Y , que es el que está más lejos a partir de x_k , e intenta reemplazarlo con un nuevo punto que maximice el valor absoluto del correspondiente polinomio de Lagrange en la región de confianza.

El esquema general de NEWUOA es el siguiente:

Paso 0: Inicio. Dados x_0 , NPT el número de puntos de interpolación, Y el conjunto de puntos de interpolación, $\rho_0 > 0$, $0 < \rho_{end} < 0.1\rho_0$, $\Delta = \rho_0$, $\rho = \rho_0$, $t = 0$, $k \leftarrow 0$.

Construir un modelo cuadrático inicial $m_0(x_0 + s)$ de la función $f(x)$.

Paso 1: Resolver el problema cuadrático en la región de confianza. Calcular $\bar{s} = \operatorname{argmin} m_k(x_k + s)$ sujeta a $\|s\|_2 \leq \Delta$.

Paso 2: Test de aceptación. Si $\|\bar{s}\|_2 \geq 0.5\rho$, calcular $ratio = (f(x_k) - f(x_k + \bar{s})) / (m_k(x_k) - m_k(x_k + \bar{s}))$.

Paso 2.1: Actualización del radio de la región de confianza. Reducir Δ y mantener ρ . Establecer y^t el punto de interpolación más lejano desde x_k . Si $\|y^t - x_k\| \geq 2\Delta$ ir al Paso 3, sino ir al Paso 5.

Paso 2.2: Actualización del radio de la región de confianza y de ρ . Ir al Paso 6.

Paso 3: Iteración alternativa. Recalcular \bar{s} con el fin de mejorar la geometría del conjunto de interpolación.

Paso 4: Actualización del conjunto de interpolación y del modelo cuadrático.

Paso 5: Actualización de la aproximación. Si $ratio > 0$, fijar $x_{k+1} \leftarrow x_k + \bar{s}$, $k \leftarrow k + 1$ e ir al Paso 1.

Paso 6: Criterio de parada. Si $\rho = \rho_{end}$ declarar "fin del proceso", si no fijar $x_{k+1} \leftarrow x_k$, $k \leftarrow k + 1$ e ir al Paso 1.

El procedimiento anterior finaliza si se satisface alguna de las siguientes opciones: el modelo cuadrático no decrece, $\rho = \rho_{end}$ o si se alcanza la cantidad máxima de iteraciones.

El algoritmo BOBYQA

Los resultados numéricos de NEWUOA [91] alentaron al autor a introducir algunas modificaciones en éste con el fin de resolver problemas de optimización con cotas. Como resultado, Powell ha propuesto el algoritmo BOBYQA [92] para resolver el problema

$$\text{mín } f(x) \text{ sujeto a } x \in \Omega,$$

con $\Omega = \{x \in \mathbb{R}^n \mid L \leq x \leq U\}$, y sin hacer uso de las derivadas de la función objetivo la cual, como antes, se la considera diferenciable.

BOBYQA tiene un esquema similar al NEWUOA aunque las principales diferencias son:

1. Los iterados x_k satisfacen las cotas del problema. Luego, en el Paso 1 se minimiza la aproximación cuadrática de $f(x)$ en la intersección de la región de confianza con $\Omega = \{x \in \mathbb{R}^n \mid L \leq x \leq U\}$.

2. También tiene iteraciones alternativas en donde el nuevo punto es determinado de tal manera de asegurar la independencia lineal en las condiciones de interpolación. En este paso, se ignora el modelo m_k con el fin de darle más atención a la actualización de la matriz inversa del sistema, necesaria para adaptar el modelo cuadrático de $f(x)$ cuando se cambia el conjunto de puntos de interpolación.

3. Además tiene un nuevo dispositivo que intenta restablecer normalidad en el caso en que haya pérdida de la precisión al calcular la matriz para adaptar el modelo cuadrático m_k .

2.3.3. Convergencia global de algoritmos de Región de Confianza en optimización sin derivadas. La convergencia de algoritmos de optimización sin derivadas ha recibido mucha atención en los trabajos de Conn, Scheinberg y Vicente (1997, 2009a, 2009b)[31, 34, 35]. Ellos han desarrollado gran parte de la teoría publicada sobre métodos sin derivadas que usan pasos de región de confianza, usando aproximaciones lineales o cuadráticas a $f(x)$ en toda iteración.

Powell, en [93], hace un estudio de estos métodos para optimización sin restricciones, cuando $f(x)$ es diferenciable. Considera el modelo lineal o cuadrático. Los modelos interpolan a f en $n + 1$ puntos, definiendo unívocamente al modelo lineal. En el caso cuadrático, G es derivada de

la información de las iteraciones previas aunque tales pocos datos no estiman necesariamente bien las segundas derivadas. Propone un método donde los iterados provienen de los pasos de región de confianza o son diseñados para mantener el volumen y diámetro de la cápsula convexa de los puntos de interpolación.

Se asume que la aproximación m_k satisface las $n + 1$ condiciones de interpolación:

$$(2.19) \quad m_k(y^i) = f(y^i), i = 0, 1, \dots, n,$$

donde y^0 es el punto x_k , y todos los valores $f(y^i)$ han sido previamente calculados. Es importante que las condiciones de interpolación previas definan unívocamente al gradiente g_k . Un enunciado equivalente de esa condición de no degeneramiento es que la matriz $D_k \in \mathbb{R}^{(n+1) \times (n+1)}$, de los vectores $[y^i; 1]$, para $i = 0, 1, \dots, n$ sea no singular. Otro resultado equivalente es que el volumen de la cápsula convexa en \mathbb{R}^n de los puntos y^i sea no nula, ya que tal volumen es $|\det D_k|/n!$.

En el análisis realizado por los autores citados, G_k es una matriz simétrica y acotada y g_k está definido por las ecuaciones con $m_k(x_k) = f(x_k) = f(y^0)$. En general, se usa G_k no nula, porque frecuentemente el progreso de algoritmos iterativos es demasiado lento si se ignora la curvatura de la función objetivo.

Sea $\Delta_k > 0$ el radio de la región de confianza en la k -iteración, lo cual significa que, el paso s_k satisface que $\|s_k\| \leq \Delta_k$. Además consideran que las aproximaciones lineales a f , definidas por condiciones de interpolación, satisfacen las condiciones

$$(2.20) \quad \|y^i - x_k\| \leq c_1 \Delta_k, \quad \text{para } i = 1, 2, \dots, n,$$

y la condición $|\det D_k| \geq c_2 \Delta_k^n$, donde c_1 y c_2 son constantes positivas. Estas condiciones también fueron empleadas en el Algoritmo COBYLA de Powell (1994) [85]. Las desigualdades anteriores, junto con la acotación de $\nabla^2 f(x_k)$ y $\nabla^2 m_k$, implican que el gradiente g_k del modelo, satisface:

$$(2.21) \quad \|g_k - \nabla f(x_k)\| \leq c_3 \Delta_k$$

donde c_3 es una constante positiva. Una demostración de ese resultado está dada en Conn et al. (2009b) [35].

También se observa que, si la norma de $\nabla f(x_k)$ es mucho más grande que $c_3 \Delta_k$, es posible demostrar que la condición anterior implica que el error relativo de la aproximación

$$f(x_k) - f(x_k + s_k) \simeq m(x_k) - m(x_k + s_k),$$

es pequeño. En ese caso, la reducción de m_k debido al paso de la “región de confianza”, es heredada por f . En esa situación $\Delta_{k+1} \geq \Delta_k$. Entonces, se deduce que, cuando $\Delta_{k+1} < \Delta_k$, $\nabla f(x_k) \leq c_5 \Delta_k$.

Basado en los resultados previos, se define el conjunto K de índices, que incluye el índice de una iteración k si y solamente si Δ_{k+1} es menor a todos los Δ_j , tales que $j = 1, 2, \dots, k$. Luego, si el número de elementos de K es infinito, y si la sucesión decreciente $\{\Delta_{k+1}\}_{k \in K}$ tiende a cero, entonces se puede demostrar que $\|\nabla f(x_k)\|_{k \in K}$ tiende a cero.

Finalmente, Powell [93] demuestra que un algoritmo de región de confianza, bajo las condiciones descriptas, y satisfaciendo ciertas hipótesis comunes para esta clase de problemas, satisface el resultado siguiente:

TEOREMA 7. Sea un algoritmo descrito en la Sección 2 del trabajo [93], aplicado a una función f , que es acotada inferiormente y que tiene segundas derivadas continuas, y el número de iteraciones es infinito. Entonces, la sucesión de los $\nabla f(x_k)$, $k = 0, 1, 2, \dots$ converge al vector nulo.

Método de restricciones activas en el Método de Powell

En este capítulo presentamos el algoritmo TRB-Powell (Trust-Region Box)[12]. Este método se utiliza para resolver problemas de optimización, irrestrictos o con restricciones en las variables, sin calcular derivadas de la función objetivo. El método TRB-Powell está basado en el método de NEWUOA de Powell [90].

Consideramos el siguiente problema de optimización,

$$\text{mín } f(x) \text{ s. a } x \in \Omega,$$

donde $\Omega = \mathbb{R}^n$ para problemas irrestrictos y $\Omega = \{x \in \mathbb{R}^n \mid L \leq x \leq U\}$, $L < U$, para problemas de optimización con restricción en las variables.

Recordemos que, para la resolución de problemas irrestrictos, en cada iteración construimos un modelo cuadrático que interpola a la función objetivo alrededor del iterado actual y, luego, minimizamos el modelo para obtener un nuevo punto. Todo el proceso se realiza dentro de un esquema de región de confianza. En TRB-Powell utilizamos norma infinito, en vez de la norma Euclídea usada en NEWUOA, y resolvemos el subproblema cuadrático restringido a una caja usando una estrategia de conjuntos activos para explorar las caras de la caja, según la estrategia descrita por Birgin y Martínez en [23]. Luego, extendimos la implementación para problemas con restricciones de cotas.

Comparamos numéricamente TRB-Powell con NEWUOA y BOBYQA, los algoritmos sin derivadas para problemas irrestrictos y problemas con restricciones de cotas de Powell [90, 92].

3.1. Conjuntos activos para la resolución del subproblema cuadrático en una caja

En los algoritmos NEWUOA y BOBYQA, el subproblema de minimización del modelo cuadrático en la región de confianza es una de las partes más caras en cuanto a cálculo. Ambos algoritmos usan la norma Euclídea para definir la región de confianza. En nuestro caso, por tener restricciones de cotas en las variables, usamos la norma infinito.

Dados x_k y $\Delta_k > 0$, la aproximación y el radio de la región de confianza actual, definimos la región de confianza por

$$B_k = \{s \in \mathbb{R}^n \mid \|s\|_\infty \leq \Delta_k\}.$$

El subproblema de región de confianza está dado por

$$(3.1) \quad \begin{aligned} \text{mín} \quad & m_k(x_k + s) = f_k + g_k^T s + \frac{1}{2} s^T G_k s \\ \text{s. a} \quad & s \in \Omega_k \end{aligned}$$

donde $\Omega_k = \{s \in B_k : x_k + s \in \Omega\}$, $f_k = f(x_k)$, $g_k \in \mathbb{R}^n$ y G_k es una matriz simétrica de $n \times n$. En TRB-Powell hemos reemplazado el solver de NEWUOA y BOBYQA por un método de conjuntos activos que utiliza la estrategia descrita por Birgin y Martínez en [23].

Específicamente, como lo han propuesto Friedlander y Martínez en [46], definimos una cara abierta de $\Omega_k = \{s \in \mathbb{R}^n : l \leq s \leq u\}$ como el conjunto $F_I \subset \Omega_k$ tal que

$$F_I = \{s \in \Omega_k \mid s_i = l_i \text{ si } i \in I, s_i = u_i \text{ si } n+i \in I, l_i < s_i < u_i \text{ caso contrario}\}$$

siendo I un subconjunto (posiblemente vacío) de $\{1, 2, \dots, 2n\}$ tal que i y $n+i$ no pueden pertenecer simultáneamente a I para todo $i \in \{1, \dots, n\}$.

EJEMPLO 3.1. Sea $\Omega_k = \{x \in \mathbb{R}^3 : 0 \leq x_1 \leq 1, 0 \leq x_2 \leq 1, 0 \leq x_3 \leq 1\}$.

Luego,

$$F_{\{1\}} = \{x \in \mathbb{R}^3 : x_1 = 0, 0 < x_2 < 1, 0 < x_3 < 1\},$$

$$F_{\{1, 5\}} = \{x \in \mathbb{R}^3 : x_1 = 0, x_2 = 1, 0 < x_3 < 1\},$$

$$F_{\{1, 2, 6\}} = \{(0, 0, 1)\}, F_\emptyset = \{x \in \mathbb{R}^3 : 0 < x_1 < 1, 0 < x_2 < 1, 0 < x_3 < 1\}, \text{ etc.}$$

También, definimos V_I , como el *menor subespacio afín* que contiene a F_I , y S_I , el *subespacio lineal paralelo* a V_I . La clausura de F_I , V_I y S_I dividen al conjunto factible, Ω_k , en 3^n caras disjuntas.

El *gradiente proyectado continuo* en $s \in \Omega_k$ se define como

$$g_P(s) = P_{\Omega_k}(s - g(s)) - s,$$

donde $g(s)$ es el gradiente del modelo evaluado en $x_k + s$. Ver Figura 1.

Para todo $s \in F_I$, definimos el *gradiente proyectado interno*

$$g_I(s) = P_{S_I}[g_P(s)]$$

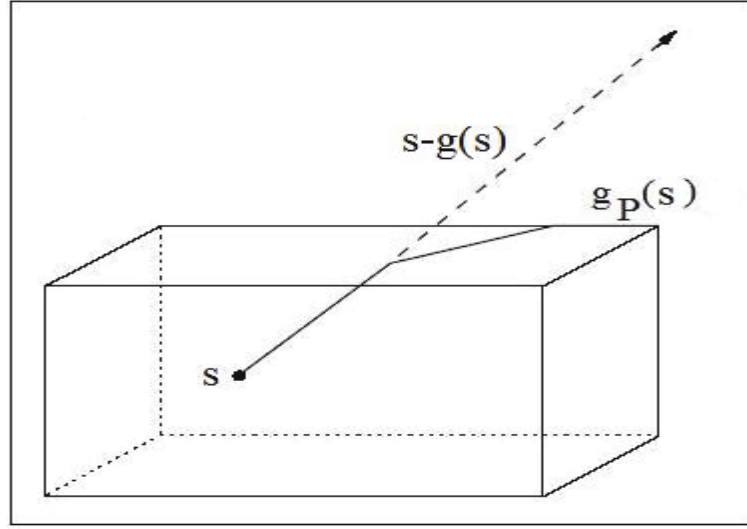


FIGURA 1.

el cual es la proyección de g_P en el subespacio lineal S_I .

Luego, supongamos que s^j es el iterado actual, el cual se encuentra en una cara, F_I , de Ω_k . Para decidir si es conveniente dejar esa cara o continuar explorándola para obtener un valor menor de m_k , calculamos el gradiente $\nabla m_k(x_k + s^j)$, su proyección sobre Ω_k ($g_P(s^j)$) y la proyección sobre S_I ($g_I(s^j)$).

Dado $\eta \in (0, 1)$, si

$$(3.2) \quad \|g_I(s^j)\| \geq \eta \|g_P(s^j)\|$$

se calcula un mínimo aproximado de $m_k(x_k + s)$ sobre la clausura de F_I usando el método de Gradientes Conjugados ([83]). La Figura 2 ilustra este caso.

Si la condición (3.2) no se satisface, la cara es abandonada y s^{j+1} es calculado realizando una iteración del método de gradiente espectral proyectado (SPG) como se describe en [23].

El procedimiento de conjuntos activos finaliza cuando $\|g_P(s^*)\|$ es menor que cierta tolerancia, $\epsilon_P > 0$, para algún iterado s^* .

Usando el paso de SPG varias restricciones activas pueden agregarse o eliminarse en cada iteración, al pasar de un s^j a un s^{j+1} , por lo que el método resulta muy útil en problemas grandes. Además, los resultados numéricos han demostrado que esta estrategia es exitosa y eficiente para problemas medianos y grandes [7, 8, 22, 23, 25].

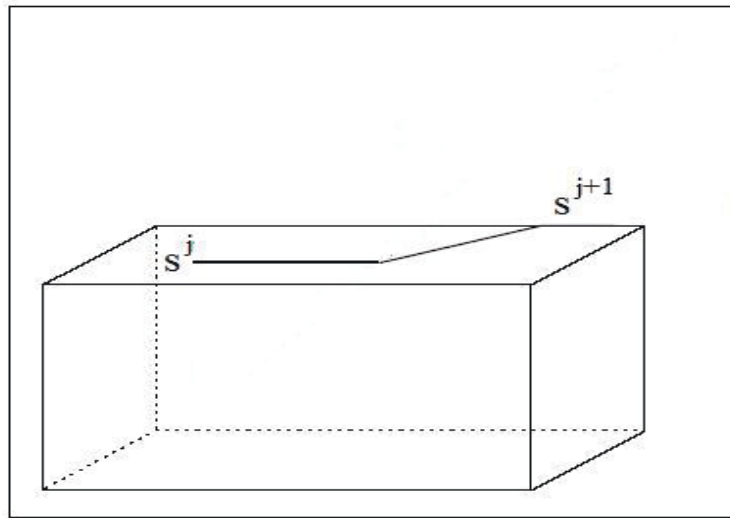


FIGURA 2.

En la Figura 2 se observa que, usando una iteración del método SPG se puede pasar de un punto interior ($s \in F_\emptyset$) a un punto sobre una arista o sobre un vértice, ilustrando la posibilidad de pasar de un punto sin restricciones activas a otro con tres restricciones activas. Análogamente, se podría ilustrar el caso en el cual desde un punto que satisface varias restricciones activas se pasa a un punto en donde no hay restricciones activas.

Los resultados teóricos en [23] nos permiten asegurar que la aplicación de éste método al modelo cuadrático restringido a Ω_k está bien definido y se obtiene el criterio de convergencia del método. De hecho, Birgin y Martínez han demostrado que es posible hallar, con una precisión arbitraria, un punto que satisface las condiciones de Karush-Kuhn-Tucker (KKT). También, asumiendo que todos los puntos estacionarios son no degenerados, el algoritmo identifica la cara a la cual pertenece en una cantidad finita de iteraciones [23].

3.2. Detalles de implementación

Hemos usado los parámetros default para los códigos NEWUOA y BOBYQA. Ejecutamos ambos códigos con $2n+1$ puntos de interpolación usando la aproximación de la norma Frobenius, tal como lo sugiere Powell [90].

Los puntos iniciales y el radio inicial de la región de confianza son los mismos que en las referencias citadas [55, 62, 80, 81, 90].

El criterio de parada que hemos usado es el mismo que ha usado Powell, este es, el proceso iterativo para cuando el radio de la región de confianza es menor que una tolerancia dada, $\rho_{end} = 10^{-6}$ o cuando el máximo de evaluaciones funcionales permitidas es alcanzado.

La cantidad máxima de evaluaciones funcionales permitidas para el caso irrestricto son: $maxfun = 9000$, para problemas de pequeño porte, y $maxfun = 80000$, para problemas de medio porte.

La cantidad máxima de evaluaciones funcionales permitidas para el caso de restricciones de cota es:

$maxfun = 9000$, para problemas de pequeño porte, y $maxfun = 20000$, para problemas de medio porte.

En las siguientes tablas, el símbolo (**) indica que el solver respectivo no ha podido encontrar una solución o que se ha alcanzado el número máximo de evaluaciones funcionales permitido.

Los parámetros algorítmicos en TRB-Powell (Paso 1) usados:

$$\varepsilon_P = 10^{-6}.$$

$\eta = 0.1$. Hemos analizado otros valores, pero los mejores resultados se obtuvieron con este valor de η .

$$\gamma = 10^{-4}, \sigma_{\min} = 10^{-10} \text{ y } \sigma_{\max} = 10^{10}.$$

En el método de gradientes conjugados hemos usado $\bar{\varepsilon} = 10^{-8}$.

3.3. Experimentación numérica

Hemos comparado TRB-Powell con NEWUOA y BOBYQA en cantidad de evaluaciones funcionales, como usualmente se hace en los artículos de optimización sin derivadas. Hemos elegido dos grupos de problemas test, de pequeño y medio porte. TRB-Powell ha sido desarrollado en Fortran 77, como así también NEWUOA y BOBYQA. Hemos usado Intel Fortran Compiler 9.1.036. Los códigos han sido compilados y ejecutados en una PC que opera con Linux OS, AMD 64 4200 Dual Core.

3.3.1. Problemas test. En el caso irrestricto, hemos considerado un conjunto de problemas de pequeño porte propuestos por Moré y Wild en [81]. La mayoría de estos problemas de minimización son de mínimos cuadrados con restricciones no lineales los cuales fueron adquiridos de la colección CUTer [53]. La cantidad de variables de estos problemas varía entre 2 y 12. Además, con el objetivo de testear problemas con dimensiones más grandes, hemos considerado varios problemas empleados por Powell [90] en donde n varía de 20 a 100. Estos problemas son

Arwhead, Chrosen, Chebyqad, Extended Rosenbrock, Penalty1, Penalty2, Penalty3 y Vardim. Todas estas funciones son suaves y los respectivos resultados verdaderos se encuentran en la Tabla 3.

Por otro lado, para el caso de optimización con restricciones de cotas, hemos considerado algunos problemas de pequeño porte de [62, 80, 81] cuyos nombres y dimensiones se exponen en la Tabla 4.

Con el fin de testear problemas de medio porte, hemos considerado las funciones Arwhead, Chebyqad, Penalty1 y Chrosen sujetas a restricciones de cotas. Las restricciones de cotas de los primeros tres problemas requieren que todas las componentes de x estén en el intervalo $[-10, 10]$ y en el caso de la función Chrosen en el intervalo $[-3, 0]$. Más aún, hemos considerado un problema test en particular estudiado por Powell en [92]. Este problema, llamado “puntos en el cuadrado (Invdist2)”, tiene muchos mínimos locales diferentes. La función objetivo es la suma de los recíprocos de todas las distancias de a pares de puntos entre los puntos P_i , $i = 1, \dots, M$ en dos dimensiones, donde $M = n/2$ y las componentes de P_i son $x(2i - 1)$ y $x(2i)$. Luego, cada vector $x \in \mathbb{R}^n$ define los M puntos P_i . El punto inicial considerado, x_0 , está dado como el punto que está equidistante a todos los puntos sobre un círculo. Las restricciones de cotas de este problema requieren que todas las componentes de x esten en el intervalo $[-1, 1]$.

Los detalles de los resultados dados en [92] muestran que estos son muy sensibles a errores de redondeo computacionales. También hemos considerado problemas de medio porte de [80]. Estos son Ackley, Griewank y Rastrigin los cuales tienen muchos mínimos locales al tomarse los diferentes dominios de búsqueda: $15 \leq x(i) \leq 30$, $-600 \leq x(i) \leq 600$, y $-5,12 \leq x(i) \leq 5,12$, $i = 1, \dots, n$. Estas funciones tienen mínimo global en $x^* = 0$ y el correspondiente valor funcional es $f(x^*) = 0$.

Más aún, hemos considerado algunos problemas de CUTer que han sido usados en Gratton et al. [55], en donde los autores comparan su programa BC-DFO con BOBYQA. La lista detallada de estos problemas y sus características se encuentra en la Tabla 1. La Tabla muestra el nombre y la dimensión del problema, la cantidad de variables que están acotadas inferior y superiormente y el valor mínimo f^* el cual ha sido reportado en [55]. Si alguna variable no tiene cota inferior o superior, nosotros hemos utilizado una cota muy grande para consideraciones numéricas (-10^{10} or 10^{10} respectivamente). Los resultados de estos experimentos pueden verse en la Tabla 6.

Problema	n	lbound	ubound	(l+u)bound	f^*
Biggsb1	25			24	1.50000000000000D-02
Bqp1var	1			1	0.00000000000000D+00
Camel6	2			2	-1.03162845348988D+00
Chebyqad	4			4	2.56057805386809D-22
Chenhark	10	10			-2.00000000000000D+00
Cvxbqp1	10			10	2.47500000000000D+00
Explin2	12			13	-7.09247239439664D+03
Hatflda	4	4			1.61711062151584D-25
Hatfldc	25			24	3.43494690036517D-27
HS1	1	1			7.13660798093435D-24
HS110	10			10	-4.57784755318868D+01
HS2	2	1			4.94122931798918D+00
HS25	3			3	1.81845940377455D-16
HS3	2	1			1.97215226305253D-36
HS38	4			4	2.02675622883580D-28
HS3mod	2	1			0.00000000000000D+00
HS4	2	2			2.66666666400000D+00
HS45	5			5	1.0000000040000D+00
HS5	2			2	-1.91322295498104D+00
Logros	2	2			0.00000000000000D+00
Mccormck	10			10	-9.59800619474625D+00
Mdhole	2	1			7.52316384526264D-35
Ncvxbqp1	10			10	-2.20500000000000D+04
Ncvxbqp2	10			10	-1.43818650000000D+04
Oslbqp	8	5			6.25000000000000D+00
Probpenl	10			10	-2.11912948080046D+05
Pspdoc	4		1		2.41421356237309D+00
Qudlin	12			12	-7.20000000000000D+03
Simbqp	2			1	0.00000000000000D+00
Sineali	4			4	-2.83870492243045D+02

TABLA 1. Problemas test con restricciones de cota de CUTer

3.3.2. Resultados numéricos: problemas irrestrictos. Las Tablas 2 y 3 muestran el nombre de los problemas de optimización irrestrictos de pequeño y medio porte respectivamente, la cantidad de evaluaciones funcionales (Feval) realizadas hasta alcanzar el criterio de parada y los valores funcionales obtenidos para los códigos TRB-Powell y NEWUOA ($f(x_{end})$).

Los resultados en la Tabla 2 nos permiten hacer las siguientes observaciones.

La cantidad de evaluaciones funcionales requeridas por NEWUOA es menor en 16 de los 36 problemas mientras que para TRB-Powell este número es menor en 17 problemas. En el resto de los problemas, ambos algoritmos hicieron la misma cantidad de evaluaciones funcionales.

El promedio de evaluaciones requeridas en todo el conjunto de problemas fue de 1119 para TRB-Powell y 1272 para NEWUOA.

En el problema Bard, NEWUOA encuentra un mínimo local mientras que TRB-Powell obtuvo el mínimo global.

Los valores funcionales obtenidos por ambos métodos son similares.

NEWUOA obtuvo menores valores funcionales en 18 problemas mientras que TRB-Powell lo hizo en 14 de ellos.

Los resultados obtenidos se resumen en la Figura 3 usando el perfil de desempeño descrito por Dolan y Moré en [43]. Dados un conjunto de problemas P y un conjunto S de solvers de optimización, se compara el desempeño del problema $p \in P$ por un algoritmo en particular $s \in S$, con el mejor desempeño hecho por alguno de los solvers en este problema. Notamos $t_{p,s}$ la cantidad de evaluaciones funcionales requeridas para resolver el problema $p \in P$ por el método $s \in S$, se define el cociente de desempeño

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}}$$

y se asume que $r_{p,s} \in [1, r_M]$ con $r_{p,s} = r_M$ sólo en el caso en que el problema p no es resuelto por el solver s . También se define la fracción

$$\varrho_s(\tau) = \frac{1}{n_p} \text{size}\{p \in P : r_{p,s} \leq \tau\},$$

donde n_p es la cantidad de problemas resueltos. Luego, dibujamos $\varrho_s(\tau)$, con $\tau \in [1, r_M]$. En el gráfico de perfil de desempeño, la curva superior representa al método más eficiente dentro de un factor τ de la mejor medida. El porcentaje de los problemas test para los cuales un método dado es mejor, en cuanto al criterio de desempeño que ha sido estudiado, se representa en el eje izquierdo del gráfico. Es necesario señalar que cuando ambos métodos coinciden con el mejor resultado, ambos reciben la misma marca. Esto significa que la suma de los porcentajes exitosos puede exceder el 100 %.

La Figura 3 muestra el perfil de desempeño para ambos solvers en el intervalo $[1, 1.97]$. Se puede observar que NEWUOA realiza menos evaluaciones funcionales en el 52 % de los

Problema		Feval		$f(x_{end})$	
nombre	n	NEWUOA	TRB-Powell	NEWUOA	TRB-Powell
Linear Full Rank	9	42	42	3.600000D+01	3.599990D+01
Linear Rank 1	7	159	308	8.380200D+00	8.380202D+00
Linear Rank 1*	7	151	225	9.880500D+00	9.880507D+00
Rosenbrock	2	195	134	1.382200D-17	1.089153D-16
Helical valley	3	124	146	2.066500D-13	4.180530D-14
Powell singular	4	476	477	1.099100D-10	1.350267D-10
Freudenstein & Roth	2	76	74	4.898420D+01	4.898420D+01
Bard	3	114	145	1.774900D+00	8.215013D-03
Kowalik & Osborne	4	278	274	3.075000D-04	3.075056D-04
Watson	6	937	1023	2.287600D-03	2.287600D-03
	9	9000	9000	(**) 2.160293D-06	(**) 1.782082D-06
	12	9000	9000	(**) 5.562963D-07	(**) 4.902148D-08
Box 3-dimensional	3	212	282	3.336000D-17	2.476166D-12
Jennrich & Sampson	2	55	65	1.243621D+02	1.243621D+02
Brown & Dennis	4	191	180	8.582220D+04	8.582220D+04
Chebyquad	6	177	212	4.763800D-13	1.298316D-14
	7	229	267	3.148000D-13	2.429278D-14
	8	390	247	3.516800D-03	3.516803D-03
	9	529	533	2.201700D-13	3.092237D-13
	10	666	535	4.772700D-03	4.772713D-03
	11	538	657	2.799700D-03	2.799751D-03
Brown almost-linear	10	1255	1153	1.553300D-12	1.588711D-12
Osborne 1	5	1012	1000	6.745600D-05	6.869259D-05
Osborne 2	11	1709	1024	4.013700D-02	4.013741D-02
Bdqrtic	8	432	507	1.023890D+01	1.023897D+01
	10	670	522	1.828100D+01	1.828118D+01
	11	758	614	2.226000D+01	2.226062D+01
	12	781	636	2.627200D+01	2.627277D+01
Cube	5	2842	2050	7.617000D-07	1.277429D-07
	6	4625	3080	4.772300D-06	3.328595D-06
	8	6825	4580	5.613800D-06	4.744310D-06
Mancino	5	39	55	3.712400D-11	9.547387D-14
	8	52	68	1.013600D-08	1.067564D-10
	10	68	67	2.292100D-09	1.024540D-08
	12	83	89	1.216000D-08	9.571743D-10
Heart 8	8	1118	1001	1.516400D-11	5.108511D-13

* con columnas y filas nulas

TABLA 2. Problemas irrestrictos de pequeño porte: NEWUOA vs TRB-Powell

Problema		Feval		$f(x_{end})$	
nombre	n	NEWUOA	TRB-Powell	NEWUOA	TRB-Powell
Arwhead	20	409	495	6.827872D-12	3.321787D-13
	40	1441	1079	6.320278D-12	1.322941D-12
	80	3226	2791	8.715428D-11	3.022693D-11
	100	3859	3705	4.438583D-11	4.041300D-11
Penalty 1	20	7348	6768	1.577771D-04	1.577771D-04
	40	14620	18217	3.392511D-04	3.392511D-04
	80	31180	20037	7.130502D-04	7.131754D-04
	100	39364	22341	9.024910D-04	9.027157D-04
Penalty 2	20	19060	11180	6.389751D-03	6.389759D-03
	40	15301	11889	5.569125D-01	5.569250D-01
	80	19357	16863	1.776315D+03	1.776315D+03
	100	15388	12523	9.709608D+04	9.709608D+04
Penalty 3	20	4660	1645	3.636060D+02	1.008361D-02
	40	80000	5423	(**) 1.045973D-03	1.000000D-03
	80	80000	33231	(**) 6.285525D+03	1.000000D-03
	100	80000	78017	(**) 9.882811D+03	4.658901D-02
Chrosen	20	911	1005	2.031934D-11	2.128075D-13
	40	2048	2329	4.950602D-12	5.195116D-12
	80	4764	4529	1.803505D-10	5.955498D-11
	100	4987	5814	4.664747D-10	1.350874D-10
Vardim	20	4791	6449	5.365843D-11	6.547327D-12
	40	18725	28972	7.200291D-11	6.995020D-12
	80	62562	46687	4.743123D-10	1.196846D-09
	100	80000	77995	(**) 2.408154D-08	9.065575D-08
Rosenbrock Ext.	20	8585	9301	1.491622D-10	5.035342D-12
	40	36435	30310	7.109087D-09	3.092178D-12
	80	80000	78360	(**) 1.131952D-07	2.712245D-08
	100	80000	52891	(**) 2.812546D-07	2.572285D-09
Chebyqad	20	1817	2177	4.572955D-03	4.572955D-03
	40	26743	19970	5.960843D-03	5.961027D-03
	80	73006	30800	4.931312D-03	4.931938D-03
	100	46964	35200	8.715769D-03	4.517172D-03

TABLA 3. Problemas irrestrictos de porte medio: NEWUOA vs. TRB-Powell

problemas mientras que TRB-Powell lo hace en el 55% de los problemas y éste último tiene un mejor desempeño para $\tau \in [1.18, 1.58]$.

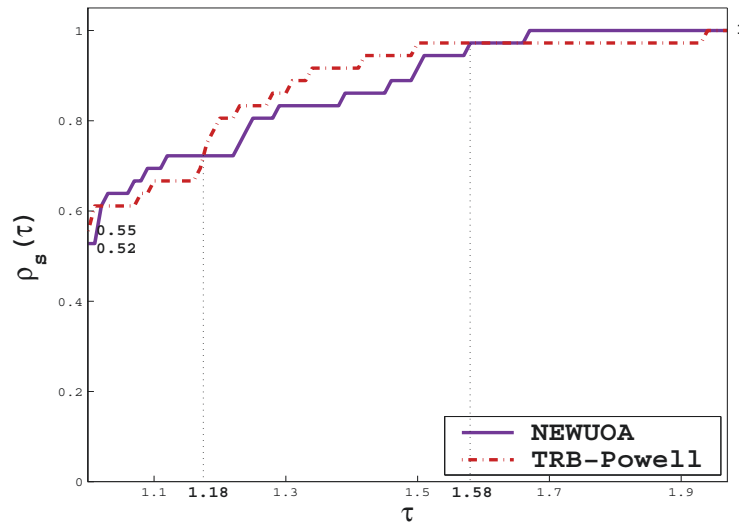


FIGURA 3. Evaluaciones funcionales para problemas irrestrictos de porte pequeño.

Para los problemas de porte medio hemos testado ocho problemas obtenidos de [90], con dimensiones $n = 20, 40, 80, 100$. En la Tabla 3 están volcados los resultados numéricos, de los cuales se puede sacar las siguientes conclusiones:

TRB-Powell requiere menos evaluaciones funcionales que NEWUOA en 23 de los 32 problemas.

El promedio de evaluaciones hechas por TRB-Powell fue 21219 y para NEWUOA fue 29611. Para ocho problemas, TRB-Powell obtuvo mejores resultados que NEWUOA (NEWUOA necesitó más de 10000 evaluaciones funcionales que TRB-Powell).

En los problemas Arwhead, Penalty 3, Rosenbrock Extended, Chrosen ($n = 20$) y Chrosen ($n = 80$), se puede ver que TRB-Powell fue más eficiente respecto al valor funcional final obtenido ($f(x_{end})$). En particular, Penalty 3 fue resuelto significativamente mejor por TRB-Powell que por NEWUOA.

TRB-Powell obtuvo menores valores funcionales que NEWUOA en 18 de los 32 problemas mientras que NEWUOA lo hizo en 9. En el resto de los problemas ambos métodos alcanzaron el mismo valor funcional.

Esto parece indicar que para los problemas de medio porte TRB-Powell tiene un mejor rendimiento que NEWUOA. Estos resultados se pueden observar en la Figura 4 usando el perfil de desempeño descrito anteriormente. Como $r_{p,s}$ es grande para varios problemas de medio porte, hemos usado una escala logarítmica de base 2 en el eje x , tal como se recomienda en [43].

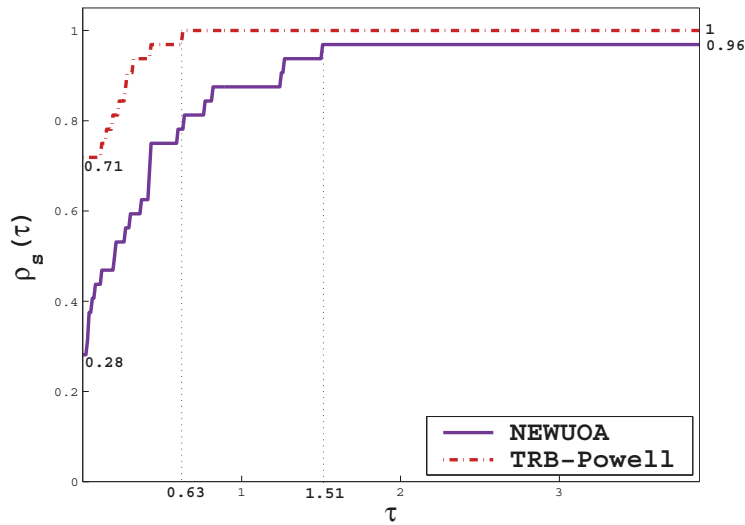


FIGURA 4. Evaluaciones funcionales para problemas irrestrictos de porte medio.

Luego, graficamos $\varrho_s(\tau) = \frac{1}{n_p} \text{size}\{p \in P : \log_2(r_{p,s}) \leq \tau\}$, con $\alpha \in [0, \log_2(r_M))$, donde $r_M > 0$ es tal que $r_{p,s} \leq r_M$, para todo p y s .

En la Figura 4 se muestra que TRB-Powell resolvió satisfactoriamente este conjunto de problemas. TRB-Powell ha tenido un mejor comportamiento en el 71 % de los problemas.

3.3.3. Resultados numéricos: Problemas con restricciones de cotas. Las Tablas 4 y 5 muestran el desempeño de BOBYQA y TRB-Powell para problemas con restricciones de cota de pequeño y medio porte, respectivamente. Se observa en la Tabla 4 que la cantidad de evaluaciones funcionales requeridas por TRB-Powell es menor que en BOBYQA en 14 de los 16 problemas test. Más aún, los valores funcionales obtenidos son similares para ambos métodos. Este desempeño de TRB-Powell se ve reflejado en la Figura 5, en donde éste ha requerido menos evaluaciones funcionales en el 93 % de los problemas.

Por otro lado, para problemas con restricciones de cota de medio porte, se observa en la Tabla 5 que TRB-Powell requiere menos evaluaciones funcionales que BOBYQA en 23 de los 33 problemas test. Además, nuestro método es comparable con BOBYQA en el sentido que TRB-Powell necesita menos evaluaciones funcionales con valores funcionales finales similares para ambos métodos. La Figura 6 muestra que TRB-Powell resuelve los problemas usando menos evaluaciones funcionales en el 84 % de ellos.

En la Tabla 6 se muestran los resultados obtenidos por BOBYQA (BQA) y TRB-Powell (TRB) para los problemas descriptos en la Tabla 1. La primera columna se corresponde con el

Problema		Feval		$f(x_{end})$	
nombre	n	BOBYQA	TRB-Powell	BOBYQA	TRB-Powell
Linear Full Rank	9	31	31	1.699999D+02	1.699999D+02
Linear Rank 1	7	84	82	8.380281D+00	8.380281D+00
Linear Rank 1*	7	90	82	9.880597D+00	9.880597D+00
Rosenbrock	2	171	170	4.720012D-12	4.213517D-09
Powell singular	4	131	83	4.840000D-04	4.840000D-04
Kowalik & Osborne	4	162	90	4.470176D-04	4.470176D-04
Brown Dennis	4	170	168	8.582220D+04	8.582220D+04
HS45	5	44	43	1.000000D+00	1.000000D+00
HS46	5	49	38	4.930380D-32	4.930380D-32
HS56	7	119	98	1.052166D-12	2.148472D-12
HS77	5	140	108	2.130995D-11	2.424337D-11
HS79	5	128	133	8.368490D-13	1.026634D-12
HS107	9	195	158	5.909464D-13	3.809796D-13
HS110	10	206	172	-4.577847D+01	-4.577847D+01
Rastrigin	10	69	68	2.487372D+02	2.487372D+02
Schwefel	10	90	73	-4.189829D+03	-4.189829D+03

* con columnas y filas nulas

TABLA 4. Problemas con restricciones de cota de pequeño porte: BOBYQA vs. TRB-Powell

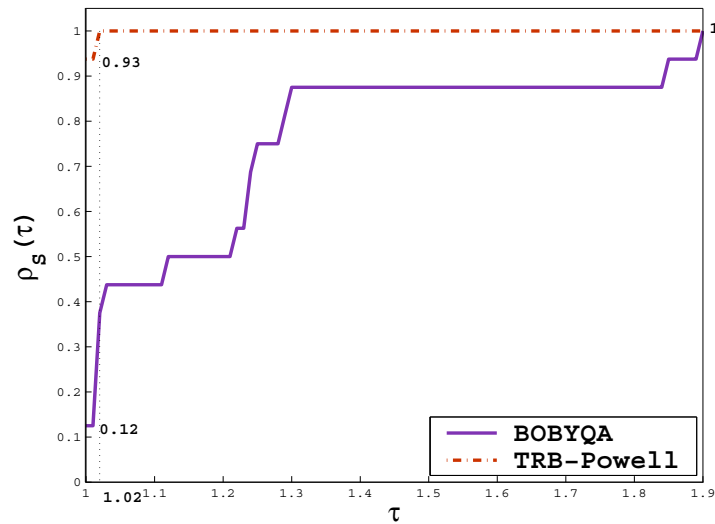


FIGURA 5. Evaluaciones funcionales para problemas con restricciones de cota de pequeño porte.

nombre del problema, las siguientes columnas contienen la cantidad de evaluaciones funcionales requeridas por TRB-Powell y BOBYQA hasta obtener 2, 4, 6 y 8 cifras significativas del valor de

Problema		Feval		$f(x_{end})$	
nombre	n	BOBYQA	TRB-Powell	BOBYQA	TRB-Powell
Invdist2	20	182	236	3.220305D+01	3.220305D+01
	40	943	826	1.677709D+02	1.677709D+02
	60	6990	4832	4.081079D+02	4.087195D+02
	80	5158	6125	7.699010D+02	7.689744D+02
	100	20000	20000	(**) 1.251662D+03	(**) 1.248745D+03
Arwhead	20	636	558	1.826050D-11	4.961897D-11
	40	1458	1468	1.208688D-09	1.054593D-08
	80	3677	4332	8.622840D-10	1.097986D-09
	100	6605	2927	3.969482D-08	2.774640D-09
Chrosen	20	242	235	1.900000D+01	1.900000D+01
	40	464	612	3.900000D+01	3.900000D+01
	80	864	825	7.900000D+01	7.900000D+01
	100	1055	1037	9.900000D+01	9.900000D+01
Penalty 1	20	5677	5253	1.577706D-04	1.577901D-04
	40	15006	11645	3.392511D-04	3.392517D-04
	80	19410	15680	7.130502D-04	7.130518D-04
	100	20000	17458	(**) 9.024911D-04	9.024942D-04
Chebyqad	20	2245	2020	4.572955D-03	4.572956D-03
	40	20000	19617	(**) 5.960849D-03	5.960854D-03
	80	20000	20000	(**) 4.932619D-03	(**) 4.933963D-03
	100	20000	20000	(**) 4.523622D-03	(**) 4.566568D-03
Rastrigin	20	120	114	4.974745D+02	4.974745D+02
	40	1301	297	9.949489D+02	9.949489D+02
	80	600	613	1.989898D+03	1.989898D+03
	100	820	795	2.487372D+03	2.487372D+03
Griewank	20	882	523	6.312884D-11	3.234968D-12
	40	2091	1309	7.525736D-11	1.494938D-11
	80	12436	3031	2.297593D-10	1.809503D-10
	100	5213	3927	2.405037D-10	6.107597D-10
Ackley	20	570	526	9.132573D-06	9.355251D-07
	40	1129	1231	9.987129D-07	1.001365D-06
	80	2112	1923	9.156053D-06	9.086830D-06
	100	2968	2406	8.297680D-07	9.516777D-06

TABLA 5. Problemas con restricciones de cota de medio porte: BOBYQA vs. TRB-Powell

Problema		Feval							
nombre	n	2-dig		4-dig		6-dig		8-dig	
		TRB	BQA	TRB	BQA	TRB	BQA	TRB	BQA
Biggsb1	25	107	144	315	341	389	489	464	548
Bqp1var	1	4	7	5	7	5	7	5	7
Camel6	2	15	17	25	29	29	37	31	41
Chebyqad	4	10	15	18	50	28	60	48	64
Chenhark	10	108	90	124	121	143	151	161	172
Cvxbqp1	10	30	26	30	39	30	39	30	39
Explin2	12	126	224	210	233	219	236	227	258
Hatflda	4	36	46	152	104	167	141	180	182
Hatfldc	25	56	131	87	247	196	360	276	441
HS1	2	102	135	128	158	145	167	158	172
HS110	10	23	144	44	260	82	436	188	521
HS2	2	40	33	44	35	44	39	44	39
HS25	3	97	107	710	734	968	978	f	995
HS3	2	9	6	9	9	9	10	9	10
HS38	4	375	408	431	440	468	474	501	503
HS3mod	2	16	21	21	24	21	24	21	24
HS4	2	6	7	7	7	7	7	7	7
HS45	5	16	16	16	16	16	16	16	16
HS5	2	9	13	12	15	17	18	22	21
Logros	2	82	443	399	609	445	652	547	661
Mccormck	10	40	29	82	54	83	75	99	87
Mdhole	2	218	220	220	220	227	225	228	225
Ncvxbqp1	10	28	31	28	31	28	31	28	31
Ncvxbqp2	10	28	31	28	31	28	31	28	31
Oslbqp	8	20	22	20	22	20	22	20	22
Probpenl	10	f	f	f	f	f	f	f	f
Pspdoc	4	40	41	50	55	57	57	65	65
Qudlin	12	32	34	32	34	32	34	32	34
Simbqp	2	12	12	12	12	12	12	12	12
Sineali	4	12	36	17	260	17	387	17	432

TABLA 6. Resultados de la cantidad de evaluaciones funcionales alcanzadas hasta obtener la precisión de 2, 4, 6 y 8 dígitos en f^* en problemas con restricciones de cota del conjunto test de CUTer.

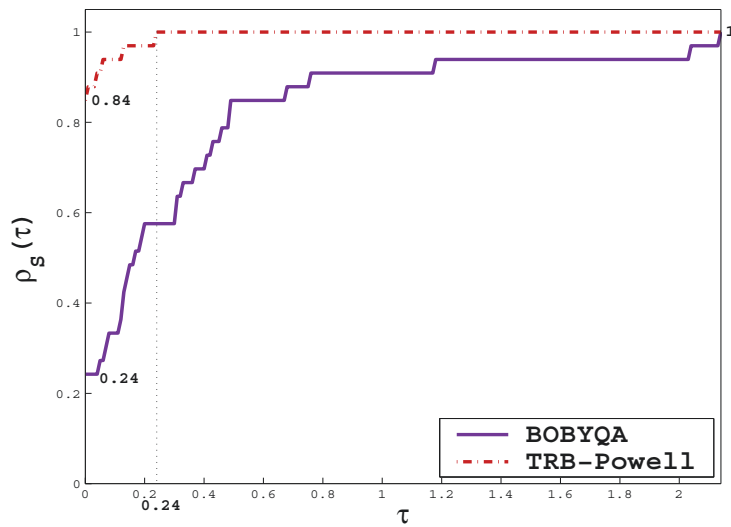


FIGURA 6. Evaluaciones funcionales para problemas con restricciones de cota de medio porte (escala logarítmica).

la función objetivo f^* , reportado por los autores de [55]. Indicamos con “f” cuando el problema no ha logrado alcanzar la precisión deseada.

Los resultados reportados en la Tabla 6 muestran que ambos métodos fallan en tratar de resolver uno de los problemas en los cuatro casos. Más aún, TRB-Powell no alcanzó la precisión de 8 dígitos en el problema HS25. Para ser más exactos, BOBYQA resolvió 17 % de los problemas test más rápido que TRB-Powell, mientras que TRB-Powell lo hizo más rápido en el 73 % de los casos. Para 8 cifras significativas correctas, BOBYQA resolvió el 17 % de los problemas test más rápido y TRB-Powell el 67 %. Para 4 y 6 cifras significativas, BOBYQA resolvió 13 % de los problemas test más rápido que TRB-Powell y TRB-Powell resolvió 67 % y 70 % de los problemas más rápidamente, respectivamente.

Restauración Inexacta para optimización sin derivadas

Los métodos de optimización sin derivadas (DFO) son necesarios para resolver problemas en los cuales las derivadas de la función objetivo y/o de las restricciones no están disponibles para su aplicación en un procedimiento. En este capítulo proponemos un algoritmo en la metodología DFO basado en el esquema de resolución del método de Restauración Inexacta (IR) propuesto por Martínez y Pilotta (2000) [77] y atendiendo a sus posteriores modificaciones [24, 45]. En el algoritmo propuesto los subproblemas son apropiadamente modelizados con el objetivo de no emplear derivadas, usando interpolación polinomial lineal o cuadrática en una cantidad mínima de puntos en los que se conocen los valores funcionales. Se estudian las condiciones del decrecimiento requerido en las dos etapas del algoritmo para obtener convergencia global a puntos estacionarios del problema. Se presentan resultados numéricos obtenidos con una preliminar implementación computacional del método.

Consideramos el problema de optimización no lineal con restricciones

$$(4.1) \quad \text{Min } f(x) \text{ s.a } x \in \Omega, C(x) = 0,$$

siendo $\Omega = \{x \in \mathbb{R}^n \mid L \leq x \leq U, L < U\}$, $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $C : \mathbb{R}^n \rightarrow \mathbb{R}^m$, funciones continuamente diferenciables. Asumimos que las derivadas de f y $C(x)$ no pueden ser calculadas. Como hemos mencionado, esta situación frecuentemente ocurre en problemas donde los valores funcionales de $f(x)$ y $C(x)$ provienen de medidas experimentales o son el resultado de complejos cálculos de simulación. En [12, 35, 42, 92] se describen diferentes situaciones de esta clase de problemas de optimización y el estado de avance en la resolución correspondiente.

En este capítulo proponemos la resolución sin derivadas del problema (4.1) aplicando el esquema algorítmico del método de Restauración Inexacta introducido por Martínez y Pilotta [77], originalmente propuesto con el uso de derivadas. En la literatura existen varios métodos tradicionales [1, 17, 18, 19, 96] y también actuales [52, 54, 75, 76, 77] que usan un procedimiento de resolución en dos fases.

Los pasos característicos de esta clase de metodología son los siguientes:

Paso de Restauración: Dado un iterado actual $x^k \in \mathbb{R}^n$, se calcula un punto y donde una función de mérito, que mide la infactibilidad del problema, decrece en relación al valor en el iterado x^k . El procedimiento algorítmico utilizado para obtener ese punto puede ser elegido de acuerdo al que resulte más conveniente para la estructura de cada problema particular.

Paso de Minimización: Se calcula un punto z en una región de confianza centrada en y , sobre una aproximación lineal al dominio $C(x) = C(y)$, satisfaciendo que el valor de la función que mide la “optimalidad” mejore en z respecto del valor de la misma en y .

Aceptación o rechazo del minimizador: Se define $x^{k+1} = z$ si el punto z , obtenido en el paso previo, es “aceptable” de acuerdo a un criterio que combina factibilidad y optimalidad. Si no es aceptable, se reduce el radio de la región de confianza centrada en y y se repite el Paso de Minimización.

Las dos fases, optimalidad y factibilidad, son totalmente independientes y, con el fin de medir el progreso del proceso, se combinan los valores funcionales de la función objetivo y de las restricciones, evaluadas en el punto de prueba, usando funciones de mérito [76, 77] o filtros [52, 65]. Aquí utilizaremos la función de mérito ψ

$$(4.2) \quad \psi(z, \theta) = \theta f(z) + (1 - \theta)|C(z)|,$$

donde $\theta \in [0, 1]$ es un parámetro de penalidad y $|\cdot|$ es una norma en \mathbb{R}^m , para analizar la aceptación o rechazo de z como el nuevo iterado x^{k+1} .

Como hemos mencionado, utilizaremos el método IR propuesto por Martínez y Pilotta [77]. Este trabajo es posterior a otro publicado por Martínez [76] en el cual se considera la función lagrangiana del problema en vez de la función objetivo $f(x)$. En este trabajo, sin el cálculo de derivadas en los pasos del esquema previo, consideramos sólo el método de [77] ya que usamos modelos y aproximaciones de la función objetivo y de las restricciones.

Para concretar la resolución sin derivadas se utilizan modelos cuadráticos o lineales obtenidos mediante interpolación funcional en $2n + 1$ o $n + 1$ puntos, respectivamente.

Notaciones. $|\cdot|$ indica una norma en \mathbb{R}^m ; $\|\cdot\|$ norma arbitraria en \mathbb{R}^n ; $\|\cdot\|_\infty$, la norma infinito; $\|\cdot\|_2$ la norma Euclídea en \mathbb{R}^n ; $h(x) = \|C(x)\|_2$ representa una medida de la infactibilidad en x ; $C'(y)$ indica el Jacobiano de $C(y)$.

4.1. Método de Restauración Inexacta (IR)

En esta sección describimos el algoritmo presentado en [77] para resolver el problema (4.1). Mencionamos algunos resultados de los autores que son relevantes para nuestra investigación.

Como se dijo antes, el método de Restauración Inexacta introducido por Martínez y Pilotta [77], genera una sucesión $\{x^k\}$ considerando la factibilidad y optimalidad en diferentes fases de una iteración. En la fase de factibilidad, un punto “menos infactible” (con respecto del iterado corriente) se calcula y en la fase de optimalidad, un punto “más cercano a la optimalidad” se obtiene sobre un hiperplano aproximante a la región factible y en una región de confianza conveniente. En este procedimiento, este último punto obtenido es un punto intermedio o tentativo en el cual se debe evaluar una función de mérito. Si su comportamiento es satisfactorio se acepta como el nuevo x^{k+1} . En el caso contrario, se desecha y se repite el paso de minimización en una región de confianza con un radio menor.

Así, la región de confianza se usa para reducir el paso en la fase de optimalidad. Tal región de confianza está centrada en el punto obtenido en el paso de restauración. Por tal motivo, cuando un punto no es aceptado, se repite el paso de minimización después de reducir el radio de la región de confianza. Esta es una característica particular de esta metodología.

4.1.1. Fase de Factibilidad. Dado el iterado $x^k \in \mathbb{R}^n$, el algoritmo calcula un punto y^k que es menos infactible en relación al punto x^k . Tal punto intermedio se obtiene al resolver aproximadamente el problema

$$(4.3) \quad \text{mín} \quad h(y)^2$$

$$(4.4) \quad \text{sujeto a} \quad \|y - x^k\| \leq \beta h(x^k)$$

$$(4.5) \quad h(y) \leq \alpha h(x^k),$$

siendo $0 < \alpha < 1$, $\beta > 0$, parámetros independientes de k .

La primer restricción controla que el paso de restauración no sea significativamente grande e impone que y^k debe ser igual a x^k cuando x^k es factible. La segunda restricción establece la necesidad de obtener un punto al menos tan bueno como el punto x^k con respecto de la factibilidad.

El procedimiento para la obtención del y^k puede fallar, en una particular iteración, dependiendo de las propiedades del algoritmo que se aplique para la resolución de esta fase. Una causa, por la cual puede no encontrarse una solución y^k , es la existencia de un punto estacionario

de $h(x)$, con $C(x) \neq 0$. En tal caso el algoritmo se detiene, declarando falla en obtener la factibilidad.

4.1.2. Fase de Optimalidad. Se asocia a $y^k \in \mathbb{R}^n$ un conjunto $T(y^k)$ el cual es una linealización del conjunto factible,

$$T(y^k) = \{x \in \Omega \mid C'(y^k)(x - y^k) = 0\}.$$

El algoritmo de resolución de esta fase de optimalidad debe hallar, en el conjunto $T(y^k)$, un punto x^{k+1} tal que $f(x^{k+1}) \leq f(y^k)$ y, además, satisfaga un criterio de aceptación.

En el caso del uso de derivadas para calcular el punto en $T(y^k)$ se puede aplicar cualquier algoritmo conveniente que resuelva aproximadamente el problema ([83]):

$$(4.6) \quad \begin{array}{ll} \text{mín} & f(x) \\ \text{sujeto a} & x \in T(y^k) \\ & \|x - y^k\| \leq \delta, \end{array}$$

donde $\delta > 0$ es el radio de la región de confianza.

Notar que, si la norma $\|\cdot\|$ es la norma infinito, el problema que se resuelve en esta fase es un problema de minimización con restricciones lineales

4.1.3. Función de mérito. En el algoritmo propuesto por Martínez y Pilotta [77], el progreso se mide mediante la función de mérito que depende de un parámetro de penalización θ . Este último es usado para dar diferentes pesos a la función objetivo f y a la norma del residuo ($|C(x)|$) de las restricciones del problema original.

La función de mérito propuesta es

$$(4.7) \quad \psi(x, \theta) = \theta f(x) + (1 - \theta)|C(x)|,$$

donde $\theta \in [0, 1]$ es el parámetro de penalización.

En cada iteración se elige el parámetro θ mediante consideraciones prácticas y teóricas [77]. Por ejemplo, si $|C(x)|$ es muy grande, el peso asignado a $f(x)$ debe ser pequeño, pues no es razonable exigirle demasiado a la minimización de f en un punto que está alejado de la región factible.

En el trabajo original de Martínez y Pilotta [77], el parámetro de penalización se adapta usando una estrategia no monótona introducida en [51]. Eso se obtiene mediante la introducción

de un parámetro no negativo ω , permitiendo un moderado incremento del parámetro de penalización entre diferentes iterados. Tal estrategia evita la posibilidad de trabajar con parámetros de penalización demasiado pequeños en relación a los usados al comienzo del proceso iterativo.

Otra consideración a tener en cuenta es que se desea que la evaluación de la función de mérito en un nuevo iterado x^{k+1} , proveniente de la optimización sobre $T(y^k)$, sea menor que en el iterado previo x^k , es decir

$$(4.8) \quad Ared_k := \psi(x^k, \theta_k) - \psi(x^{k+1}, \theta_k) > 0.$$

4.1.4. Condición AGP. Sea $T(z)$ un conjunto cerrado, se define la dirección $d_c(z)$, como la proyección ortogonal de la dirección $-\nabla f(z)$ sobre $T(z)$,

$$(4.9) \quad d_c(z) = P_{T(z)}(z - \nabla f(z)) - z.$$

LLamamos *punto estacionario* del problema (4.1) al punto factible z tal que $d_c(z) = 0$.

Se dice que un punto factible \bar{z} satisface la *condición de optimalidad AGP* si existe una sucesión $\{z^k\}$ que converge a \bar{z} y tal que $d_c(z^k) \rightarrow 0$. Esta condición de optimalidad fue introducida por Martínez y Svaiter [79].

Los puntos que satisfacen la condición de optimalidad AGP se llaman *puntos AGP*. La condición AGP es más fuerte que la condición de Fritz-John, ya que se ha demostrado ([79]) que el conjunto de minimizadores locales de un problema de optimización no lineal está contenido en el conjunto de puntos AGP y éste está contenido en el conjunto de puntos Fritz-John. También, bajo condiciones de calidad en el \bar{z} , la condición AGP es equivalente a la condición de Karush-Kuhn-Tucker (KKT) [9, 79].

En [76] y [77] se demuestra que el método de Restauración Inexacta converge a puntos que satisfacen la condición de optimalidad AGP.

4.1.5. Algoritmo IR. A continuación se describe el Algoritmo IR de Martínez y Pilotta [77]:

ALGORITMO 1. Algoritmo IR

Dados $x^0 \in \mathbb{R}^n$, $\beta > 0$, $\alpha \in (0, 1)$, $\delta_{min} > 0$, $\theta_{-1} \in (0, 1)$, una sucesión $\omega_k > 0$ tal que $\sum_k \omega_k$ sea convergente.

$k \leftarrow 0$.

Iteración k -ésima :

Paso 0: Parámetro de penalización Definir $\bar{\theta}_k = \min\{1, \min\{\theta_{k-1}, \dots, \theta_{-1}\} + \omega_k\}$

Paso I: Fase de Factibilidad Si $h(x^k) = 0$, definir $y^k = x^k$,

sino, calcular y^k tal que satisfaga las condiciones del problema (4.3).

Si no es posible satisfacer las condiciones requeridas,

terminar declarando “*falla en mejorar la factibilidad*”.

Paso II: Fase de Optimización Si $y^k = x^k$ y $d_c(x^k) = 0$,

terminar declarando “*convergencia finita*”.

Definir $\delta = \max(\delta_{min}, 2\delta)$

▪ Repetir mientras el punto x^{k+1} no sea obtenido:

(i) Calcular z^k sobre $T(y^k)$ resolviendo el problema (4.6).

(ii) **Elección del parámetro de penalización** Definir, para todo $\theta \in [0, 1]$,

$$Pred(\theta) = \theta(f(x^k) - f(y^k)) + (1 - \theta)(|C(x^k)| - |C(y^k)|).$$

◦ Calcular θ_k^s , el valor máximo de $\theta \in [0, \bar{\theta}_k]$ tal que

$$Pred(\theta_k^s) \geq \frac{1}{2}(|C(x^k)| - |C(y^k)|)$$

(iii) **Criterio para aceptar x^{k+1}** Definir

$$Ared_k = \psi(x^k, \theta_k^s) - \psi(x^{k+1}, \theta_k^s), \quad Pred_k = Pred(\theta_k^s)$$

◦ Si $Ared_k < 0.1Pred_k$, adaptar $\delta = 0.5\delta$ y $\bar{\theta}_k = \theta_k^s$, ir al paso (i).

◦ Si $Ared_k \geq 0.1Pred_k$, definir $x^{k+1} = z^k$, $\theta_k = \theta_k^s$,

$k \leftarrow k + 1$.

Fin k - iteración.

4.2. Hipótesis generales y resultados preliminares para optimización sin derivadas

4.2.1. Hipótesis generales. Asumimos que valen las siguientes hipótesis.

(A1) Ω es convexo y compacto.

(A2) La matriz Jacobiana de $C(x)$, $C'(x)$, existe y satisface la condición de Lipschitz:

$$(4.10) \quad \|C'(y) - C'(x)\| \leq L_1 \|y - x\|, \quad \text{para todo } x, y \in \Omega.$$

También,

$$(4.11) \quad \|\nabla C_j(y) - \nabla C_j(x)\| \leq \bar{L}_j \|y - x\|, \quad \text{para todo } x, y \in \Omega,$$

si $C_j(x)$ es la j -ésima restricción $C_j(x) = 0$, para $j = 1, \dots, m$.

(A3) El gradiente de f existe y satisface la condición de Lipschitz:

$$(4.12) \quad \|\nabla f(y) - \nabla f(x)\| \leq L_2 \|y - x\|, \text{ para todo } x, y \in \Omega.$$

(A4) Asumimos que cada conjunto de puntos interpolantes $Y = \{y, z^1, \dots, z^n\} \subset \mathbb{R}^n$, contenido en un entorno $B(y, \Delta(Y))$ con radio $\Delta(Y) = \max_{1 \leq i \leq n} \|z^i - y\|$, es “equilibrado” para construir un modelo de interpolación lineal. Se asume que satisface las condiciones propuestas en ([35]).

La hipótesis A4 nos permite obtener resultados de acotación (2.11 y 2.12 en el Capítulo 2) para las magnitudes que miden la diferencia entre el comportamiento de las funciones y la de los modelos aproximantes obtenidos usando el conjunto de interpolación Y .

Sean $L(x) = f(y) + gf^T(x - y)$ y $m_c(x) = |C(y)| + g_c^T(x - y)$ las aproximaciones lineales de $f(x)$ y $|C(x)|$, respectivamente. Luego, para $x \in B(y, \Delta(Y))$, se cumple:

$$(4.13) \quad |f(x) - L(x)| \leq L_2 \kappa_{eg} \Delta^2,$$

$$(4.14) \quad ||C(x)| - m_c(x)| \leq L_1 \kappa_{eg} \Delta^2,$$

$$(4.15) \quad \|\nabla f(x) - gf\| \leq L_2 \kappa_{eg} \Delta,$$

siendo $\kappa_{eg} = (1 + n^{0,5} \|\hat{M}^{-1}\|/2)$, $\hat{M} = M/\Delta$, $M_{ij} = y_j^i - y_j$ y $\|\hat{M}^{-1}\| < c$ para algún $c > 0$ ([35]).

Otra propiedad que usaremos en este trabajo, relacionando los gradientes exactos con los gradientes simplex calculados mediante el conjunto Y :

$$(4.16) \quad \|C'(x) - A(y)\| \leq L_1 \kappa_{eg} \Delta,$$

para $x \in B(y, \Delta(Y))$, si $A(y)$ es la matriz cuyas filas son los gradientes simplex, calculados en y , de las restricciones $C_j(x)$, para $j = 1, \dots, m$.

Remarcamos las siguientes propiedades provenientes de las hipótesis de Lipschitz continuidad asumidas en (4.10) y (4.12):

$$(4.17) \quad |C(x) - C(y) - C'(y)(x - y)| \leq \frac{L_1}{2} \|y - x\|^2,$$

$$(4.18) \quad |f(x) - f(y) - \nabla f(y)^T(x - y)| \leq \frac{L_2}{2} \|y - x\|^2.$$

4.2.2. Resultados preliminares. En el presente trabajo, para resolver los problemas (4.3) y (4.6) del método IR, reemplazamos las derivadas exactas por el cálculo de las derivadas simplex.

En particular, para aproximar al hiperplano $C'(y)(x - y) = 0$, en la fase de optimalidad, se obtiene una aproximación del Jacobiano $C'(y)$ calculando las derivadas simplex de C_j , $j = 1, \dots, m$, mediante interpolación polinomial de primer grado usando un conjunto de puntos $Y = \{y, z^1, z^2, \dots, z^n\}$, tal que $\Delta = \max\{\|z^i - y\| : i = 1, \dots, n\}$. Como antes, denominamos $A(y)$ a la matriz obtenida.

DEFINICIÓN 4.1. Dado y , $A(y)$ y el hiperplano $A(y)(x - y) = 0$, denominamos

$$(4.19) \quad \pi := \{x \in \Omega \mid A(y)(x - y) = 0\}.$$

Notación: En particular, en cada punto intermedio y^k proveniente del Paso de Restauración del algoritmo, se obtendrá un hiperplano aproximado al $C'(y^k)(x - y^k) = 0$.

$$(4.20) \quad \pi_k = \{x \in \Omega \mid A(y^k)(x - y^k) = 0\}.$$

En el trabajo de Martínez y Pilotta [77], usando las hipótesis (A1)-(A3), se establece una cota para el deterioro de la factibilidad sobre el hiperplano $C'(y)(x - y) = 0$, comparando la factibilidad en x respecto de la factibilidad en y . Tal resultado muestra que existe $L_1 > 0$ tal que $|C(x)| \leq |C(y)| + L_1\|x - y\|^2$, y prueba que el deterioro de la factibilidad sobre el hiperplano tangente es de segundo orden respecto de la distancia entre los puntos x e y .

En nuestro caso, calculando $A(y)$ como una aproximación del $C'(y)$, el comportamiento del deterioro de la factibilidad sobre $A(y)(x - y) = 0$, es de orden 1 y dependiente del radio Δ del conjunto de puntos interpolantes.

TEOREMA 8. Dado el sistema de ecuaciones no lineales $C(x) = 0$ del problema (4.1). Existe $L_1 > 0$ (independiente de k) tal que, si $y \in \Omega$ es el resultado de la fase de factibilidad, $x \in \pi$ entonces

$$|C(x)| \leq |C(y)| + L_1\kappa_{eg}\Delta\|x - y\| + L_1\|x - y\|^2.$$

Demostración. Como $x \in \pi$, $A(y)(x - y) = 0$, siendo $A(y)$ la matriz cuyas filas son los gradientes simplex de las restricciones $C_j(x)$, $j = 1, \dots, m$. Se asume que tales cálculos fueron realizados sobre un conjunto de puntos de interpolación con las características dadas en A4.

Luego, considerando (4.17),

$$|C(x)| \leq |C(y)| + |C'(y)(x - y)| + L_1 \|x - y\|^2.$$

Usando la desigualdad (4.16), se tiene

$$\begin{aligned} |C(x)| &\leq |C(y)| + \|C'(y) - A(y)\| \|x - y\| + L_1 \|x - y\|^2 \\ &\leq |C(y)| + L_1 \kappa_{eg} \Delta \|x - y\| + L_1 \|x - y\|^2. \end{aligned}$$

Obteniéndose así el resultado deseado. □

OBSERVACIÓN 4.2.1. Cuando Δ tiende a 0 el deterioro de la factibilidad tiende a comportarse como en el caso del verdadero hiperplano tangente.

Los siguientes resultados son necesarios para establecer el Algoritmo 3 IR-DFO y obtener los resultados de convergencia del mismo.

Dada una aproximación lineal de la función objetivo f del problema (4.1) en y , $L(x) = f(y) + gf^T(x - y)$, el hiperplano $A(y)(x - y) = 0$, el conjunto π definido en (4.19), denotamos P_π al proyector ortogonal sobre π .

Entonces la proyección de $-gf$ sobre π se define mediante:

DEFINICIÓN 4.2. Dado el vector $-gf \in \mathbb{R}^n$, y P_π , se denomina

$$(4.21) \quad gf_{tan} = P_\pi(y - gf) - y.$$

4.2.3. Propiedades de gf_{tan} .

PROPIEDAD 4.2.1. $\langle gf_{tan}, gf \rangle \leq -\frac{\|gf_{tan}\|_2^2}{2}$

Demostración. Como $y \in \pi$, $P_{\pi^\perp}(gf) = (gf) - P_\pi(gf)$. Sea α el ángulo comprendido ente π^\perp y gf ,

$$\begin{aligned} \|(y - gf) - P_\pi(y - gf)\|_2^2 &= \|(y - gf) - P_\pi(y) + P_\pi(gf)\|_2^2 \\ &= \|P_\pi(gf) - gf\|_2^2 \\ &= \|gf - P_\pi(gf)\|_2^2 \\ &= \|P_{\pi^\perp}(gf)\|_2^2 \end{aligned}$$

$$\begin{aligned}
&= \|gf\|_2^2 \cos^2(\alpha) \\
&\leq \|gf\|_2^2.
\end{aligned}$$

Por otro lado,

$$\begin{aligned}
\|(y - gf) - P_\pi(y - gf)\|_2^2 &= \|(-y + P_\pi(y - gf)) + gf\|_2^2 \\
&= \|P_\pi(y - gf) - y\|_2^2 + \|gf\|_2^2 + 2\langle P_\pi(y - gf) - y, gf \rangle \\
&\leq \|gf\|_2^2.
\end{aligned}$$

Entonces,

$$\langle P_\pi(y - gf) - y, gf \rangle \leq -\frac{\|P_\pi(y - gf) - y\|_2^2}{2} = -\frac{\|gf_{tan}\|_2^2}{2}.$$

Luego,

$$\langle gf_{tan}, gf \rangle \leq -\frac{\|gf_{tan}\|_2^2}{2}.$$

□

Observación: En los desarrollos de esta sección se usará la siguiente relación entre normas:

$$\|x\|_\infty \leq \|x\|_2 \leq \sqrt{n}\|x\|_\infty, \text{ si } x \in \mathbb{R}^n.$$

La siguiente propiedad indica que, si el modelo $L(x)$ es una buena aproximación lineal de la función objetivo f , la norma de la proyección de $-\nabla f$ es pequeña si y sólo si la norma de la proyección del gradiente del modelo, gf_{tan} , lo es.

PROPIEDAD 4.2.2. Sea $\epsilon_1 > 0$ tal que $\|\nabla f - gf\|_2 < \frac{\epsilon_1}{2}$.

Si $\|y - P_\pi(y - \nabla f)\|_\infty > \epsilon_1$ entonces $\|gf_{tan}\|_\infty > \frac{\epsilon_1}{2}$. Análogamente, si $\|gf_{tan}\| > \epsilon_1$ entonces $\|y - P_\pi(y - \nabla f)\|_\infty > \frac{\epsilon_1}{2}$.

Demostración. Primero demostraremos la primer implicación, $\|gf_{tan}\|_\infty > \frac{\epsilon_1}{2}$ siempre que $\|y - P_\pi(y - \nabla f)\|_\infty > \epsilon_1$.

Para ello, observamos que

$$(4.22) \quad \|y - P_\pi(y - \nabla f)\|_\infty = \|y - P_\pi(y - \nabla f - gf + gf)\|_\infty$$

$$(4.23) \quad \leq \|y - P_\pi(y - gf)\|_\infty + \|\nabla f - gf\|_\infty$$

$$(4.24) \quad \leq \|y - P_\pi(y - gf)\|_\infty + \|\nabla f - gf\|_2,$$

donde gf es el gradiente del modelo lineal $L(x)$. Luego,

$$\begin{aligned}\|gf_{tan}\|_{\infty} &= \|y - P_{\pi}(y - gf)\|_{\infty} \\ &\geq \|y - P_{\pi}(y - \nabla f)\|_{\infty} - \|\nabla f - gf\|_2 \\ &> \epsilon_1 - \frac{\epsilon_1}{2} \\ &= \frac{\epsilon_1}{2},\end{aligned}$$

como se quería demostrar.

Análogamente se puede ver que

$$\begin{aligned}\|y - P_{\pi}(y - gf)\|_{\infty} &= \|y - P_{\pi}(y - gf - \nabla f + \nabla f)\|_{\infty} \\ &\leq \|y - P_{\pi}(y - \nabla f)\|_{\infty} + \|\nabla f - gf\|_{\infty} \\ &\leq \|y - P_{\pi}(y - \nabla f)\|_{\infty} + \|\nabla f - gf\|_2\end{aligned}$$

Luego,

$$\begin{aligned}\|y - P_{\pi}(y - \nabla f)\|_{\infty} &\geq \|y - P_{\pi}(y - gf)\|_{\infty} - \|\nabla f - gf\|_2 \\ &> \epsilon_1 - \frac{\epsilon_1}{2} = \frac{\epsilon_1}{2},\end{aligned}$$

como se quería demostrar. □

4.2.4. ¿Cuándo gf_{tan} es una dirección de descenso de f ?

Analizamos cuándo se puede asegurar que la dirección gf_{tan} , la proyección del gradiente negativo del modelo ($-gf$) sobre la aproximación tangente π , es una dirección de descenso de la función objetivo f .

Es importante analizar este tema para conocer si usando el gradiente del modelo se puede obtener una dirección de descenso de f .

LEMA 4.2.1. Sea $\|gf_{tan}\|_{\infty} > \epsilon_1$, $\epsilon_1 > 0$. Si $\|\nabla f - gf\|_2 < \frac{\epsilon_1}{4}$ entonces $\langle gf_{tan}, \nabla f \rangle < 0$. En particular, $\langle gf_{tan}, \nabla f \rangle < -\frac{1}{4}\|gf_{tan}\|_2^2$.

Demostración. Calculamos

$$\begin{aligned}\langle gf_{tan}, \nabla f \rangle &= \langle gf_{tan}, \nabla f + gf - gf \rangle \\ &= \langle gf_{tan}, \nabla f - gf \rangle + \langle gf_{tan}, gf \rangle\end{aligned}$$

$$\begin{aligned}
&\leq \|gf_{tan}\|_2 \|\nabla f - gf\|_2 + \langle gf_{tan}, gf \rangle \\
&< \|gf_{tan}\|_2^2 \left(\frac{\|\nabla f - gf\|_2}{\|gf_{tan}\|_2} - \frac{1}{2} \right) \\
&< \|gf_{tan}\|_2^2 \left(\frac{1}{4} - \frac{1}{2} \right) \\
&< -\frac{1}{4} \|gf_{tan}\|_2^2.
\end{aligned}$$

Luego, como $\|gf_{tan}\|_\infty > \epsilon_1$ (también lo es la norma 2), se obtiene $\langle gf_{tan}, \nabla f \rangle < 0$, cuando la aproximación entre ∇f y gf es buena. Por lo tanto, gf_{tan} es una dirección de descenso para f .

□

PROPIEDAD 4.2.3. Sea $\epsilon_1 > 0$, $y \in \pi$ y el modelo $L(x) = f(y) + gf^T(x - y)$ que interpola a $f(x)$ en un entorno de y . Si $\|y - P_\pi(y - gf)\| < \frac{\epsilon_1}{2}$ y $\|\nabla f(y) - gf\|_2 < \frac{\epsilon_1}{2}$ entonces $\|P_\pi(y - \nabla f(y)) - y\| < \epsilon_1$.

Demostración.

$$\begin{aligned}
\|P_\pi(y - \nabla f(y)) - y\|_\infty &= \|y - P_\pi(y - \nabla f(y) - gf + gf)\|_\infty \\
&\leq \|y - P_\pi(y - gf)\|_\infty + \|\nabla f(y) - gf\|_2 < \frac{\epsilon_1}{2} + \frac{\epsilon_1}{2} = \epsilon_1.
\end{aligned}$$

□

OBSERVACIÓN 4.2.2. Decimos que el modelo $L(x)$ es *bueno* si éste es una buena aproximación lineal de la función objetivo f .

Si el modelo $L(x)$ es bueno y $\|gf_{tan}\|_\infty \geq \epsilon_1$, es decir, suficientemente grande, se puede asegurar que en la dirección de gf_{tan} la función objetivo desciende.

Por otra parte, cuando $\|gf_{tan}\| < \epsilon_1$, pequeño, y el modelo es bueno, se sabe que la proyección de $-\nabla f$ también tiene una magnitud pequeña, por ello no es posible asegurar que f desciende significativamente en esa dirección.

El Algoritmo 3 IR-DFO que presentamos en la sección siguiente incluye, cuando no es posible asegurar que el modelo se ajusta suficientemente bien a la función f en un iterado y^k o cuando la magnitud de gf_{tan} no permite asegurar el descenso de la función f , el estudio del descenso de la función objetivo sobre un conjunto de direcciones D_k en π_k . A continuación se describen tales direcciones.

4.2.5. Base positiva de un espacio. Conjunto generador positivo.

La idea de buscar una base positiva $D \subset \mathbb{R}^n$ es la de garantizar que, dado un vector no nulo $v \in \mathbb{R}^n$, existe al menos un vector $d \in D$ tal que $v^T d < 0$. Si el vector no nulo es gf_{tan} , la proyección del gradiente negativo ($-gf$) del modelo lineal (L) de la función objetivo (f), cualquier dirección $d \in D$ que forme un ángulo agudo con gf_{tan} es una dirección de descenso, al menos en un entorno del punto base para la linealización. Por lo que, en un entorno del punto base, existe $\alpha > 0$ y $d \in D$ tal que $f(y + \alpha d) < f(y)$.

Seguiremos las notaciones del libro [35].

Se llama *conjunto generador positivo* de vectores $[v_1, \dots, v_r] \subset \mathbb{R}^n$ al cono convexo

$$\{v \in \mathbb{R}^n : v = \sum_{i=1}^r \alpha_i v_i, \alpha_i > 0, i = 1, \dots, r\}.$$

Luego, un *conjunto generador positivo* de vectores en \mathbb{R}^n es un conjunto que genera \mathbb{R}^n positivamente. El conjunto $[v_1, \dots, v_r]$ se dice *positivamente dependiente* si uno de los vectores es combinación positiva de los otros; caso contrario, el conjunto es *positivamente independiente*. Una *base positiva* en \mathbb{R}^n es un conjunto positivamente independiente que genera positivamente a \mathbb{R}^n .

Si $[v_1, \dots, v_r]$ genera \mathbb{R}^n positivamente, entonces este contiene un subconjunto de $r - 1$ elementos que genera \mathbb{R}^n (Teorema 2.2, [35]) pero una base positiva de \mathbb{R}^n no puede contener un subconjunto propio positivamente generador de \mathbb{R}^n . Una base positiva tiene un número de elementos entre $n + 1$ y $2n$ y esos números se refieren a bases positivas minimal y maximal, respectivamente.

Propiedades y ejemplos de bases positivas de \mathbb{R}^n se pueden encontrar [6, 38, 40, 66].

Algunos ejemplos de bases positivas en \mathbb{R}^n son $[I \quad -e]$ y $[I \quad -I]$, donde e es un vector de “unos”, y la matriz I es la matriz identidad en \mathbb{R}^n .

Consideremos, en la iteración k -ésima del Algoritmo 3 IR-DFO, el hiperplano tangente en y^k , $A_k(x - y^k) = 0$. Notemos $\mathcal{N}(A_k)$ al subespacio anulador de A_k en \mathbb{R}^n y sea B_k una base positiva de $\mathcal{N}(A_k)$. Observar que, como A_k es una matriz de $\mathbb{R}^{m \times n}$ de rango máximo con $m < n$ (o sea, de rango m), entonces B_k es una matriz de $n \times |B_k|$, donde $|B_k|$ es la dimensión de B_k .

Si en el iterado y^k , las restricciones de cotas no son activas, es posible considerar una base positiva minimal de $\mathcal{N}(A_k)$ relacionada a $[I \quad -e]$. En nuestro caso, si Z_k es una base de $\mathcal{N}(A_k)$ consideramos $B_k = Z_k[I \quad -e]$.

Consideremos ahora el caso en que hay cotas activas en y^k , es decir, se tiene $y_i^k = U_i$, o $y_i^k = L_i$, para alguna i -componente de y^k . En este caso, se debe construir un conjunto generador positivo D_k de los gradientes de las restricciones activas en y^k .

En tal situación, una posibilidad para estudiar el decrecimiento de f en y^k es considerar un problema equivalente en \mathbb{R}^{n-m} , haciendo un cambio de variables.

Considerando $f(y^k + (y - y^k))$, $y - y^k \in \mathcal{N}(A_k)$, Z_k una base ortogonal de $\mathcal{N}(A_k)$, y haciendo $Z_k w = y - y^k$ obtenemos el problema equivalente:

$$(4.25) \quad \text{mín } f(y^k + Z_k w) = g(w), \quad w \in \mathbb{R}^{n-m},$$

con las restricciones, $Z_k w \leq U - y^k$, $-Z_k w \leq y^k - L$.

Así, si en y^k existe alguna componente $(y^k)_i = U_i$, o $(y^k)_j = L_j$, se genera el cono polar de $(Z_k)_i w \leq U_i - y_i^k$, $-(Z_k)_j w \leq y_j^k - L_j$, en el espacio \mathbb{R}^{m-n} .

En la literatura hay distintos procedimientos para construir el conjunto de direcciones D_k [5, 72] en los casos particulares antes citados.

En el software SID-PSM [37, 39], que resuelve problemas con restricciones de desigualdades ($g_i(x) \leq 0$), la función *gen.m* calcula un conjunto generador positivo de \mathbb{R}^n , cuando el problema es irrestricto o cuando tiene sólo restricciones de cotas, o cuando en el punto analizado no hay restricciones casi activas, es decir no ocurre $g_i(x) \simeq 0$, para las restricciones del problema. En otro caso, cuando hay restricciones de desigualdad activas, SID-PSM, calcula un conjunto de generadores positivos para el cono tangente de los gradientes de las restricciones aproximadamente activas. La implementación se hace usando la descomposición en valores singulares (SVD) de la matriz de los apropiados gradientes, siguiendo el algoritmo descrito en [72].

En [72] Lewis y Torczon proponen un algoritmo, basado en el uso de la factorización LU, para construir el conjunto de generadores en el caso no degenerado. El caso degenerado no fue tratado por los autores en ese trabajo.

En el trabajo [5] Abramson et al. estudian en especial el caso de restricciones activas redundantes (degenerado). Desarrollan un algoritmo para obtener el conjunto generador del cono polar del cono de los gradientes de las restricciones activas para el caso de desigualdades ($g_i(x) \leq 0$). Para construir el conjunto de generadores, identifican las restricciones activas redundantes y activas no redundantes mediante la metodología de proyecciones y usan la descomposición QR o LU. Ellos dan un algoritmo para determinar el conjunto de generadores y discuten detalles de la implementación para obtener las direcciones [5].

En este trabajo utilizamos la idea de Lewis y Torczon [72] para el caso no degenerado y el de Abramson et al [5] para el caso degenerado.

4.3. Algoritmo de Restauración Inexacta sin derivadas

En esta sección presentamos el algoritmo de Restauración Inexacta sin derivadas para resolver el problema (4.1).

ALGORITMO 3. IR-DFO

Dados $0 < \epsilon_f < 1$, $0 < \epsilon_M < 1$. Dado $\{r_k\} > 0$, $\lim_{k \rightarrow \infty} r_k = 0$, $\alpha \in (0, 1)$, $\beta > 1$, $x^0 \in \Omega$, $f(x^0)$, $h(x^0) = |C(x^0)|$, $\delta_{min} > 0$, $\{\omega_k\} > 0$, $\sum_k^\infty \omega_k < \infty$, $\theta_{-1} \in [0, 1]$.

$k \leftarrow 0$

Iteración k -ésima: Para obtener x_{k+1} , se hacen los siguientes pasos:

Paso 0: Definir $\theta_{k,-1} = \min\{1, \min\{\theta_{-1}, \dots, \theta_{k-1}\} + \omega_k\}$.

Paso I: Fase de Factibilidad Dado $x^k \in \Omega$, $r_k \in [0, 1]$.

Calcular $y^k \in \Omega$, resolviendo en forma aproximada el problema

$$(4.26) \quad \min_{y \in \Omega} h(y)^2 \quad \text{sujeto a} \quad \|y - x^k\| \leq \beta h(x^k)$$

sin usar derivadas, satisfaciendo la condición $h(y^k) \leq \alpha h(x^k)$.

- Si no es posible obtener tal y^k , *terminar declarando falla al mejorar la factibilidad:*

FIN.

Paso II: Cálculo aproximado del Jacobiano en y^k :

Paso II.a Considerar un conjunto de puntos interpolantes centrados en y^k :

$Y^k = \{y^k\} \cup \{z^1, z^2, \dots, z^n\}$, tal que $\Delta_k = \max\{\|z^i - y^k\|\}$ satisfaga

$$(4.27) \quad \Delta_k \leq \min\{r_k, \beta_k\},$$

siendo $\beta_k = \beta h(x^k)$ si $h(x^k) \neq 0$, si no $\beta_k = r_k$.

Paso II.b Generar A_k , aproximación del Jacobiano de $C(x)$ en y^k . Calcular Z_k , base del $\mathcal{N}(A_k)$. Definir $\pi_k = \{z \in \Omega \mid A_k(z - y^k) = 0\}$.

Paso III: Fase de Optimización:

Paso III.a Dado $\delta_{k,0} \geq \delta_{min}$, $i \leftarrow 0$.

Resolver aproximadamente el problema (4.6), reemplazando $T(y^k)$ por π_k , con $\|z - y^k\| \leq \delta_{k,0}$, mediante un algoritmo sin derivadas, siguiendo el esquema del Algoritmo

Básico 4 el cual se encuentra más adelante.

- Si $z^{k,0} = y^k$ ($gf_{tan} = P_k(y^k - gf) - y^k$ suficientemente pequeño), definir $x_{k+1} = y^k$, $d_k = gf_{tan}$, $iacc(k) = 0$.
 $k \leftarrow k + 1$. Ir al Paso 0.
- Si $z^{k,0} \neq y^k$ ($f(z^{k,0}) < f(y^k)$), definir $d_{k,0} = gf_{tan}$.

Paso III.b Mientras (x_{k+1} no esté definido) repetir:

Si ($i > 0$), calcular $z^{k,i} \in \pi_k$ siguiendo el esquema del Algoritmo Básico 4, satisfaciendo $\|z^{k,i} - y^k\| \leq \delta_{k,i}$, tal que $f(z^{k,i}) < f(y^k)$.

- **Elección del parámetro de penalización** Definir

$$Pred_{k,i}(\theta) = \theta[f(x^k) - f(z^{k,i})] + (1 - \theta)[|C(x^k)| - |C(y^k)|].$$

Asignar $\theta_{k,i}$ al supremo de los valores de θ en el intervalo $[0, \theta_{k,i-1}]$ que satisfacen:

$$(4.28) \quad Pred_{k,i}(\theta_{k,i}) \geq \frac{1}{2}[|C(x^k)| - |C(y^k)|]$$

- **Criterio de aceptación del punto** $z^{k,i}$ Definir

$$Ared_{k,i}(\theta_{k,i}) = \theta_{k,i}[f(x^k) - f(z^{k,i})] + (1 - \theta_{k,i})[|C(x^k)| - |C(z^{k,i})|].$$

- Si $Ared_{k,i}(\theta_{k,i}) < 0.1 Pred_{k,i}(\theta_{k,i})$:
 - cuando x^k es factible ($C(x^k) = 0$) y $\delta_{k,i} \leq \min\{\epsilon_M, r_k\}$,
definir $x_{k+1} = y^k$, $\theta_k = \theta_{k,i}$, $iacc(k) = i$, $d_k = gf_{tan}$;
 - en otros casos, adaptar $\delta_{k,i+1} \in [0.1\delta_{k,i}, 0.9\delta_{k,i}]$, adaptar $i \leftarrow i + 1$.
- Si $Ared_{k,i}(\theta_{k,i}) \geq 0.1 Pred_{k,i}(\theta_{k,i})$,
definir $x_{k+1} = z^{k,i}$, $\theta_k = \theta_{k,i}$, $iacc(k) = i$, $d_k = gf_{tan}$.

$k \leftarrow k + 1$.

Fin k -ésima iteración.

OBSERVACIÓN 4.3.1. La fase de factibilidad (Paso I), del algoritmo previo, está definida por las condiciones (4.5) y (4.4) como en el trabajo [77]. Notar que no interviene la función objetivo en tales condiciones.

Un aspecto atractivo de los algoritmos del tipo IR es que no es necesario establecer con precisión el método interno ha utilizar, es decir, se tiene libertad de disponer de diferentes procedimientos para los pasos de Restauración y de Minimización. En nuestro caso, se pueden utilizar distintos algoritmos sin derivadas para obtener el decrecimiento requerido de la infactibilidad. En este trabajo se darán mayores precisiones en la sección de experiencias numéricas.

El cálculo del Paso II, aproximación del $C'(y^k)$, se hace usando derivadas simplex en un conjunto interpolante siguiendo las definiciones descriptas en el Capítulo 2.

El Paso III requiere la resolución aproximada de un problema de optimización con restricciones lineales. La idea básica de tal paso es la minimización de la función f en una región de confianza de radio δ , sujeta a la restricción lineal, con el objetivo de no perder una magnitud significativa de la factibilidad lograda en el Paso I. A continuación damos un esquema de resolución del Paso III.

El procedimiento para adaptar el parámetro de penalización θ en la función de mérito es similar al empleado en [77].

Parámetro de penalización y Función de mérito. En el Paso III. b del Algoritmo 3 IR-DFO se compara el punto $z^{k,i}$ con el punto x^k usando una función de mérito que involucra un parámetro de penalización. Este último es usado para dar diferentes pesos a la función objetivo y a la norma de las restricciones del problema original, tal como se usa en [77]. La función de mérito que usamos es

$$(4.29) \quad \psi(x, \theta) = \theta f(x) + (1 - \theta)|C(x)|$$

donde $\theta \in [0, 1]$ es el parámetro de penalización.

En cada iteración se elige el parámetro $\theta_{k,i}$ mediante consideraciones prácticas y teóricas [77].

Una consideración a tener en cuenta es que se desea que la evaluación de la función de mérito en el nuevo punto $z^{k,i}$ sea menor que en el iterado actual x^k , es decir

$$(4.30) \quad Ared_{k,i} := \psi(x^k, \theta_{k,i}) - \psi(z^{k,i}, \theta_{k,i}) > 0,$$

siendo $Ared_{k,i}$ la reducción verdadera o actual. Una reducción pequeña en la función de mérito no es “suficiente” para garantizar convergencia. Por tanto, se desea una reducción “suficiente” de la función de mérito, la que es evaluada mediante el siguiente test:

$$(4.31) \quad Ared_{k,i} \geq 0.1Pred_{k,i},$$

donde $Pred_{k,i}$ es la predicción de la reducción de la función de mérito entre x^k y $z^{k,i}$,

$$(4.32) \quad Pred_{k,i} = \theta_{k,i}[f(x^k) - f(z^{k,i})] + (1 - \theta_{k,i})[|C(x^k)| - |C(y^k)|].$$

Se observa que $Pred_{k,i}$ puede ser no positiva, dependiendo del valor de $\theta_{k,i}$. Si $\theta_{k,i}$ es pequeño, entonces el valor que toma $Pred_{k,i}$ es muy cercano al de $|C(x^k)| - |C(y^k)|$ el cual es no negativo.

Por tanto, se elegirá como parámetro de penalización un $\theta_{k,i} \in [0, 1]$ tal que

$$(4.33) \quad \text{Pred}_{k,i} \geq \frac{1}{2} [|C(x^k)| - |C(y^k)|].$$

Luego, cuando se satisface el criterio (4.31), se acepta el punto $x^{k+1} = z^{k,i}$. Si no se obtiene la reducción suficiente estipulada, se reduce el radio $\delta_{k,i}$ de la región de confianza y se calcula un nuevo punto $z^{k,i}$ con el objetivo de mejorar la relación con $\text{Pred}_{k,i}$.

ALGORITMO 4. Algoritmo Básico del Paso III (Minimización)

Dados $\pi_k, y^k, x^k, f(y^k), A_k, Z_k$ base ortogonal del $\mathcal{N}(A_k)$, $Y^k = \{y^k\} \cup \{z_1, z_2, \dots, z_n\}$, B_k base positiva de $\mathcal{N}(A_k)$, D_k conjunto generador positivo, $i \geq 0$, $\delta_{k,i} > 0$, $0 < \gamma_2 < 1$, $0 < \eta_1 < 1$.

Paso 1: Si ($i = 0$), calcular $L(w) = f(y^k) + gf^T(w - y^k)$ por interpolación en Y^k cuyo máximo radio es menor a r_k .

Calcular $gf_{tan} = P_{\pi_k}(y^k - gf) - y^k$.

Paso 1.a Si $\|gf_{tan}\| \neq 0$, calcular $t_{max}^{k,0} = \min\{1, \delta_{k,0}/\|gf_{tan}\|\}$.

-Analizar si $f(y^k + tgf_{tan}) < f(y^k)$, para algún $t = t_{max}^{k,0}/2^j$, $j = 0, 1, 2, \dots$, hasta cierta tolerancia.

Si existe tal t , definir $z^{k,0} = y^k + tgf_{tan}$ y $d_{k,0} = gf_{tan}$. **Retornar.**

Paso 1.b Caso contrario, definir $\tau_{max} = \min_{d_j \in D_k} \{1, \delta_{k,0}/\|d_j\|\}$, $\nu = 0$.

- Definir $\tau_\nu = \tau_{max}/2^\nu$, analizar si existe un $d_j \in D_k$ tal que $f(y^k + \tau_\nu d_j) < f(y^k)$.

Si existe d_j , tal que f decrece, definir $z^{k,0} = y^k + \tau_\nu d_j$, adaptar Y^k y gf_{tan} usando las nuevas evaluaciones y $d_{k,0} = gf_{tan}$. **Retornar.**

Sino, adaptar $\tau_{\nu+1} = \tau_{max}/2^{\nu+1}$, y repetir la búsqueda para $\nu = \nu + 1$, hasta cierta tolerancia.

Paso 1.c Si no hay éxito en el Paso 1.a y Paso 1.b, definir $z^{k,0} = y^k$, $d_{k,0} = gf_{tan}$. **Retornar.**

Paso 2: Si ($i > 0$) (se conoce el gf , calculado en $i = 0$, y el decrecimiento de f en $z^{k,0}$ y $d_{k,0} = gf_{tan}$)

Paso 2.a Calcular $t_{max}^{k,i} = \min\{1, \delta_{k,i}/\|gf_{tan}\|\}$.

-Determinar el mayor $t \in (0, t_{max}^{k,i}]$ tal que $f(y^k + tgf_{tan}) < f(y^k)$.

Si existe tal t , definir $\bar{z}^{k,i} = y^k + tgf_{tan}$ e ir al Paso 2.c. Sino, ir al Paso 2.b.

Paso 2.b Definir $\tau_{max} = \min_{d_j \in D_k} \{1, \delta_{k,i}/\|d_j\|\}$

Determinar el mayor $\tau \in (0, \tau_{max}]$ tal que $f(y^k + \tau d_j) < f(y^k)$ para algún $d_j \in D_k$.

Definir $\bar{z}^{k,i} = y^k + \tau d_j$.

Adaptar gf_{tan} usando las últimas evaluaciones de f en D_k .

Paso 2.c Definir $z^{k,i}$, tal que $f(z^{k,i}) \leq \max\{f(\bar{z}^{k,i}), f(y^k) - \gamma_2, f(y^k) - \eta_1 \|gf_{tan}\| \delta_{k,i}\}$

$$d_{k,i} = gf_{tan}$$

Detalles del Algoritmo Básico del Paso III

1. El conjunto de puntos interpolantes, Y^k , dependiente de $r_k > 0$, garantiza la asintótica calidad del gradiente simplex calculado al inicio del Paso III (en $i = 0$). De igual modo, la alternativa de recalculer (en $i = 0$) el gradiente simplex usando los pasos calculados en las direcciones D_k , también contribuye al mejoramiento de la calidad del gradiente simplex. (ver Teoremas 3.1 y 5.1 en [39]).
2. La alternativa de analizar el decrecimiento de f sobre las direcciones D_k , cuando no se ha encontrado decrecimiento de f sobre gf_{tan} , tiene el propósito de averiguar si la función tiene o no un punto estacionario en y^k sobre π_k .

Bajo propiedades de diferenciabilidad de f , se conoce que el paso de búsqueda sobre D_k debe resultar exitoso salvo que el iterado y^k sea un punto “estacionario” sobre π_k [39]. En efecto, en el caso que no hay restricciones activas de cotas en y^k , sean w en $\mathcal{N}(A_k)$ un vector arbitrario y B_k base positiva en $\mathcal{N}(A_k)$. Luego, existe por lo menos un vector $d_j \in B_k$ tal que $w^T d_j > 0$.

Por tanto, si se considera $w = -P_{\pi_k}(\nabla f(y^k))$, no nulo, entonces el resultado previo garantiza la existencia de una dirección en B_k , tal que $w^T d_j > 0$. En consecuencia, existe alguna dirección de descenso de f , d_j en B_k .

Así, si y^k no es estacionario, se puede hallar una dirección d_j en B_k , donde existe un paso $\tau > 0$ suficientemente pequeño, tal que $f(y^k + \tau d_j) < f(y^k)$.

En el caso que existan cotas activas en y^k , como se explicó en Sección 4.2.5, se debe considerar un conjunto D_k , generador positivo del cono polar del cono de las restricciones activas, y de manera similar a lo explicado previamente, si $w = -P_{\pi_k}(\nabla f(y^k))$ es no nulo, se garantiza la existencia de una dirección de descenso para f en D_k . Debe existir alguna dirección d_j , tal que $w^T d_j > 0$. Por tanto, esa dirección d_j es de descenso para f y se podría hallar un punto tal que $f(y^k + \tau d_j) < f(y^k)$.

3. En el Algoritmo Básico del Paso III, si en la dirección gf_{tan} se obtiene un punto $z^{k,i}$, se declara la iteración exitosa, y no se realiza el estudio sobre el conjunto de vectores D_k .

En caso contrario, se analiza el conjunto $P_k = \{y^k + \tau d_j : d_j \in D_k\}$ en cierto orden, para un cierto τ , $0 < \tau$. Si se encuentra un punto $y^k + \tau d_j$ satisfaciendo que

$f(y^k + \tau d_j) < f(y^k)$, se define $z^{k,i} = y^k + \tau d_j$, declarando éxito y terminación. En caso contrario, se declara no exitoso al paso τ y se retorna al análisis de D_k con un nuevo paso $\tau/2$.

Si no se obtiene éxito, en gf_{tan} ni en las direcciones D_k , se declara “no éxito” y se sale con el punto y^k , asumiendo que y^k es un punto estacionario en π_k .

4. En el Algoritmo Básico del Paso III, se ha incluido, cuando es necesario analizar las direcciones en D_k y existe una dirección de descenso d_j en el Paso 1.b, la redefinición del gf_{tan} usando las evaluaciones de f , al menos sobre la d_j .

Es interesante observar que el nuevo gradiente simple, gf , cumplirá $\tau d_j^T gf = f(y^k + \tau d_j) - f(y^k) < 0$, $d_j \in \mathcal{N}(A_k)$, por tanto el nuevo modelo de la función f hereda la propiedad de descenso de f en la dirección d_j . Es decir, que el nuevo modelo también desciende en esa dirección d_j .

Teniendo en cuenta las observaciones realizadas en los párrafos previos obtenemos el siguiente resultado:

LEMA 4.3.1. El Algoritmo Básico del Paso III está bien definido.

Demostración. Si la proyección del verdadero $-\nabla f(y^k)$ sobre π_k es cero, después de un número finito de pasos en Paso 1.a y Paso 1.b, el algoritmo termina en el Paso 1.c con el mismo punto $z^{k,0} = y^k$.

Si la proyección sobre π_k de $-\nabla f(y^k)$ es no nula, el algoritmo encuentra en un número finito de pasos un $z^{k,0}$ donde $f(z^{k,0}) < f(y^k)$. Tal punto se encontrará sobre gf_{tan} si es una dirección de descenso de f , o en caso contrario, de acuerdo a lo explicado previamente, con seguridad existirá una dirección de descenso d_j en D_k y un $0 < \tau$, tal que $f(y^k + \tau d_j) < f(y^k)$. En consecuencia, en un número finito de pasos aceptará un $z^{k,0} \neq y^k$.

Cuando $i > 0$, se hace una búsqueda en una región de confianza $\delta_{k,i} < \delta_{k,0}$, correspondiente al caso con “éxito” en $i = 0$, por tanto existe una dirección de descenso.

□

En el siguiente teorema de esta sección probamos que el Algoritmo 3 IR-DFO está bien definido. Esto equivale a demostrar que existe un $\delta_{k,i}$ suficientemente pequeño, tal que la desigualdad $Ared_{k,i}(\theta) \geq 0.1 Pred_{k,i}(\theta)$ se cumple, y un punto $z^{k,i}$ es aceptado como nuevo iterado x^{k+1} , o eventualmente cuando x^k es factible y el $\delta_{k,i} \leq \min\{\epsilon_M, r_k\}$ termina con $x^{k+1} = x^k$.

TEOREMA 9. El Algoritmo 3 IR-DFO está bien definido.

Demostración. Consideramos

$$\begin{aligned}
Ared_{k,i} - 0.1Pred_{k,i} &= 0.9\theta_{k,i} \left[f(x^k) - f(z^{k,i}) \right] + (1 - \theta_{k,i}) \left[|C(x^k)| - |C(z^{k,i})| \right] - \\
&\quad - 0.1(1 - \theta_{k,i}) \left[|C(x^k)| - |C(y^k)| \right] \\
&= 0.9\theta_{k,i} \left[f(x^k) - f(z^{k,i}) \right] + 0.9(1 - \theta_{k,i}) \left[|C(x^k)| - |C(y^k)| \right] + \\
&\quad + (1 - \theta_{k,i}) \left[|C(x^k)| - |C(z^{k,i})| \right] - (1 - \theta_{k,i}) \left[|C(x^k)| - |C(y^k)| \right] \\
&= 0.9Pred_{k,i} + (1 - \theta_{k,i}) \left[|C(y^k)| - |C(z^{k,i})| \right].
\end{aligned}$$

Luego, por la propiedad (4.28), $Pred_{k,i}(\theta_{k,i}) \geq \frac{1}{2} [|C(x^k)| - |C(y^k)|]$ y por la condición (4.5), $|C(y^k)| \leq \alpha|C(x^k)|$, y teniendo en cuenta que $1 - \theta_{k,i} \geq -1$ (pues $\theta_{k,i} \in [0, 1]$), se obtiene

$$\begin{aligned}
Ared_{k,i} - 0.1Pred_{k,i} &\geq 0.45 \left[|C(x^k)| - |C(y^k)| \right] - \left| |C(y^k)| - |C(z^{k,i})| \right| \\
&\geq 0.45(1 - \alpha)|C(x^k)| - \left| |C(y^k)| - |C(z^{k,i})| \right|.
\end{aligned}$$

Entonces, si x^k no es factible, esto es $|C(x^k)| \neq 0$, como $\|y^k - z^{k,i}\| \leq \delta_{k,i}$ y $|C(x)|$ es una función continua, se tiene que $Ared_{k,i} \geq 0.1 Pred_{k,i}$ para un $\delta_{k,i}$ suficientemente pequeño. Define $x^{k+1} = z^{k,i}$.

Cuando x^k es factible, $y^k = x^k$ y $C(x^k) = C(y^k) = 0$. En este caso, en el Paso III.b del Algoritmo 3 IR-DFO hay dos posibles terminaciones. Una posibilidad es que exista $i \geq 0$ y $\delta_{k,i} > 0$ para el cual se cumpla

$$Ared_{k,i} - 0.1Pred_{k,i} \geq 0,$$

y en consecuencia, termina aceptando el nuevo iterado $x^{k+1} = z^{k,i}$. Sino, debido al decrecimiento del entorno, se llega a un radio $\delta_{k,i} \leq \min\{\epsilon_M, r_k\}$ y termina definiendo $x^{k+1} = x^k$.

Por tanto, queda demostrado que x^{k+1} está bien definido en ambos casos. Luego, el Algoritmo 3 IR-DFO está bien definido. □

Recordar que la función de mérito que se usa es $\psi(x, \theta) = \theta f(x) + (1 - \theta)|C(x)|$ donde $\theta \in [0, 1]$ es el parámetro de penalización y que, como se dijo previamente, en cada iteración la evaluación de la función de mérito en el nuevo punto $x^{k+1} = z^{k,i}$ es menor que en el iterado actual x^k , es decir $Ared_{k,i} := \psi(x^k, \theta_{k,i}) - \psi(z^{k,i}, \theta_{k,i}) > 0$.

En el Algoritmo 3 IR-DFO se pide una reducción “suficiente” para garantizar la convergencia

del mismo. Dicha reducción se evalúa, en cada iteración, mediante la desigualdad $Ared_{k,i} \geq 0.1Pred_{k,i}$ para aceptar un nuevo x^{k+1} .

Teniendo en cuenta las condiciones del Algoritmo 3 IR-DFO obtenemos el siguiente resultado.

TEOREMA 10. Supongamos que el Algoritmo 3 IR-DFO genera una sucesión infinita. Entonces

$$\lim_{k \rightarrow \infty} \psi(x^k, \theta_k) - \psi(x^{k+1}, \theta_k) = 0.$$

(Esto es, $\lim_{k \rightarrow \infty} Ared_{k,iacc(k)}(\theta_{k,iacc(k)}) = 0$.)

Demostración. La hacemos por contradicción.

Supongamos que existe un conjunto infinito de índices $K_1 \subset \mathbb{N}$ y un número $\gamma > 0$ tal que

$$\psi(x^{k+1}, \theta_k) \leq \psi(x^k, \theta_k) - \gamma,$$

para todo $k \in K_1$.

Denotamos $\psi_k = \psi(x^k, \theta_k)$ para todo $k \in \mathbb{N}$.

Para todo $k \in \mathbb{N}$ tenemos que

$$\begin{aligned} \psi_{k+1} &= \theta_{k+1}f(x^{k+1}) + (1 - \theta_{k+1})|C(x^{k+1})| \\ &= \left[\theta_{k+1}f(x^{k+1}) + (1 - \theta_{k+1})|C(x^{k+1})| \right] - \left[\theta_k f(x^{k+1}) + (1 - \theta_k)|C(x^{k+1})| \right] \\ &\quad + \left[\theta_k f(x^{k+1}) + (1 - \theta_k)|C(x^{k+1})| \right] \\ &= \left[(\theta_{k+1} - \theta_k)f(x^{k+1}) + (\theta_k - \theta_{k+1})|C(x^{k+1})| \right] + \left[\theta_k f(x^{k+1}) + (1 - \theta_k)|C(x^{k+1})| \right] \\ &= (\theta_k - \theta_{k+1}) \left(|C(x^{k+1})| - f(x^{k+1}) \right) + \left[\theta_k f(x^k) + (1 - \theta_k)|C(x^k)| \right] - \gamma_k \\ (4.34) \quad &= (\theta_k - \theta_{k+1}) \left(|C(x^{k+1})| - f(x^{k+1}) \right) + \psi_k - \gamma_k, \end{aligned}$$

donde $\gamma_k = \psi(x^{k+1}, \theta_k) - \psi(x^k, \theta_k) \geq 0$ para todo $k \in \mathbb{N}$ y $\gamma_k \geq \gamma > 0$ para todo $k \in K_1$. Ahora, por definición de $\theta_{k,-1}$ en el Paso 0 del Algoritmo 3 IR-DFO, tenemos que

$$(4.35) \quad \theta_k - \theta_{k+1} + \omega_k \geq 0, \text{ para todo } k \in \mathbb{N}.$$

Luego, por la propiedad de compacidad de Ω , existe una cota superior $c > 0$ tal que

$$\left| |C(x^k)| - f(x^k) \right| \leq c,$$

para todo $k \in \mathbb{N}$. Luego, por (4.34) y (4.35), tenemos que

$$\begin{aligned}\psi_{j+1} &= (\theta_j - \theta_{j+1} + \omega_j) (|C(x^{j+1})| - f(x^{j+1})) + \psi_j - \gamma_j - \omega_j (|C(x^{j+1})| - f(x^{j+1})) \\ &\leq (\theta_j - \theta_{j+1} + \omega_j)c + \psi_j - \gamma_j + \omega_j c,\end{aligned}$$

para $j = 0, 1, \dots, k-1$. Sumando estas k desigualdades, tenemos que

$$(4.36) \quad \psi_k \leq \psi_0 + (\theta_0 - \theta_k)c + \sum_{j=0}^{k-1} 2c\omega_j - \sum_{j=0}^{k-1} \gamma_j \leq \psi_0 + 2c + \sum_{j=0}^{k-1} 2c\omega_j - \sum_{j=0}^{k-1} \gamma_j,$$

para todo $k \geq 1$.

Como la serie $\sum_{j=0}^{\infty} \omega_j$ es convergente, y γ_k es estrictamente mayor que 0 (por estar acotado por $\gamma > 0$) para $k \in K_1$, (4.36) implica que ψ_k no está acotada inferiormente. Esto contradice la propiedad de compacidad de Ω .

Por tanto, vale que $\lim_{k \rightarrow \infty} \psi(x^k, \theta_k) - \psi(x^{k+1}, \theta_k) = 0$.

□

4.4. Convergencia a la factibilidad

TEOREMA 11. Si el Algoritmo 3 IR-DFO no termina en el Paso I, Fase de Factibilidad, para algún $k \in \mathbb{N}$, entonces

$$\lim_{k \rightarrow \infty} |C(x^k)| = 0.$$

(En particular, todo punto límite de $\{x^k\}$ es un punto factible.)

Demostración.

Por la condición (4.5), $|C(y^k)| \leq \alpha |C(x^k)|$, $0 < \alpha < 1$. Luego,

$$|C(x^k)| - |C(y^k)| \geq (1 - \alpha)|C(x^k)|.$$

Además, por la condición (4.28) se tiene que

$$Pred_{k,iacc(k)} \geq \frac{1}{2} \left[|C(x^k)| - |C(y^k)| \right].$$

Por la condición (4.31), se cumple

$$0.1Pred_{k,iacc(k)} \leq Ared_{k,iacc(k)}.$$

Por tanto, usando las tres desigualdades antes explicitadas, se tiene

$$|C(x^k)| \leq \frac{|C(x^k)| - |C(y^k)|}{1 - \alpha} \leq \frac{2}{1 - \alpha} Pred_{k,iacc(k)} \leq \frac{20}{1 - \alpha} Ared_{k,iacc(k)}.$$

Luego, usando la igualdad (4.30) se obtiene

$$|C(x^k)| \leq \frac{20}{1-\alpha} \left[\psi(x^k, \theta_k) - \psi(x^{k+1}, \theta_k) \right].$$

Por tanto, considerando el resultado del Teorema 10, $\lim_{k \rightarrow \infty} |C(x^k)| = 0$.

□

OBSERVACIÓN 4.4.1. Como la sucesión $\{y^k\}$ cumple que $|C(y^k)| \leq |C(x^k)|$, también se obtiene que $\lim_{k \rightarrow \infty} |C(y^k)| = 0$. Por tanto, todo punto límite de esa sucesión es factible.

Como consecuencia del Paso I del Algoritmo 3 IR-DFO, y^k satisface $\|y^k - x^k\| \leq \beta |C(x^k)|$, luego $\|y^k - x^k\|$ tiende a cero.

Además, si \bar{x} es punto límite de la sucesión $\{x^k\}$, es también punto límite de $\{y^k\}$, pues $\|y^k - \bar{x}\| \leq \|y^k - x^k\| + \|x^k - \bar{x}\|$. De igual manera, todo punto límite de $\{y^k\}$ es también punto límite de $\{x^k\}$.

4.5. Convergencia a la optimalidad

Demostraremos la convergencia de la sucesión $\{x^k\}$, $k \in \mathbb{N}$, generada por el Algoritmo 3 IR-DFO, suponiendo primero que vale cierta hipótesis, a la cual llamaremos Hipótesis B y luego, en el Teorema 13, probaremos que no es posible satisfacerla.

Hipótesis B. Existe $\epsilon > 0$ y $k_0 \in \mathbb{N}$ tal que

$$\|d_k\| \geq \epsilon \text{ para todo } k \geq k_0.$$

Los siguientes resultados serán usados para establecer los resultados de convergencia de la sucesión $\{x^k\}$.

TEOREMA 12. Supongamos que la Hipótesis B es verdadera. Entonces existe $k_1 \geq k_0$ tal que, para todo $k > k_1$, existen constantes c_2 y c_3 (independientes de k) para las cuales se satisface

$$f(y^k + t_{opt}d_k) \leq f(y^k) - \min\{c_2\|d_k\|^2, c_3\delta_{k,i}\|d_k\|\},$$

Demostración. Como f es Lipschitz continua

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L_2}{2} \|y - x\|^2,$$

para todo $x, y \in \Omega$. Luego, $y^k + d_k \in \Omega$ implica que para todo $t \in [0, 1]$,

$$f(y^k + td_k) \leq f(y^k) + t \langle \nabla f(y^k), d_k \rangle + \frac{t^2 L_2}{2} \|d_k\|^2$$

$$= f(y^k) + 0.1t \langle \nabla f(y^k), d_k \rangle + 0.9t \langle \nabla f(y^k), d_k \rangle + \frac{t^2 L_2}{2} \|d_k\|^2.$$

Por la Hipótesis B, para $k \geq k_0$, $\|d_k\| \geq \epsilon$, siendo $d_k = g_{f_{tan}}$. Como el cálculo de $g_{f_{tan}}$, en cada iteración k -ésima del Algoritmo 3 IR-DFO, depende de la condición del radio del conjunto interpolante Y^k , $\{r_k\}$, que es decreciente a cero, existe $k_1 \geq k_0$ tal que para todo $k \geq k_1$, $r_k < \frac{\epsilon}{4L_2\kappa_{eg}}$. Entonces, de acuerdo al resultado del Lema 4.2.1 se puede afirmar que, para todo $k \geq k_1$, $\langle \nabla f(y^k), d_k \rangle < -\frac{1}{4}\|d_k\|^2$.

Luego, para todo $k \geq k_1$,

$$\begin{aligned} f(y^k + td_k) &\leq f(y^k) + 0.1t \langle \nabla f(y^k), d_k \rangle - \frac{0.9t}{4} \|d_k\|^2 + \frac{t^2 L_2}{2} \|d_k\|^2 \\ &= f(y^k) + 0.1t \langle \nabla f(y^k), d_k \rangle + t \|d_k\|^2 \left(-\frac{0.9}{4} + \frac{t L_2}{2} \right). \end{aligned}$$

Entonces, si $t \leq \frac{0.9}{2L_2}$ se tiene, usando nuevamente el Lema 4.2.1, que

$$\begin{aligned} f(y^k + td_k) &\leq f(y^k) + 0.1t \langle \nabla f(y^k), d_k \rangle \\ &\leq f(y^k) - \frac{0.1t}{4} \|d_k\|^2. \end{aligned}$$

Considerando $t_{max} = \min \left\{ 1, \frac{\delta_{k,i}}{\|d_k\|} \right\} \in [0, 1]$, $y^k + t_{min} d_k \in \pi_k$ tal que $\|t_{min} d_k\| \leq \delta_{k,i}$ ($y^k + t_{min} d_k$ se encuentra dentro de la región de confianza) y por tanto, el “mejor” t , t_{opt} es tal que

$$t_{opt} \leq \min \left\{ 1, \frac{\delta_{k,i}}{\|d_k\|}, \frac{0.9}{2L_2} \right\}.$$

Entonces,

$$\begin{aligned} f(y^k + t_{opt} d_k) &\leq f(y^k) - \frac{0.1}{4} \min \left\{ 1, \frac{\delta_{k,i}}{\|d_k\|}, \frac{0.9}{2L_2} \right\} \|d_k\|^2 \\ &\leq f(y^k) - \min \left\{ \frac{0.1}{4} \|d_k\|^2, \frac{0.1}{4} \delta_{k,i} \|d_k\|, \frac{0.1}{4} \frac{0.9}{2L_2} \|d_k\|^2 \right\}. \end{aligned}$$

En consecuencia,

$$f(y^k + t_{opt} d_k) \leq f(y^k) - \min \{ c_2 \|d_k\|^2, c_3 \delta_{k,i} \|d_k\| \},$$

con $c_3 = \frac{0.1}{4}$ y $c_2 = c_3 \min \left\{ 1, \frac{0.9}{2L_2} \right\}$.

□

LEMA 4.5.1. Si vale la Hipótesis B, existe un $k_2 \geq k_1$, tal que si $k > k_2$ y x^k es factible ($C(x^k) = 0$), entonces existe un $\delta_{k,i} > 0$, $i \geq 0$, para el cual se cumple la condición de aceptación

$$Ared_{k,i} - 0.1Pred_{k,i} \geq 0,$$

en el Paso III.b del Algoritmo 3 IR-DFO.

Demostración. Si x^k es factible, $|C(x^k)| = 0$, e $y^k = x^k$. Por tanto, dado el $z^{k,i}$ calculado para un $i \geq 0$, siendo $f(z^{k,i}) \leq f(y^k)$, $Ared_{k,i} = \theta_{k,i}[f(x^k) - f(z^{k,i})] - (1 - \theta_{k,i})|C(z^{k,i})|$ y $Pred_{k,i} = \theta_{k,i}[f(x^k) - f(z^{k,i})]$, se tiene la expresión:

$$Ared_{k,i} - 0.1Pred_{k,i} = 0.9\theta_{k,i}[f(x^k) - f(z^{k,i})] - (1 - \theta_{k,i})|C(z^{k,i})|.$$

Como $f(x^k) = f(y^k)$, para todo i , $i \geq 0$ y para todo $\theta \in (0, 1]$ se cumple que $Pred_{k,i}(\theta) = \theta(f(x^k) - f(z^{k,i})) \geq \frac{1}{2}(|C(x^k)| - |C(y^k)|) = 0$. Así, resulta $\theta_{k,i} = \theta_{k,-1}$ para todo i , $i \geq 0$.

Por tanto,

$$Ared_{k,i} - 0.1Pred_{k,i} = 0.9\theta_{k,-1}[f(x^k) - f(z^{k,i})] - (1 - \theta_{k,-1})|C(z^{k,i})|.$$

Por contradicción, asumimos que para todo $k \geq k_1$, si $C(x^k) = 0$ no se cumple la condición $Ared_{k,i} - 0.1Pred_{k,i} \geq 0$, así como $\delta_{k,i}$ decrece se llega a un $\delta_{k,i} \leq \min\{\epsilon_M, r_k\}$, tal que $Ared_{k,i} - 0.1Pred_{k,i} < 0$.

En tal caso, si $z^{k,i} \neq y^k = x^k$, por el Teorema 12 y el Teorema 8 se tiene para $k \geq k_1$,

$$\begin{aligned} 0 &> Ared_{k,i} - 0.1Pred_{k,i} \\ &\geq 0.9\theta_{k,-1} \min\{c_2\|d_k\|^2, c_3\|d_k\|\delta_{k,i}\} - (1 - \theta_{k,-1})(L_1\kappa_{eg}\Delta_k\delta_{k,i} + L_1\delta_{k,i}^2). \end{aligned}$$

Luego, por las condiciones del Algoritmo 3 IR-DFO, en particular la condición (4.27) requiere que $\Delta_k \leq r_k$,

$$0.9\theta_{k,-1}\|d_k\| \min\{c_2\|d_k\|, c_3\delta_{k,i}\} < (1 - \theta_{k,-1})(L_1\kappa_{eg}\Delta_k\delta_{k,i} + L_1\delta_{k,i}^2).$$

Dividiendo por $\delta_{k,i} > 0$, en la desigualdad anterior se obtiene

$$(4.37) \quad 0.9\theta_{k,-1}\|d_k\| \min\{c_2\|d_k\|/\delta_{k,i}, c_3\} < (1 - \theta_{k,-1})(L_1\kappa_{eg}\Delta_k + L_1\delta_{k,i})$$

Luego, si $\|d_k\| < \delta_{k,i}$, entonces $\|d_k\| < r_k$ (por ser $\delta_{k,i} \leq r_k$).

En el caso contrario, si $\|d_k\| \geq \delta_{k,i}$, entonces reemplazando en el lado derecho de (4.37), $\delta_{k,i} \leq r_k$ y $\Delta_k \leq r_k$, y considerando que $\frac{\|d_k\|}{\delta_{k,i}} \geq 1$,

$$0.9\theta_{k,-1}\|d_k\| \min\{c_2, c_3\} < (1 - \theta_{k,-1})(L_1\kappa_{eg}r_k + L_1r_k).$$

Luego, si $A = \min\{c_2, c_3\}$ y $B = \frac{1-\theta_{k,-1}}{0.9\theta_{k,-1}}L_1(\kappa_{eg} + 1)$, se obtiene $\|d_k\| < \frac{B}{A}r_k$.

Por la Hipótesis B, sabemos que $\|d_k\| \geq \epsilon$ para todo $k \geq k_0$.

Como no puede ocurrir simultáneamente que $\epsilon \leq \|d_k\|$ y $\|d_k\| < r_k$ o $\|d_k\| < \frac{B}{A}r_k$, para un k suficientemente grande, pues r_k tiende a cero, se llega a un absurdo.

Tal absurdo provino de suponer que para todo $k > k_1 \geq k_0$, si $C(x^k) = 0$, en el Paso III.b del Algoritmo 3 IR-DFO, se aceptaba $x^{k+1} = x^k$ debido a que no existía un $\delta_{k,i} \geq \min\{\epsilon_M, r_k\}$ para el cual se cumpliera $Ared_{k,i} - 0.1Pred_{k,i} \geq 0$.

Por tanto, si la Hipótesis B vale, existe un k_2 suficiente grande $k_2 > k_1$, para el cual si $k > k_2$ y $C(x^k) = 0$, existe un $i \geq 0$, para el cual se cumple la condición de aceptación para $x^{k+1} = z^{k,i}$.

□

OBSERVACIÓN 4.3. La demostración previa es similar a la realizada para el caso de IR con derivadas con la dificultad que, en nuestro caso, el deterioro de la factibilidad sobre el hiperplano tangente es de primer orden. Ver Teorema 8. Esta dificultad plantea la necesidad de una cota dependiente de los términos de una serie convergente (r_k) para $\delta_{k,i}$. Los términos r_k también son una cota para la distancia de los puntos de interpolación (Δ_k).

LEMA 4.5.2. Supongamos que la Hipótesis B es verdadera. Entonces existen $c_4, c_5 > 0$ (independientes de k) tales que

$$f(y^k) - f(z^{k,i}) \geq \min\{c_4, c_5\delta_{k,i}\}$$

para todo $k \geq k_1 > k_0$, $i = 0, 1, \dots, iacc(k)$.

Demostración. Por el Teorema 12, para todo $k \geq k_1 > k_0$, existen constantes c_2 y c_3 independientes de k tales que

$$f(y^k) - f(z^{k,i}) \geq \min\{c_2\|d_k\|^2, c_3\delta_{k,i}\|d_k\|\}.$$

Luego, por la Hipótesis B,

$$f(y^k) - f(z^{k,i}) \geq \min\{c_2\epsilon^2, c_3\delta_{k,i}\epsilon\}.$$

Sean $c_4 = c_2\epsilon^2$ y $c_5 = c_3\epsilon$ queda demostrado el Lema.

□

LEMA 4.5.3. Supongamos que se cumple la Hipótesis B. Si existen constantes $\gamma_1, \gamma_2 > 0$, independientes de k e i , tal que $|C(x^k)| \leq \min\{\gamma_2, \gamma_1\delta_{k,i}\}$ entonces se puede definir $\theta_{k,i} = \theta_{k,i-1}$, para $k > k_1$.

Demostración. Considerando que

$$\begin{aligned} \text{Pred}_{k,i}(1) &= f(x^k) - f(z^{k,i}) \\ &\geq f(y^k) - f(z^{k,i}) - |f(x^k) - f(y^k)| \\ &\geq f(y^k) - f(z^{k,i}) - c\|y^k - x^k\|, \end{aligned}$$

donde c es una constante que indica una cota superior del $\|\nabla f(x)\|$ en Ω . Luego, por (4.4) se tiene $\|y^k - x^k\| \leq \beta|C(x^k)|$, y usando la desigualdad del Lema 4.5.2, para $k \geq k_1$, se obtiene

$$\text{Pred}_{k,i}(1) - \frac{1}{2}|C(x^k)| \geq \min\{c_4, c_5\delta_{k,i}\} - (c\beta + \frac{1}{2})|C(x^k)|.$$

Definiendo $\gamma_2 = \frac{c_4}{c\beta+0,5}$ y $\gamma_1 = \frac{c_5}{c\beta+0,5}$, y si $|C(x^k)| \leq \min\{\gamma_2, \gamma_1\delta_{k,i}\}$ se obtiene que

$$\text{Pred}_{k,i}(1) - \frac{1}{2}|C(x^k)| \geq 0.$$

Por tanto, $f(x^k) - f(z^{k,i}) \geq \frac{1}{2}(|C(x^k)| - |C(y^k)|)$.

Como $\text{Pred}_{k,i}(\theta_{k,i}) = \theta_{k,i}[f(x^k) - f(z^{k,i})] + (1 - \theta_{k,i})[|C(x^k)| - |C(y^k)|]$, la desigualdad previa implica que para cualquier valor de $\theta_{k,i}$ en el intervalo $(0, 1]$ se cumple la condición (4.28), $\text{Pred}_{k,i}(\theta_{k,i}) \geq \frac{1}{2}(|C(x^k)| - |C(y^k)|)$.

Entonces, en particular, $\theta_{k,i-1}$ satisface (4.28), como se quería probar.

□

LEMA 4.5.4. Asumiendo que la Hipótesis B es verdadera y sea γ_1 del Lema 4.5.3. Si $|C(x^k)|$ es suficientemente pequeña, un $\delta_{k,i}$ que satisface

$$(4.38) \quad \frac{\gamma_1}{10}\delta_{k,i} \leq |C(x^k)|,$$

es necesariamente aceptado en el Paso III.b del Algoritmo 3 IR-DFO.

Demostración. Asumamos que (4.38) se cumple. Entonces, por (4.5), $|C(y^k)| \leq \alpha|C(x^k)|$, y condición (4.28) del algoritmo IR-DFO

$$\text{Pred}_{k,i} \geq \frac{1}{2} [|C(x^k)| - |C(y^k)|] \geq \frac{1-\alpha}{2}|C(x^k)| \geq \frac{(1-\alpha)\gamma_1}{20}\delta_{k,i}.$$

Por tanto, (4.38) implica que

$$(4.39) \quad \delta_{k,i} \leq \frac{20}{(1-\alpha)\gamma_1} \text{Pred}_{k,i}.$$

Por el resultado del Teorema 8, $|C(z^{k,i})| \leq |C(y^k)| + L_1\kappa_{eg}\Delta_k\|z^{k,i} - y^k\| + L_1\|z^{k,i} - y^k\|^2$, y por ser $1 - \theta_{k,i} > -1$,

$$\text{Ared}_{k,i} = \text{Pred}_{k,i} + (1 - \theta_{k,i}) \left[|C(y^k)| - |C(z^{k,i})| \right] \geq \text{Pred}_{k,i} - L_1\kappa_{eg}\Delta_k\delta_{k,i} - L_1\delta_{k,i}^2.$$

Luego, por (4.39), (4.38) implica que

$$\begin{aligned} \text{Ared}_{k,i} &\geq \text{Pred}_{k,i} - L_1\kappa_{eg}\Delta_k \frac{20}{(1-\alpha)\gamma_1} \text{Pred}_{k,i} - L_1\delta_{k,i} \frac{20}{(1-\alpha)\gamma_1} \text{Pred}_{k,i} \\ &\geq \text{Pred}_{k,i} - L_1\kappa_{eg}\Delta_k \frac{20}{(1-\alpha)\gamma_1} \text{Pred}_{k,i} - L_1 \frac{10}{\gamma_1} |C(x^k)| \frac{20}{(1-\alpha)\gamma_1} \text{Pred}_{k,i} \\ &\geq \text{Pred}_{k,i} \left[1 - L_1\kappa_{eg}\Delta_k \frac{20}{(1-\alpha)\gamma_1} - L_1 \frac{10}{\gamma_1} |C(x^k)| \frac{20}{(1-\alpha)\gamma_1} \right] \\ &\geq \text{Pred}_{k,i} \left[1 - L_1\kappa_{eg}\beta |C(x^k)| \frac{20}{(1-\alpha)\gamma_1} - L_1 \frac{10}{\gamma_1} |C(x^k)| \frac{20}{(1-\alpha)\gamma_1} \right] \\ &\geq \text{Pred}_{k,i} \left[1 - |C(x^k)| \left(\frac{L_1\kappa_{eg}\beta\gamma_1 20 + L_1 200}{(1-\alpha)\gamma_1^2} \right) \right]. \end{aligned}$$

Luego, cuando (4.38) vale y $|C(x)| \leq \frac{0.9(1-\alpha)\gamma_1^2}{L_1\gamma_1\beta 20 + 200L_1} = H$, el punto de prueba $z^{k,i}$ cumple la condición para ser aceptado en el Paso III.b del Algoritmo 3 IR-DFO.

□

LEMA 4.5.5. Supongamos que la Hipótesis B es verdadera. Entonces, existe $\bar{\theta} > 0$ tal que $\theta_k \geq \bar{\theta}$ para todo $k \geq k_3$, $k \in \{1, 2, \dots\}$.

Demostración.

Sea $\epsilon_2 = \min\{\gamma_2, H, \gamma_1\delta_{min}\}$, siendo γ_1 y γ_2 las constantes definidas en el Lema 4.5.3, y $H = \frac{0.9(1-\alpha)\gamma_1^2}{L_1\gamma_1\beta 20 + 200L_1}$ la constante de la demostración del Lema 4.5.4.

Sea $k_3 \geq k_2 > k_1$ tal que $|C(x^k)| \leq \epsilon_2$ para todo $k \geq k_3$, es posible considerar lo anterior por el resultado conocido que $|C(x^k)| \rightarrow 0$.

Como $\delta_{min} \geq \frac{|C(x^k)|}{\gamma_1}$, por ser $\delta_{min} \geq \frac{\epsilon_2}{\gamma_1}$, implica que para todo $k \geq k_3$, $\delta_{k,0} \geq \frac{|C(x^k)|}{\gamma_1}$ por ser $\delta_{k,0} \geq \delta_{min}$.

Entonces, un radio de región de confianza tal que $\delta_{k,i} < \frac{|C(x^k)|}{\gamma_1}$ no corresponde a $i = 0$. Entonces, debe existir un $\delta_{k,i-1}$ el cual verifica

$$\delta_{k,i-1} \leq 10\delta_{k,i} < 10 \frac{|C(x^k)|}{\gamma_1}.$$

Es decir, $\delta_{k,i-1}$, cumpliría la condición (4.38), $\frac{\gamma_1}{10}\delta_{k,i-1} \leq |C(x^k)|$, entonces por lo demostrado en Lema 4.5.4, se hubiese aceptado el $z^{k,i-1}$.

Luego, debe cumplirse que $\delta_{k,i} \geq \frac{|C(x^k)|}{\gamma_1}$.

En consecuencia, como $|C(x^k)| \leq \epsilon_2$ para todo $k \geq k_3$, y $|C(x^k)| \leq \delta_{k,i}\gamma_1$, por el resultado del Lema 4.5.3 el parámetro de penalización $\theta_{k,i}$ no cambia para todo $k \geq k_3$, $i = 0, 1, \dots, iacc(k)$. Luego, existe $\bar{\theta} > 0$ tal que $\theta_k \geq \bar{\theta}$ para todo $k \geq k_3$, $k \in \{1, 2, \dots\}$. Se tiene el resultado deseado. □

Finalmente probaremos que la Hipótesis B no puede ser verdadera. Se verá que los resultados obtenidos bajo esa hipótesis llevan a un absurdo.

TEOREMA 13. Sea $\{x^k\}$ la sucesión infinita generada por el Algoritmo 3 IR-DFO. Entonces, existe K_2 , un subconjunto infinito de \mathbb{N} , tal que

$$(4.40) \quad \lim_{\substack{k \rightarrow \infty \\ k \in K_2}} \|d_k\| = 0.$$

Demostración. Supongamos que la tesis de este Teorema no es cierta. Entonces, dado $\epsilon > 0$, existe $k_0 \in \mathbb{N}$ para el cual la Hipótesis B es verdadera.

Consideramos la expresión de $Ared_{k,i} - 0.1Pred_{k,i}$, y el resultado del Teorema 8,

$$\begin{aligned} Ared_{k,i} - 0.1Pred_{k,i} &= \\ &= 0.9 \left\{ \theta_{k,i} [f(x^k) - f(z^{k,i})] + (1 - \theta_{k,i}) [|C(x^k)| - |C(y^k)|] \right\} + \\ &\quad + (1 - \theta_{k,i}) [|C(y^k)| - |C(z^{k,i})|] \\ &\geq 0.9\theta_{k,i} [f(y^k) - f(z^{k,i})] + 0.9\theta_{k,i} [f(x^k) - f(y^k)] - (1 - \alpha)|C(x^k)| - \\ &\quad - L_1\kappa_{eg}\Delta_k\delta_{k,i} - L_1\delta_{k,i}^2. \end{aligned}$$

Entonces, por Lema 4.5.2, existen $c_4, c_5 > 0$ tales que

$$f(y^k) - f(z^{k,i}) \geq \min\{c_4, c_5\}\delta_{k,i} \quad \forall k \geq k_1 > k_0, \quad i = 0, 1, \dots, iacc(k).$$

Por el resultado del Lema 4.5.5, existe $\bar{\theta} > 0$ tal que $\theta_k \geq \bar{\theta}$, y $\delta_{k,i} \geq \frac{|C(x^k)|}{\gamma_1}$, para todo $k \geq k_3 > k_1$, por (4.4), se cumple $\|y^k - x^k\| \leq \beta|C(x^k)|$. Además, por la condición del radio del

conjunto de puntos interpolación Y^k , se cumple $\Delta_k \leq \beta|C(x^k)|$, entonces,

$$\delta_{k,i} \geq \frac{|C(x^k)|}{\gamma_1} \geq \frac{\Delta_k}{\beta\gamma_1} \text{ para todo } k \geq k_3, i = 0, 1, \dots, i_{acc}(k).$$

$$\begin{aligned} Ared_{k,i} - 0.1Pred_{k,i} &\geq 0.9\bar{\theta} \min\{c_4, c_5\delta_{k,i}\} - c\beta|C(x^k)| - L_1\kappa_{eg}\Delta_k\delta_{k,i} - L_1\delta_{k,i}^2 \\ &\geq 0.9\bar{\theta} \min\{c_4, c_5\delta_{k,i}\} - c\beta|C(x^k)| - L_1\kappa_{eg}\beta\gamma_1\delta_{k,i}^2 - L_1\delta_{k,i}^2 \\ &\geq 0.9\bar{\theta} \min\{c_4, c_5\delta_{k,i}\} - c\beta|C(x^k)| - c_6\delta_{k,i}^2, \end{aligned}$$

para todo $k \geq k_3 \geq k_2 \geq k_1 \geq k_0$, $i = 0, 1, \dots, i_{acc}(k)$, donde c es una constante que depende de una cota de $\|\nabla f(x)\|$ en Ω y $c_6 = L_1\kappa_{eg}\beta\gamma_1 + L_1$.

Definimos

$$\bar{\delta} = \min \left\{ \left(\frac{0.45\bar{\theta}c_4}{c_6} \right)^{\frac{1}{2}}, \frac{0.45\bar{\theta}c_5}{c_6} \right\}.$$

Si $\delta_{k,i} \leq \bar{\delta}$ tenemos que

$$c_6\delta_{k,i}^2 \leq 0.45\bar{\theta} \min\{c_4, c_5\delta_{k,i}\},$$

por tanto, cuando $\delta_{k,i} \leq \bar{\delta}$ tenemos que

$$(4.41) \quad Ared_{k,i} - 0.1Pred_{k,i} \geq 0.45\bar{\theta} \min\{c_4, c_5\delta_{k,i}\} - c\beta|C(x^k)|,$$

para todo $k \geq k_3$, $i = 0, 1, \dots, i_{acc}(k)$. Sea $k_4 \geq k_3$ tal que

$$(4.42) \quad c\beta|C(x^k)| \leq 0.45\bar{\theta} \min \left\{ c_4, c_5 \frac{\bar{\delta}}{10} \right\},$$

para todo $k \geq k_4$. Por (4.41) y (4.42) tenemos que para todo $k \geq k_4$, si $i \in \mathbb{N}$ se corresponde con el primer radio de confianza $\delta_{k,i}$ menor o igual que $\bar{\delta}$ (por tanto, $\bar{\delta} \geq \delta_{k,i} \geq \frac{\bar{\delta}}{10}$),

$$Ared_{k,i} - 0.1Pred_{k,i} \geq 0.$$

Esto significa que $\delta_{k,i} \geq \frac{\bar{\delta}}{10}$ debe ser aceptado. Luego,

$$\delta_{k,i_{acc}(k)} \geq \frac{\bar{\delta}}{10},$$

para todo $k \geq k_4$.

Observar que,

$$f(y^k) - f(x^k) \geq -|f(y^k) - f(x^k)|,$$

por (4.5),

$$(1 - \theta_{k,i_{acc}(k)}) \left[|C(x^k)| - |C(y^k)| \right] \geq (1 - \theta_{k,i_{acc}(k)})(1 - \alpha)|C(x^k)|$$

$$\geq -|C(x^k)|,$$

ya que $(1 - \theta_{k,iacc(k)})(1 - \alpha) > -1$.

Por tanto, si $k \geq k_4$, por el resultado del Lema 4.5.2, Lema 4.5.5 y (4.4), se tiene que

$$\begin{aligned} \text{Pred}_{k,iacc(k)} &= \theta_{k,iacc(k)} [f(x^k) - f(z^{k,i})] + (1 - \theta_{k,iacc(k)}) [|C(x^k)| - |C(y^k)|] \\ &= \theta_{k,iacc(k)} [f(y^k) - f(z^{k,i})] + \theta_{k,iacc(k)} [f(y^k) - f(x^k)] + \\ &\quad + (1 - \theta_{k,iacc(k)}) [|C(x^k)| - |C(y^k)|] \\ &\geq \bar{\theta} [f(y^k) - f(z^{k,i})] - |f(y^k) - f(x^k)| - |C(x^k)| \\ (4.43) \quad &\geq \bar{\theta} \min \left\{ c_4, \frac{c_5 \bar{\delta}}{10} \right\} - c' |C(x^k)|, \end{aligned}$$

donde c' es una constante que depende de $\|\nabla f(x)\|$ en Ω .

Sea $k_5 \geq k_4$ tal que $c' |C(x^k)| \leq 0,5\bar{\theta} \min \left\{ c_4, \frac{c_5 \bar{\delta}}{10} \right\}$, para todo $k \geq k_5$.

Luego, se obtiene $\text{Pred}_{k,iacc(k)} \geq 0,5\bar{\theta} \min \left\{ c_4, \frac{c_5 \bar{\delta}}{10} \right\}$, para $k \geq k_5$. Esto implica, por (4.31), como $\text{Ared}_{k,iacc(k)} \geq 0,1\text{Pred}_{k,iacc(k)}$, que $\text{Ared}_{k,iacc(k)}$ se mantiene acotado por arriba de cero por una constante, para todo $k \geq k_5$.

Claramente lo anterior contradice el resultado del Teorema 10. Esto significa que la Hipótesis B no puede ser cierta, y así se obtiene el resultado deseado. □

En el siguiente Teorema 14 demostramos que existe una subsucesión de la sucesión $\{x^k\}$, $k \in \mathbb{N}$, generada por el Algoritmo 3 IR-DFO, tal que la norma de la proyección del gradiente de la función objetivo sobre el hiperplano tangente aproximado a las restricciones π_k tiende a cero. Con este resultado, el Teorema 13 y si se satisfacen algunas otras propiedades demostramos que si la sucesión, $\{x^k\}$, $k \in \mathbb{N}$, generada por el Algoritmo 3 IR-DFO converge a un punto \bar{x} factible entonces \bar{x} es un punto KKT.

TEOREMA 14. Sea $\{y^k\}$ una sucesión infinita generada por el Algoritmo 3 IR-DFO. Entonces,

$$(4.44) \quad \lim_{k \in K_2} \|P_{\pi_k}(y^k - \nabla f(y^k)) - y^k\| = 0.$$

Demostración. Por la condición (4.15),

$$\|\nabla f(y^k) - gf\| \leq L_2 \kappa_{eg} \Delta_k,$$

con $\{\Delta_k\}$ una sucesión que tiende a cero. Por tanto, considerando una desigualdad similar a la (4.22), usada en la Propiedad 4.2.2:

$$\begin{aligned} \|y^k - P_{\pi_k}(y^k - \nabla f(y^k))\|_2 &= \|y^k - P_{\pi_k}(y^k - \nabla f(y^k) - gf + gf)\|_2 \\ &\leq \|y^k - P_{\pi_k}(y^k - gf)\|_2 + \|\nabla f(y^k) - gf\|_2 \\ &\leq \|y^k - P_{\pi_k}(y^k - gf)\|_2 + \|\nabla f(y^k) - gf\|_2. \end{aligned}$$

Por tanto, cuando $k \in K_2$, k tiende a infinito se obtiene el resultado requerido. \square

Sea \bar{x} un punto factible del problema de programación no lineal. Se dice que \bar{x} satisface la condición de calidad de Dependencia Lineal Constante Positiva (CPLD) [7] si cualquier combinación lineal no negativa (no trivial) de los gradientes de las restricciones activas, correspondientes a las desigualdades activas, implica que los gradientes en esa combinación lineal son linealmente dependientes para todo z en un entorno de \bar{x} .

En [7] se demostró que la condición CPLD es una condición de calidad y que es más débil que la condición de Independencia Lineal de los gradientes de las restricciones activas (LICQ) y la condición de Mangasarian-Fromovitz (MFCQ). Luego, los resultados de convergencia a puntos KKT basados en la condición CPLD, son más fuertes que aquellos resultados dependientes de la condición LICQ.

A continuación, demostramos que si el punto límite de la sucesión generada por el Algoritmo 3 IR-DFO satisface la condición de calidad de CPLD [7, 11], entonces es un punto KKT.

TEOREMA 15. Sea $\{x^k\}$ una sucesión infinita generada por el Algoritmo 3 IR-DFO. Sean $\{x^k\}$ e $\{y^k\}$, con $k \in K_2$, las subsucesiones de los Teoremas 13 y 14 respectivamente, y \bar{x} el punto límite factible. Si $\|A_k - C'(y^k)\|$ tiende a cero y si en \bar{x} se cumple la condición de calidad CPLD entonces \bar{x} es un punto KKT.

Demostración. Sabemos por el Teorema 14 que

$$\lim_{k \in K_2} \|(P_{\pi_k}(y^k - \nabla f(y^k)) - y^k)\| = 0.$$

Considerando que $p_k = P_{\pi_k}(y^k - \nabla f(y^k))$, es la solución óptima del problema:

$$(4.45) \quad \min \|p - (y^k - \nabla f(y^k))\|^2, \text{ sobre } p \in \pi_k$$

$$\pi_k = \{p \in \Omega \mid A_k(p - y^k) = 0\},$$

usaremos en lo que sigue que p_k satisface la condición de optimalidad de ese problema, por ser un problema con restricciones lineales.

Sea $\Omega = \{x \in \mathbb{R}^n : L_i \leq x_i \leq U_i\}$. Existen vectores $\lambda_k \in \mathbb{R}^m$, $\mu_k^u \in \mathbb{R}_+^n$, y $\mu_k^l \in \mathbb{R}_+^n$, para los cuales se cumple,

$$-\left(p_k - (y^k - \nabla f(y^k))\right) = A_k^T \lambda_k + \sum_{i=1}^n \mu_{k,i}^u e_i + \sum_{i=1}^n \mu_{k,i}^l (-e_i),$$

sujeto a $A_k(p_k - y^k) = 0$, y $\mu_{k,i}^u \geq 0$, $\mu_{k,i}^l \geq 0$, $\mu_{k,i}^u(U_i - p_{k,i}) = 0$, y $\mu_{k,i}^l(p_{k,i} - L_i) = 0$, $i = 1, \dots, n$.

Luego, se tiene

$$p_k - y^k = (-\nabla f(y^k)) - A_k^T \lambda_k - \sum_{i=1}^n \mu_{k,i}^u e_i - \sum_{i=1}^n \mu_{k,i}^l (-e_i).$$

Como $p_k - y^k$ tiende a cero, e y^k tiende a \bar{x} como punto límite, entonces p_k también tiene a \bar{x} como punto límite.

En el punto límite factible \bar{x} , consideramos los siguientes conjuntos de índices $I_U = \{i \in \{1, 2, \dots, n\} : \bar{x}_i = U_i\}$, $I_L = \{i \in \{1, 2, \dots, n\} : \bar{x}_i = L_i\}$ y $I_0 = \{i \in \{1, 2, \dots, n\} : L_i < \bar{x}_i < U_i\}$.

Entonces, para $i \in I_0$, existe un $k_0 \in K_2$, tal que para $k > k_0$, $k \in K_2$, se cumple que $L_i < p_{k,i} < U_i$, considerando que p_k tiende a \bar{x} . En consecuencia, $\mu_{k,i}^u = 0$ para todo i que no pertenece a I_U . Análogamente, $\mu_{k,i}^l = 0$ para todo i que no pertenece a I_L .

Así, para $k > k_0$, $k \in K_2$,

$$p_k - y^k = (-\nabla f(y^k)) - A_k^T \lambda_k - \sum_{i \in I_U} \mu_{k,i}^u e_i - \sum_{i \in I_L} \mu_{k,i}^l (-e_i),$$

$A_k(p_k - y^k) = 0$, siendo $\mu_{k,i}^u \geq 0$, $\mu_{k,i}^l \geq 0$, $\mu_{k,i}^u(U_i - p_{k,i}) = 0$, $i \in I_U$ y $\mu_{k,i}^l(p_{k,i} - L_i) = 0$, $i \in I_L$.

Por tanto, por el Teorema de Caratheodory sobre Conos [21] (en Exercise B.1.7, pag. 689), para todo $k \in K_2$, $k > k_0$ existe $I^k \subseteq \{1, 2, \dots, n\}$, $I_U^k \subseteq I_U$, e $I_L^k \subseteq I_L$ tal que el conjunto de vectores:

$$\left\{ \{a_i^k\}_{i \in I^k}, \{e_i\}_{i \in I_U^k}, \{-e_i\}_{i \in I_L^k} \right\},$$

es linealmente independiente. En consecuencia, siendo $a_i^k = [A_k]_i^T$, se puede expresar

$$p_k - y^k = (-\nabla f(y^k)) - \sum_{i \in I^k} a_i^k \lambda_{k,i} - \sum_{i \in I_U^k} \mu_{k,i}^u e_i - \sum_{i \in I_L^k} \mu_{k,i}^l (-e_i),$$

que sabemos que tiende a cero, porque $p_k - y^k$ tiende a cero.

Como hay un número finito de posibilidades para generar tales conjuntos independientes, existe un conjunto infinito de índices $K_3 \subset \{k \in K_2 : k > k_0\}$, para los cuales tales conjuntos deben repetirse. Así, para $k \in K_3$, $I^k = \bar{I}$, $I_U^k = \bar{I}_U$, $I_L^k = \bar{I}_L$.

Por tanto, para todo $k \in K_3$,

$$p_k - y^k = (-\nabla f(y^k)) - \sum_{i \in \bar{I}} a_i^k \lambda_{k,i} - \sum_{i \in \bar{I}_U} \mu_{k,i}^u e_i - \sum_{i \in \bar{I}_L} \mu_{k,i}^l (-e_i),$$

el conjunto de vectores $\{\{a_i^k\}_{i \in \bar{I}}, \{e_i\}_{i \in \bar{I}_U}, \{-e_i\}_{i \in \bar{I}_L}\}$ es linealmente independiente.

Si se considera $S_k = \max\{\max\{|\lambda_{k,i}|, i \in \bar{I}\}, \max\{\mu_{k,i}, i \in \bar{I}_U\}, \max\{\mu_{k,i}, i \in \bar{I}_L\}\}$, $k \in K_3$, analizamos los casos siguientes:

Si $\{S_k\}_{k \in K_3}$ es acotada, entonces $\lim_{k \in K_3} \lambda_{k,i} = \lambda_i$,

$\lim_{k \in K_3} \mu_{k,i}^u = \mu_i^u$, y $\lim_{k \in K_3} \mu_{k,i}^l = \mu_i^l$.

Además, por hipótesis cuando k tiende a infinito, $\|A_k - C'(y^k)\|$ tiende a cero y $C'(y^k)$ tiende a $C'(\bar{x})$. Por tanto, tomando límites en la igualdad

$$p_k - y^k = (-\nabla f(y^k)) - \sum_{i \in \bar{I}} (a_i^k - \nabla C_i(y^k) + \nabla C_i(y^k)) \lambda_{k,i} - \sum_{i \in \bar{I}_U} \mu_{k,i}^u e_i - \sum_{i \in \bar{I}_L} \mu_{k,i}^l (-e_i),$$

$\mu_{k,i}^u \geq 0$, $\mu_{k,i}^l \geq 0$, $\mu_{k,i}^u (U_i - p_{k,i}) = 0$, para $i \in \bar{I}_U$ y $\mu_{k,i}^l (p_{k,i} - L_i) = 0$, $i \in \bar{I}_L$, se obtiene

$$0 = (-\nabla f(\bar{x})) - \sum_{i \in \bar{I}} \nabla C_i(\bar{x}) \lambda_i - \sum_{i \in \bar{I}_U} \mu_i^u e_i - \sum_{i \in \bar{I}_L} \mu_i^l (-e_i),$$

$\mu_i^u \geq 0$, $\mu_i^l \geq 0$, $\mu_i^u (U_i - \bar{x}_i) = 0$, para $i \in \bar{I}_U$ y $\mu_i^l (\bar{x}_i - L_i) = 0$, $i \in \bar{I}_L$. Por lo tanto, \bar{x} es un punto que satisface las condiciones de primer orden de Karush-Kuhn-Tucker (KKT).

Si $\{S_k\}_{k \in K_3}$ no es acotada, dividiendo por S_k se tiene,

$$\frac{p_k - y^k}{S_k} = \frac{-\nabla f(y^k)}{S_k} - \sum_{i \in \bar{I}} (a_i^k - \nabla C_i(y^k) + \nabla C_i(y^k)) \frac{\lambda_{k,i}}{S_k} - \sum_{i \in \bar{I}_U} \frac{\mu_{k,i}^u}{S_k} e_i - \sum_{i \in \bar{I}_L} \frac{\mu_{k,i}^l}{S_k} (-e_i),$$

$$\frac{\mu_{k,i}^u}{S_k} \geq 0 \text{ y } \frac{\mu_{k,i}^l}{S_k} \geq 0.$$

Tomando límite para $k \in K_3$ se obtiene

$$0 = - \sum_{i \in \bar{I}} \nabla C_i(\bar{x}) \lambda_i - \sum_{i \in \bar{I}_U} \mu_i^u e_i - \sum_{i \in \bar{I}_L} \mu_i^l (-e_i),$$

$\mu_i^u \geq 0$ y $\mu_i^l \geq 0$, con coeficientes no todos nulos, pues para cada k hay un coeficiente que vale 1 o -1, entonces existe una subsucesión de índices en K_3 que tiende al valor 1 o -1.

Por tanto, lo anterior indica que los vectores en el conjunto de índices I , I_U , I_L son positivamente dependientes en \bar{x} . Como por hipótesis, en \bar{x} se cumple la condición de calidad CPLD, en un entorno de ese punto en particular los vectores en I , I_U e I_L , deberían ser linealmente dependientes. Esa consecuencia, bajo la hipótesis de no acotación de S_k , es absurda, ya que existe una sucesión de puntos en el entorno de \bar{x} , donde los vectores asociados a I , I_U e I_L son linealmente independientes.

Por tanto, vale que cuando k tiende a infinito se cumple en \bar{x} la condición KKT bajo las hipótesis dadas.

□

4.6. Detalles de implementación

Una de las más atractivas características de los métodos de Restauración Inexacta es la libertad para elegir los métodos para resolver la fase de factibilidad y la fase de optimalidad. Ello permite usar métodos diferentes según la estructura o característica de cada problema.

4.6.1. Fase de Factibilidad.

En esta fase, resolvemos, en cada iteración del Algoritmo 3 IR-DFO, en forma aproximada el problema de factibilidad:

$$\text{mín } h(y)^2, \text{ sobre } x \in \Omega, \|y - x^k\| \leq \beta h(x^k),$$

siendo $h(x) = \|C(x)\|_2$. Para resolver el problema de minimizar una función con restricciones de cotas se conocen distintos métodos sin derivadas (ver [35]). Así, se puede usar para resolver este subproblema varios métodos, incluyendo métodos de búsqueda directa por patrones [71], métodos simplex [82] o algoritmos basados en interpolación polinomial [12, 92]. Estos últimos basados en el uso de aproximaciones cuadráticas de la función objetivo y técnicas de región de confianza.

Por ejemplo, la aplicación de BOBYQA [92] o TRB-Powell [12] para resolver esta fase en cada iteración del método IR-DFO (Paso I), requiere iterar hasta obtener el descenso requerido por la condición $\|C(y)\| \leq \alpha \|C(x)\|$, o hasta que se cumpla la condición de terminación propia del algoritmo usado.

Otra implementación, sin usar un método conocido, puede seguir el esquema siguiente:

ALGORITMO 2. Paso de Restauración

Dado x_c , $\Sigma_c = \{x_c\} \cup \{y^1, y^2, \dots, y^n\}$, Δ_c , $m(x_c + s)$:

$$x^+ = x_c, \Sigma_+ = \Sigma_c, \Delta_+ = \Delta_c.$$

- (i): Hallar el punto de Σ_+ más alejado del iterado actual: $y^{l_{out}}$.
- (ii): Calcular el paso s que resuelve el subproblema y evaluar la función $h(x)$ en el punto hallado.
- Si $x^+ + s$ satisface la condición requerida en el Paso I, definir $y = x^+ + s$, eliminar el satélite $y^{l_{out}}$ e incluir el nuevo y al Σ_+ . Terminar.
 - Si no, hacer una búsqueda en la dirección s :
 - Si existe un $y = x^+ + ts$, $0 < t < 1$, que cumple la condición del Paso I, eliminar $y^{l_{out}}$ como punto satélite en Σ_+ e incluir y . Terminar.
 - Si existe un $x^+ + \tau s$, tal que $h(x^+ + \tau s) < h(x_c)$, definir $x^+ = x^+ + \tau s$, eliminar $y^{l_{out}}$ del conjunto satélite e incluir x^+ e ir al paso (iii).
 - Si no, reemplazar el punto satélite $y^{l_{out}}$ por un punto conveniente $x^+ + \tau s$.
- (iii): Adaptar el modelo $m(x^+ + s)$, $\Sigma_+ = \{x^+\} \cup \{y^1, y^2, \dots, y^n\}$, y Δ_+ , ir al paso (i).

Como hay libertad para elegir el algoritmo sin derivadas para aplicar al subproblema con restricciones de cotas, hemos testado dos algoritmos sin derivadas: el software BOBYQA [92], y el TRB-Powell [12], basados en aproximaciones cuadráticas de la función $h(x)^2 = \sum_{i=1}^m C_i(x)^2$ y región de confianza.

4.6.2. Fase de Optimalidad.

Por otro lado, en la fase de optimización el Algoritmo 3 IR-DFO resuelve aproximadamente la minimización de la función objetivo, en una región de confianza, con restricciones lineales de igualdad y cotas en las variables.

Para problemas de optimización con restricciones lineales existen algoritmos conocidos sin derivadas. En particular, el método GSS introducido por Kolda, Lewis y Torczon [72, 69], también el SID-PSM [39, 37], que es una implementación en MATLAB, para problemas con restricciones lineales de desigualdad. La diferencia principal entre el SID-PSM y otros softwares de uso libre, sin derivadas del tipo de búsqueda por patrones, como el NOMAD/NOMADm [2] y el APPSPACK [84], es el uso de las evaluaciones previas guardadas de la función objetivo.

En particular en el SID-PSM, las evaluaciones previas de la función objetivo, se usan en un iterado corriente para calcular el gradiente simplex, o el Hessiano simplex cuando se requiere, lo cual mejora la eficiencia de esta clase de métodos. Se reporta en [36, 39], que el uso del

gradiente simplex puede conducir a una significativa reducción del número de las evaluaciones de la función, y también su uso aporta una estrategia para ordenar los vectores del conjunto de direcciones analizadas en el procedimiento.

En este trabajo, como la fase de optimización está dirigida a resolver la minimización de f en una región de confianza, creemos conveniente implementar un algoritmo especial, usando el esquema propuesto en la sección 4.3, Algoritmo 4. Tal implementación para resolver el subproblema del Paso III de IR-DFO, usa un modelo aproximante de la función objetivo, sobre el tangente aproximado en la región de confianza $\|y - y^k\| \leq \delta_{k,i}$. El modelo usado, podría ser cuadrático o lineal. En nuestra actual implementación usamos un modelo lineal.

Para implementar el Paso III del IR-DFO, hemos usado dos subrutinas del IMSL, la subrutina DLSVRR para realizar la descomposición de valores singulares de A_k , y la subrutina DQPROG para hallar la proyección de $-\nabla f(x)$ cuando hay cotas en las variables. Para calcular el gradiente simplex usamos las direcciones V , de la descomposición en valores singulares SVD, con un paso r_k , de acuerdo al Algoritmo 4.

En la implementación del algoritmo IR-DFO, consideramos los siguientes valores de los parámetros:

$$\alpha = 0.7d0.$$

$$\beta = 100.$$

$$\omega_k = (nr + 1)/(1 + k)^2.$$

$$\epsilon_k = 1/(1 + k)^2.$$

$$\epsilon_f = 10^{-5}.$$

En la aplicación de BOBYQA y TRB-Powell, en la fase de Restauración, usamos los parámetros originales en ambos algoritmos, salvo para el radio inicial de la región de confianza, RHO-BEG= $\min(0.1, \beta\|C(x)\|)$.

4.7. Experimentación numérica

En esta sección, presentamos algunos resultados computacionales obtenidos con una implementación del algoritmo IR-DFO en Fortran 77.

Denominamos la implementación de IR-DFO con BOBYQA, IR-DFOBQA, y la implementación de IR-DFO con TRB-Powell, IR-DFOTRP.

Consideramos que IR-DFO encuentra una solución aceptable de un problema si la norma de la proyección del opuesto del gradiente simplex, $\|g_{fan}\| \leq 10^{-5}$, en un punto x que es suficientemente factible, en el sentido que $\|C(x)\| \leq 10^{-5}$. También, termina el proceso iterativo si la fase de Factibilidad falla a decrecer la norma $\|C(x)\|$ siendo ya menor a 10^{-5} .

Consideramos que el algoritmo falla al resolver un problema en los casos:

Falla 1: cuando no encuentra la condición de terminación en 100 iteraciones del algoritmo principal.

Falla 2: cuando en la fase de Restauración no es posible satisfacer el decrecimiento $\|C(y)\| \leq \alpha\|C(x)\|$, y el valor de $\|C(x)\| > 10^{-5}$.

Falla 3: cuando la cantidad de evaluaciones de la función (EvalF) supera la cantidad máxima de evaluaciones establecida ($fmax = 5000$).

4.7.1. Problemas de Hock-Schittkowski. Se presentan los resultados obtenidos, con nuestra implementación del IR-DFO, con los problemas test de Hock y Schittkowski [62]. Este conjunto de problemas está compuesto por 112 problemas. Hemos resuelto, en esta experimentación preliminar, 33 que tienen restricciones generales y/o restricciones de cajas. Estos problemas presentan distintas dificultades numéricas que se observan en la práctica, por ejemplo, la función objetivo y/o restricciones tienen problemas de escalamiento, problemas con mal condicionamiento, soluciones en puntos no regulares, en puntos donde las restricciones de calidad no se satisfacen, o tienen diferentes soluciones locales o infinitas soluciones óptimas. Estos test permiten analizar interesantes propiedades de problemas con diferentes dificultades que se presentan en la vida real. La dimensión (n) de los problemas corridos en nuestras experiencias varía entre 2 y 10, mientras el número de restricciones varía entre 1 y 6.

En la Tabla 1 se describen las características de los problemas elegidos, sus dimensiones en relación a las variables y restricciones.

La Tabla 1 muestra el nombre, la dimensión del problema (n), la cantidad de restricciones de igualdad (nr), la cantidad de variables con cota inferior solamente (nL), la cantidad de variables con cota superior solamente (nU), la cantidad de variables con ambas cotas ($nL+nU$) y el valor óptimo de la función objetivo (f^*).

Problema HS:	n	nr	nL	nU	$nL + nU$	f^*
5	2	1	-	-	2	-2.21560D+00
6	2	1	-	-	2	0.00000D+00
7	2	1			2	-1.73000D+00
8	2	2			2	-1.00000D+00
9	2	1			2	-0.50000D+00
14	3	2	1			1.39350D+00
18	4	2	2	-	2	5.00000D+00
26	3	1	-	-	3	0.00000D+00
27	3	1	-	-	-	4.00000D-02
32	4	2	4	0	0	1.00000D+00
33	5	2	5	1	1	-4.58580D+00
34	5	2	2	-	3	- 0.83400D+00
35	4	1	4	0	0	0.11111D+00
39	4	2	0	0	0	-1.00000D+00
40	4	3	-	-	0	- 0.25000D+00
41	4	1	-	-	4	1.92590D+00
46	5	2	0	0	0	0.00000D+00
47	5	3	0	0	0	0.00000D+00
48	5	2	0	0	0	0.00000D+00
52	5	3	-	-	0	5.32660D+00
53	5	3	-	-	5	4.09300D+00
55	6	6	4	-	2	6.33333D+00
56	7	4	7	-	0	-3.45600D+00
60	3	1	-	-	3	3.25682D-02
61	3	2	-	-	0	-1.43646D+02
63	3	2	3	-	-	9.61715D+02
77	5	2	-	-	5	0.24150D+00
78	5	3			5	-2.91970D+00
79	5	3			5	7.87768D-02
80	5	3			5	5.39498D-02
81	5	3			5	5.39498D-02
111	10	3	-	-	10	- 4.77610D+01
112	10	3	10	-	-	- 4.77610D+01

TABLA 1. Problemas de Hock-Schittkowski

La Tabla 2 muestra el desempeño de cada algoritmo para los problemas elegidos, con respecto al valor de la función objetivo ($f(x_{end})$) y al número de evaluaciones funcionales requeridas, indicando la relación entre el número de evaluaciones de $C(x)$ (EvalC) y el número de evaluaciones de la función objetivo (EvalF).

Problema		IR-DFOBQA		IR-DFOTRB	
HS	n	EvalC/EvalF	$f(x_{end})$	EvalC/EvalF	$f(x_{end})$
5	2	91/37	-2.21056D+00	94/57	-2.21084D+00
6	2	58/78	2.22279D-08	61/86	1.13584D-06
7	2	66/175	-1.73197D+00	130/166	-1.73203D+00
8	2	129/6	-1.00000D+00	96/5	-1.00000D+00
9	2	119/430	-4.99999D-01	84/44	-5.00001D-01
14	3	111/11	1.39350D+00	144/17	1.39353D+00
18	3	242/17	5.00006D+00	308/101	4.99918D+00
26	3	128/67	8.25334D-05	148/180	5.18823D-05
27	3	103/88	4.00302D-02	91/35	4.00416D-02
32	4	143/13	1.03257D+00	398/44	1.00038D+00
33	5	213/13	-4.58578D+00	160/46	-4.58534D+00
34	5	542/86	-0.47402D+00	969/72	-0.49003D+00
35	5	140/99	0.11375D+00	128/38	0.11208D+00
39	4	1440/224	-1.00032D+00	1142/122	-1.00004D+00
40	4	194/81	-0.25000D+00	247/57	-0.25000D+00
41	4	155/88	1.92990D+00	143/56	1.92364D+00
46	5	613/192	1.38629D-04	391/105	1.87575D-04
47	5	285/205	1.22742D-04	299/205	7.41711D-06
48	5	492/783	1.75504D-06	250/91	1.20194D-05
52	5	717/58	5.32668D+00	526/310	5.33456D+00
53	5	822/613	4.09134D+00	680/464	4.09330D+00
55	6	133/2	6.33330D+00	263/3	6.33328D+00
56	7	998/365	-3.45584D+00	777/223	-3.45604D+00
60	3	386/110	4.13611D-02	151/71	3.25694D-02
61	3	690/138	-1.43184D+02	528/138	-1.43599D+02
63	3	449/739	9.61715D+02	356/22	9.61715D+02
77	5	301/112	0.24607D+00	427/270	0.24154D+00
78	5	150/213	-2.91812D+00	396/293	-2.91949D+00
79	5	196/136	7.88399D-02	304/216	7.87871D-02
80	5	230/9	5.35669D-02	232/10	5.35668D-02
81	5	225/9	5.35653D-02	220/11	5.35654D-02
111	10	823/74	-4.77650D+01	907/101	-4.77691D+01
112	10	4015/1799	-4.73636D+01	1946/600	-4.73698D+01

TABLA 2. IR-DFOBQA vs IR-DFOTRB

Problema		$\ C(x_{end})\ $	
HS	n	IR-DFOBQA	IR-DFOTRB
5	2	3.74480D-06	2.53470D-06
6	2	0.00000D+00	0.00000D+00
7	2	1.19050D-07	5.37679D-06
8	2	1.75649D-06	1.65404D-06
9	2	1.20486D-06	1.72558D-06
14	3	1.65235D-06	3.16411D-06
18	3	1.43588D-06	4.06747D-06
26	3	5.38046D-06	5.09938D-07
27	3	1.67146D-06	1.00939D-08
32	4	5.02984D-06	1.39407D-06
33	5	1.02069D-06	2.43517D-06
34	5	5.95758D-05	5.28458D-05
35	5	1.33807D-06	1.73429D-06
39	4	1.82384D-06	1.59677D-06
40	4	7.33480D-06	4.32512D-06
41	4	6.35654D-06	1.78677D-06
46	5	5.02523D-06	1.30806D-06
47	5	6.12510D-06	5.59864D-06
48	5	8.47764D-06	6.17700D-06
52	5	7.27966D-10	3.14638D-11
53	5	3.09026D-06	1.52701D-06
55	6	6.81039D-06	6.13105D-06
56	7	1.54035D-06	7.51571D-06
60	3	6.57546D-06	2.40603D-06
61	3	2.58698D-06	1.28866D-06
63	3	2.83179D-06	3.13069D-06
77	5	1.17902D-06	5.68880D-06
78	5	3.97461D-06	3.62110D-06
79	5	2.61147D-06	1.06133D-06
80	5	3.35462D-06	2.35462D-06
81	5	6.00920D-05	2.35092D-05
111	10	5.31760D-05	5.39942D-05
112	10	9.55757D-06	6.56257D-06

TABLA 3. Factibilidad alcanzada por los IR-DFOBQA e IR-DFOTBR.

En la Tabla 3 se muestra la factibilidad alcanzada en cada problema ($\|C(x_{end})\|$) por los algoritmos testeados.

Los resultados de las Tablas 2 y 3 nos permiten hacer las siguientes observaciones.

Para los problemas HS34, HS81 y HS111 los dos algoritmos finalizan por Falla 2, esto es, el algoritmo falla en la Fase de Factibilidad por lo que finaliza por no poder encontrar un punto más factible que el hallado en una iteración previa. Ninguno de los problemas finaliza por la Falla 3 para ninguna de las dos implementaciones.

La cantidad de evaluaciones de la función objetivo f requeridas por IR-DFOBQA es menor en 15 de los 33 problemas mientras que para IR-DFOTRB este número es menor en 16 problemas. Cabe destacar que en 2 de los problemas ambas implementaciones obtienen la misma cantidad de evaluaciones funcionales de f . El promedio de evaluaciones requeridas por IR-DFOBQA es de 214 y de 129 para IR-DFOTRB. Los valores funcionales obtenidos para ambas implementaciones son similares.

En cuanto a la cantidad total de evaluaciones realizadas en la Fase de Factibilidad, IR-DFOBQA es menor en 16 de los 33 problemas mientras que para IR-DFOTRB este número es menor en 17 problemas y no se presentan empates. El promedio de evaluaciones requeridas por IR-DFOBQA es de 467 y de 394 para IR-DFOTRB. También, en este caso, los valores funcionales obtenidos para ambas implementaciones son similares. Notar que el problema HS6 finaliza con un punto factible ya que $\|C(x_{end})\| = 0$.

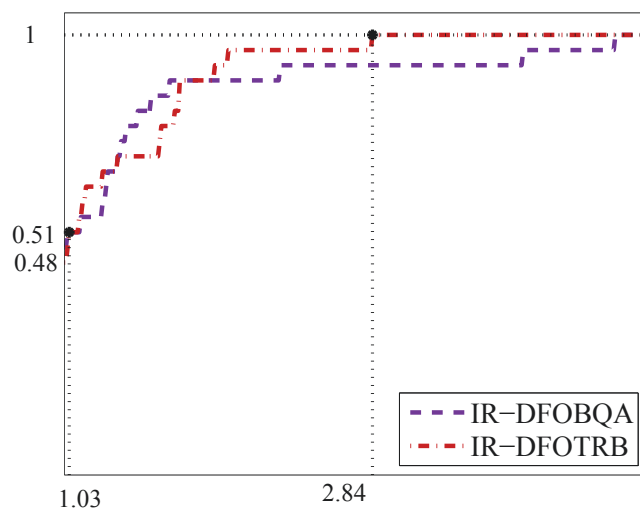


FIGURA 1. Comparación de evaluaciones totales de la función objetivo (f).

En la Figura 1 se muestra el perfil de desempeño que compara el número de evaluaciones funcionales, de la función objetivo (f), realizadas por las implementaciones IR-DFOBQA e IR-DFOTRB. Se puede observar que IR-DFOTRB realiza menos evaluaciones funcionales en el 48% de los problemas mientras que IR-DFOBQA lo hace en el 51%.

Conclusiones y perspectivas

En este trabajo de tesis hemos considerado el problema de minimizar una función no lineal suficientemente regular con restricciones no lineales de igualdad y restricciones de cotas en las variables (cajas), donde las derivadas de todas las funciones involucradas no están disponibles. Este tipo de problemas aparece en diversos problemas de la vida real donde tales funciones no están dadas en forma explícita o los datos que las definen provienen de mediciones físicas, químicas o de experimentos numéricos con alto costo computacional. Las dimensiones de estos problemas no son muy grandes debido a la complejidad de los mismos. Consideramos que es un problema de gran interés en el área de optimización numérica por sus potenciales aplicaciones a diferentes problemas y disciplinas. Como hemos mencionado, actualmente se conocen varios métodos basados en búsqueda de patrones, diferencias finitas y estrategias de región de confianza en combinación con modelos de interpolación. Tomando en cuenta los buenos resultados para el caso diferenciable, hemos trabajado sobre el último enfoque que utiliza modelos de interpolación lineales y cuadráticos.

En una primera etapa de este trabajo se consideró un problema irrestricto o sólo con restricciones de cotas en las variables. Consideramos como referencia el método NEWUOA [90], desarrollado por M. J. D. Powell, el cual ha mostrado ser el más eficiente para funciones suaves según el artículo de Moré y Wild [81]. Este algoritmo de Powell fue propuesto para resolver problemas irrestrictos donde, en cada iteración, construye un modelo cuadrático de la función objetivo, a partir de la información de las aproximaciones anteriores, y busca minimizadores usando el método de región de confianza. Este algoritmo es bastante complejo dado que debe actualizar la región de confianza y el conjunto de puntos de interpolación de modo de obtener un conjunto adecuado geoméricamente. Pensando en resolver con restricciones de cotas, se adaptó este algoritmo usando norma infinito y se utilizó una estrategia de conjuntos activos para resolver el subproblema cuadrático explorando las caras de la caja. Esta idea nos permitió extender nuestro método, de una manera natural, a problemas con restricciones de caja. El marco teórico de la estrategia del método de conjuntos activos nos permitió asegurar

la convergencia de nuestro método. Se realizó una implementación y experimentos numéricos con comparaciones con otros algoritmos similares (NEWUOA y BOBYQA) mostrando el muy buen desempeño de la nueva propuesta.

El problema con restricciones generales es mucho más complicado en el caso sin derivadas que en el caso con derivadas. Nosotros consideramos que tanto la función objetivo como sus restricciones no poseen derivadas disponibles. El método propuesto se basa en el método de Restauración Inexacta, el cual ha mostrado ser muy exitoso en programación no lineal cuando está disponible la información de primer orden de la función objetivo y de las restricciones. La complejidad, por la imposibilidad de usar derivadas, hace que obtener resultados teóricos sea una tarea difícil y ha sido una de las partes más importantes de esta tesis, junto con la respectiva implementación. Bajo adecuadas hipótesis, se mostraron resultados de buena definición del algoritmo y resultados de convergencia a puntos factibles que satisfacen adecuadas condiciones de optimalidad. La implementación incluye diferentes subalgoritmos para obtener una mejor aproximación en cada iteración. Se realizaron diferentes experimentos numéricos obteniéndose resultados muy prometedores lo cual motiva a seguir trabajando en esta dirección. Es importante destacar que este nuevo método basado en Restauración Inexacta presenta una alternativa muy sólida y atractiva para resolver problemas de optimización general sin usar derivadas de la función objetivo ni de las restricciones, resultando en un aporte significativo en el área de Optimización.

Finalmente indicamos algunas posibles líneas de investigación en las que planeamos continuar trabajando en el futuro:

- Analizar la convergencia de IR-DFO con condiciones de calidad más débiles, tales como RCPLD de Andreani, Haeser, Schuverdt y Silva [10] o lo cual implica condiciones de optimalidad más fuertes.
- Reemplazar, en la implementación computacional de la fase de minimización del algoritmo de restauración inexacta sin derivadas, las aproximaciones lineales de $f(x)$ por aproximaciones cuadráticas. También estudiar la posible aplicación del solver SID-PSM [37, 39] para resolver la fase de optimalidad de IR-DFO, y realizar comparaciones de los resultados obtenidos.
- Adaptar el algoritmo IR-DFO usando otras versiones de la metodología de dos fases para el problema de optimización general [52, 45].

- Adaptar el mismo algoritmo usando la función lagrangiana en el paso de minimización, estimando los multiplicadores convenientemente.
- Recientemente, en el año 2010, Liuzzi, Lucidi y Sciandrone [73] presentaron un algoritmo que usa penalización secuencial para un problema de optimización general sin usar derivadas. Sería interesante comparar nuestro algoritmo con este método, dado que ambos convergen secuencialmente a un punto factible.
- Aplicar el algoritmo de restauración inexacta a un problema de la vida real.

Bibliografía

1. J. Abadie and J. Carpentier, *Generalization of the Wolfe reduced-gradient method to the case of nonlinear constraints*, Academic Press, New York, 1968. [47](#)
2. M.A. Abramson, *NOMADm version 4.6 user's guide*, (2007). [3](#), [83](#)
3. M.A. Abramson and Ch. Audet, *Convergence of mesh adaptive direct search to secondorder stationary points*, SIAM J. on Optimization **17** (2006), 606–619. [3](#)
4. M.A. Abramson, Ch. Audet, Jr. J.E. Dennis, and S. Le Digabel, *Orthomads: A deterministic mads instance with orthogonal directions*, SIAM J. on Optimization **20** (2009), 948–966. [3](#)
5. M.A. Abramson, O.A. Brezhneva, J.E. Dennis, and R.L. Pingel, *Pattern search for linearly constrained minimization in the presence of degeneracy*, Optimization Methods and Software **3** (2008), no. 23, 297–319. [60](#), [61](#)
6. P. Alberto, F. Nogueira, H. Rocha, and L. Vicente, *Pattern search methods for user-provided points: Application to molecular geometry problems*, Journal on Optimization **14** (2004), no. 4, 1216–1236. [59](#)
7. R. Andreani, E. Birgin, J.M. Martínez, and M.L. Schuverdt, *Augmented lagrangian methods under the constant positive linear dependence constraint qualification*, Mathematical Programming **111** (2007), no. 1, 5–32. [33](#), [79](#)
8. ———, *On augmented lagrangian methods with general lower-level constraints*, SIAM Journal on Optimization **18** (2007), 1286–1309. [33](#)
9. R. Andreani, G. Haeser, and J.M. Martínez, *On sequential optimality conditions for smooth constrained optimization*, Optimization **60** (2011), no. 5, 627–641. [51](#)
10. R. Andreani, G. Haeser, M.L. Schuverdt, and P.J.S. Silva, *A relaxed constant positive linear dependence constraint qualification and applications*, Mathematical Programming **135** (2012), no. 1-2, 255–273. [92](#)
11. R. Andreani, J.M. Martínez, and M.L. Schuverdt, *On the relation between the constant positive linear dependence condition and quasinormality constraint qualification*, Journal of Optimization Theory and Applications **125** (2005), 473–485. [79](#)
12. M.B. Arouxét, N. Echebest, and E.A. Pilotta, *Active-set strategy in Powell's method for optimization without derivatives*, Computational and Applied Mathematics **30** (2011), 171–196. [2](#), [31](#), [47](#), [82](#), [83](#)
13. Ch. Audet, V. Bécharde, and S. Le Digabel, *Nonsmooth optimization through mesh adaptive direct search and variable neighborhood search*, J. of Global Optimization **41** (2008), 299–318. [3](#)
14. Ch. Audet and Jr. J.E. Dennis, *Mesh adaptive direct search algorithms for constrained optimization*, SIAM J. on Optimization **17** (2006), 188–217. [3](#)

15. ———, *A progressive barrier for derivative-free nonlinear programming*, SIAM J. on Optimization **20** (2009), 445–472. [3](#)
16. Ch. Audet and D. Orban, *Finding optimal algorithmic parameters using derivative-free optimization*, SIAM J. on Optimization **17** (2006), 642–664. [3](#)
17. M. Avriel and M. Rom, *Properties of the Sequential Gradient-Restoration Algorithm (SGRA), Part 1: Introduction and Comparison with Related Methods*, Journal of Optimization Theory and Applications **62** (1989), 77–98. [47](#)
18. ———, *Properties of the Sequential Gradient-Restoration Algorithm (SGRA), Part 2: Convergence Analysis*, Journal of Optimization Theory and Applications **62** (1989), 99–126. [47](#)
19. V.K. Basapur, A. Miele, and E.M. Sims, *Sequential Gradient-Restoration Algorithm for Mathematical Programming Problems with Inequality Constraints, Part 1, Theory*, Rice University, Aero-Astronautics Report (1983), no. 168. [47](#)
20. F. Berghen and H. Bersini, *CONDOR: a new parallel, constrained extension of Powell's UOBYQA algorithm: Experimental results and comparison with the DFO algorithm*, Journal of Computational and Applied Mathematics **181** (2005), no. 1, 157–175. [4](#)
21. D.P. Bertsekas, *Nonlinear Programming, 2nd edition*, Athena Scientific, Belmont, Massachusetts, 1999. [80](#)
22. E. Birgin, R. Castillo, and J.M. Martínez, *Numerical comparison of augmented lagrangian algorithms for nonconvex problems*, Computational Optimization and Applications **31** (2005), 31–56. [33](#)
23. E. Birgin and J.M. Martínez, *Large-scale active-set box-constrained optimization method with spectral projected gradients*, Computational Optimization and Applications **23** (2002), no. 1, 101–125. [31](#), [32](#), [33](#), [34](#)
24. ———, *Local Convergence of an Inexact-Restoration Method and Numerical Experiments*, Journal of Optimization Theory and Applications **127** (2005), no. 2, 229–247. [47](#)
25. ———, *Improving ultimate convergence of an augmented lagrangian method*, Optimization Methods and Software **23** (2008), 177–195. [33](#)
26. J. Borggaard and J. Burns, *A PDE sensitivity equation method for optimal aerodynamic design*, J. Comput. Phys. **13** (1997), no. 6, 366–384. [3](#)
27. L.F. Bueno, A. Friedlander, J.M. Martínez, and F.N.C. Sobral, *Inexact restoration method for derivative-free optimization with smooth constraints*, Available in Optimization Online (2011). [5](#)
28. B. Colson and P. Toint, *A derivative-free algorithm for sparse unconstrained optimization problems*, In A. H. Siddiqi and M. Kocvara, editors, Trends in Industrial and Applied Mathematics, Applied Optimization, Dordrecht, The Netherlands (2002), 131–149. [4](#)
29. ———, *Optimizing partially separable functions without derivatives*, Optimization Methods and Software **20** (2005), 493–508. [4](#)
30. P.D. Conejo, E.W. Karas, A.A. Ribeiro, L.G. Pedroso, and M. Sachine, *Global convergence of trust-region algorithms for constrained minimization without derivatives*, (2012). [5](#)

-
31. A. Conn, K. Scheinberg, and P. Toint, *On the convergence of derivative-free methods for unconstrained optimization*, Approximation Theory and Optimization: Tributes to M. Powell, M. Buhmann and A. Iserles Eds., Cambridge University Press, Cambridge, UK (1997), 83–108. [4](#), [7](#), [27](#)
 32. ———, *Recent progress in unconstrained nonlinear optimization without derivatives*, Mathematical Programming **79** (1997), 397–414. [4](#), [7](#)
 33. ———, *A derivative free optimization algorithm in practice*, In Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO (1998). [4](#), [7](#), [22](#)
 34. A. Conn, K. Scheinberg, and L. Vicente, *Global convergence of general derivative-free trust-region algorithms to first and second order critical points*, SIAM Journal on Optimization **20** (2009), 387–415. [27](#)
 35. ———, *Introduction to Derivative-Free Optimization*, SIAM, 2009. [3](#), [14](#), [15](#), [18](#), [19](#), [20](#), [21](#), [25](#), [27](#), [28](#), [47](#), [53](#), [59](#), [82](#)
 36. A.L. Custódio, J.E. Dennis Jr., and L. Vicente, *Using simplex gradients of nonsmooth functions in direct search methods*, IMA Journal of Numerical Analysis **28** (2008), no. 4, 770–784. [83](#)
 37. A.L. Custódio, H. Rocha, and L. Vicente, *Incorporating minimum Frobenius norm models in direct search*, Computational Optimization and Applications **46** (2010), 265–278. [3](#), [60](#), [83](#), [92](#)
 38. A.L. Custódio and L. Vicente, *Using Sampling and Simplex Derivatives in Pattern Search Methods (complete numerical results)*, (2006). [59](#)
 39. ———, *Using Sampling and Simplex Derivatives in Pattern Search Methods*, SIAM J. on Optimization **18** (2007), no. 2, 537–555. [3](#), [60](#), [65](#), [83](#), [92](#)
 40. C. Davis, *Theory of Positive Linear Dependence*, American Journal of Mathematics **76** (1954), 733–746. [59](#)
 41. J.E. Dennis and V. Torczon, *Direct search methods on parallel machines*, SIAM Journal on Optimization **1** (1991), 448–474. [3](#)
 42. M.A. Diniz-Ehrhardt, J.M. Martínez, and L.G. Pedroso, *Derivative-free methods for nonlinear programming with general lower-level constraints*, Computational and Applied Mathematics **30** (2011), 19–52. [5](#), [47](#)
 43. E. Dolan and J. Moré, *Benchmarking optimization software with performance profiles*, Mathematical Programming **91** (2002), 201–213. [38](#), [41](#)
 44. G. Fasano, J. Morales, and J. Nocedal, *On the geometry phase in model-based algorithms for derivative-free optimization*, Optimization Methods and Software **24** (2009), no. 1, 145–154. [4](#)
 45. A. Fischer and A. Friedlander, *A new line search inexact restoration approach for nonlinear programming*, Computational Optimization and Applications **46** (2010), no. 2, 333–346. [47](#), [92](#)
 46. A. Friedlander and J.M. Martínez, *On the maximization of a concave quadratic function with box constraints*, SIAM Journal on Optimization **4** (1994), no. 1, 177–192. [32](#)
 47. J.Ch. Gilbert, *Automatic differentiation and iterative processes*, Optimization Methods and Software **1** (1992), no. 1, 13–21. [3](#)
 48. Ph.E. Gill, W. Murray, M.A. Saunders, and M.H. Wright, *Computing forward-difference intervals for numerical optimization*, SIAM Journal on Scientific and Statistical Computing **4** (1983), no. 2, 310–321. [3](#)

49. Ph.E. Gill, W. Murray, and M.H. Wright, *Practical optimization*, Elsevier, 1981. [3](#)
50. M.S. Gockenbach, D.R. Reynolds, P. Shen, and W.W. Symes, *Efficient and automatic implementation of the adjoint state method*, ACM Trans. Math. Softw. **28** (2002), 22–44. [3](#)
51. F.A.M. Gomes, M.C. Maciel, and J.M. Martínez, *Nonlinear programming algorithms using trust regions and augmented Lagrangians with nonmonotone penalty parameters*, 1997, pp. 161–200. [50](#)
52. C.C. Gonzaga, E.W. Karas, and M. Vanti, *A globally convergent filter method for nonlinear programming*, SIAM Journal on Optimization **14** (2003), no. 3, 646–669. [47](#), [48](#), [92](#)
53. N.I. Gould, D. Orban, and P. Toint, *CUTEr and SifDec: A constrained and unconstrained testing environment, revisited*, ACM Transactions on Mathematical Software **29** (2003), no. 4, 373–394. [35](#)
54. N.I. Gould and P. Toint, *Nonlinear programming without a penalty function or a filter*, Mathematical Programming **122** (2010), no. 1, 155–196. [47](#)
55. S. Gratton, P. Toint, and A. Tröltzsch, *An active-set trust region method for derivative-free nonlinear bound-constrained optimization*, Technical Report. CERFACS. Parallel Algorithms Team. **10** (2010), 1–30. [4](#), [21](#), [34](#), [36](#), [46](#)
56. G.A. Gray and T.G. Kolda, *Algorithm 8xx: APPSPACK 4.0: Asynchronous parallel pattern search for derivative-free optimization*, ACM Transactions on Mathematical Software, To appear (Accepted Nov. 2005). [3](#)
57. A. Griewank, *Computational differentiation and optimization*, J.R. Birge and K.G. Murty, editors, Mathematical Programming : State of the Art 1994. The University of Michigan, Michigan (1994), 102–131. [3](#)
58. ———, *Evaluating derivatives: principles and techniques of algorithmic differentiation*, SIAM, Philadelphia, PA, USA, 2000. [3](#)
59. A. Griewank and G.F. Corliss, *Automatic differentiation of algorithms: Theory, implementation, and application*, SIAM, Philadelphia, PA, 1991. [3](#)
60. A. Griewank and P. Toint, *On the unconstrained optimization of partially separable functions*, In Nonlinear Optimization, Academic Press, New-York (1981). [4](#)
61. J.D. Griffin and T.G. Kolda, *Asynchronous parallel generating set search for linearly-constrained optimization*, Tech. report, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, July 2006. [3](#)
62. W. Hock and K. Schittkowski, *Nonlinear programming codes*, vol. 187, Lecture Notes in Economics and Mathematical Systems, Springer, 1980. [34](#), [36](#), [85](#)
63. P.D. Hough, T.G. Kolda, and V. Torczon, *Asynchronous parallel pattern search for nonlinear optimization*, SIAM Journal on Scientific Computing **23** (2001), no. 1, 134–156. [3](#)
64. J. Dennis Jr. and R. Schnabel, *Numerical methods for unconstrained and nonlinear equations*, vol. 16, SIAM, 1996. [3](#), [17](#)
65. E. Karas, E.A. Pilotta, and A. Ribeiro, *Numerical comparison of merit function with filter criterion in inexact restoration algorithms using hard-spheres problems*, Computational Optimization and Applications **44** (2009), no. 3, 427–441. [48](#)

-
66. C.T. Kelley, *Iterative methods for optimization*, SIAM Frontiers in Applied Mathematics (1999). 19, 59
 67. T.G. Kolda, *Revisiting asynchronous parallel pattern search for nonlinear optimization*, SIAM Journal on Optimization **16** (2005), no. 2, 563–586. 3
 68. T.G. Kolda, R.M. Lewis, and V. Torczon, *Optimization by direct search: New perspectives on some classical and modern methods*, SIAM Review **45** (2003), no. 3, 385–482. 3
 69. ———, *Stationarity Results for Generating Set Search for Linearly Constrained Optimization*, SIAM Journal on Optimization **17** (2006), no. 4, 943–968. 3, 5, 83
 70. T.G. Kolda and V. Torczon, *On the convergence of asynchronous parallel pattern search*, SIAM Journal on Optimization **14** (2004), no. 4, 939–964. 3
 71. R.M. Lewis and V. Torczon, *Pattern search algorithms for bound constrained minimization*, SIAM Journal on Optimization **9** (1999), 1082–1099. 82
 72. ———, *Pattern search methods for linearly constrained minimization*, SIAM Journal on Optimization **10** (2000), 917–941. 3, 5, 60, 61, 83
 73. G. Liuzzi, S. Lucidi, and M. Sciandrone, *Sequential penalty derivative-free methods for nonlinear constrained optimization*, Journal on Optimization **20** (2010), no. 5, 2614–2635. 93
 74. M. Marazzi and J. Nocedal, *Wedge trust region methods for derivative free optimization*, Mathematical Programming **91** (2002), no. 2, 289–305. 4, 21, 22
 75. J.M. Martínez, *Two-Phase Model Algorithm with Global Convergence for Nonlinear Programming*, Journal of Optimization Theory and Applications **96** (1998), 397–436. 47
 76. ———, *Inexact-restoration method with lagrangian tangent decrease and new merit function for nonlinear programming*, Journal of Optimization Theory and Applications **111** (2001), 39–58. 47, 48, 51
 77. J.M. Martínez and E.A. Pilotta, *Inexact Restoration algorithms for constrained optimization*, Journal of Optimization Theory and Applications **104** (2000), no. 1, 135–163. 2, 47, 48, 49, 50, 51, 54, 62, 63
 78. ———, *Inexact Restoration methods for nonlinear programming: advances and perspectives*, Applied Optimization Series, Qi, Teo, Yang, Springer Netherlands, 2005. 2
 79. J.M. Martínez and B.F. Svaiter, *A practical optimality condition without constraint qualifications for nonlinear programming*, Journal of Optimization Theory and Applications **118** (2003), 117–133. 51
 80. M. Molga and C. Smutnicki, *Test functions for optimization needs*, Available at <http://www.zsd.ict.pwr.wroc.pl/files/docs/functions.pdf> (2005). 34, 36
 81. J. Moré and S. Wild, *Benchmarking derivative-free optimization algorithms*, SIAM Journal on Optimization **20** (2009), no. 1, 172–191. 34, 35, 36, 91
 82. J. Nelder and R. Mead, *A simplex method for function minimization*, Computer Journal **7** (1965), 308–313. 3, 82
 83. J. Nocedal and S.J. Wright, *Numerical optimization*, Springer Series in Operations Research, Springer, Berlin, Heidelberg, 1999. 3, 10, 11, 12, 13, 14, 33, 50
 84. T.D. Plantenga, *HOPSPACK 2.0 User Manual*, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, SAND2009- 6265 (2009). 3, 83

85. M.J.D. Powell, *A direct search optimization method that models the objective and constraint functions by linear interpolation*, in *Advances in Optimization and Numerical Analysis* (1994), 51–67. [28](#)
86. ———, *On the Lagrange functions of quadratic models that are defined by interpolation*, *Optimization Methods and Software* **16** (2001), 289–309. [4](#), [24](#)
87. ———, *UOBYQA: Unconstrained optimization by quadratic approximation*, *Mathematical Programming* **92** (2002), no. 3, 555–582. [4](#), [24](#)
88. ———, *On trust region methods for unconstrained minimization without derivatives*, *Mathematical Programming* **97** (2003), no. 3, 605–623. [4](#), [24](#)
89. ———, *Least Frobenius norm updating of quadratic models that satisfy interpolation conditions*, *Mathematical Programming* **100** (2004), no. 1, 183–215. [4](#), [7](#), [24](#), [25](#)
90. ———, *The NEWUOA software for unconstrained optimization without derivatives*, *Nonconvex Optimization and Its Applications*, vol. 83, Springer US, 2006. [2](#), [4](#), [7](#), [24](#), [31](#), [34](#), [35](#), [41](#), [91](#)
91. ———, *Developments of NEWUOA for minimization without derivatives*, *IMA Journal of Numerical Analysis* **28** (2008), no. 4, 649–664. [2](#), [4](#), [7](#), [27](#)
92. ———, *The BOBYQA algorithm for bound constrained optimization without derivatives*, Technical report, Department of Applied Mathematics and Theoretical Physics, Cambridge University, Cambridge, England. (2009). [4](#), [27](#), [31](#), [36](#), [47](#), [82](#), [83](#)
93. ———, *On the convergence of trust region algorithms for unconstrained minimization without derivatives*, Technical Report, University of Cambridge, Cambridge, England (2011). [4](#), [27](#), [29](#)
94. ———, *On derivative-free optimization with linear constraints*, In 21st ISMP, Berlin, Germany (2012). [4](#)
95. R.G. Regis, Chr.A. Shoemaker, and S.M. Wild, *Orbit: optimization by radial basis function interpolation in trust-regions*, *SIAM J. on Scientific Computing* **30** (2008), no. 6, 3197–3219. [4](#)
96. J.B. Rosen, *Two-Phase Algorithm for Nonlinear Constraint Problems*, (1978), 97–124. [47](#)
97. K. Scheinberg and P. Toint, *Self-correcting geometry in model-based algorithms for derivative-free unconstrained optimization*, *SIAM Journal on Optimization* **20** (2010), no. 6, 3512–3532. [4](#)
98. V. Torczon, *On the convergence of pattern search algorithms*, *SIAM Journal on Optimization* **7** (1997), no. 1, 1–25. [3](#)
99. K.E. Vugrin, *On the effect of numerical noise in simulation-based optimization*, Ph.D. thesis, 2003. [3](#)
100. D. Winfield, *Functions and functional optimization by interpolation in data tables*, Ph.D. thesis, 1969. [4](#)
101. ———, *Functions minimization by interpolation in a data table*, *IMA Journal of Applied Mathematics* **12** (1973), no. 3, 339–347. [4](#)