

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE CIENCIAS

**UN MÉTODO DE OPTIMIZACIÓN DE TIPO BUNDLE NO
DIFERENCIABLE APLICADO AL FLUJO DE
MATERIALES VISCOPLÁSTICOS**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE MATEMÁTICO**

PROYECTO DE INVESTIGACIÓN

DIEGO XAVIER REYES ANDRADE

diego.reyes@epn.edu.ec

DIRECTOR: SERGIO ALEJANDRO GONZÁLEZ ANDRADE, Ph.D.

sergio.gonzalez@epn.edu.ec

QUITO, MARZO 2022

Declaración

Yo, Diego Xavier Reyes Andrade, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Escuela Politécnica Nacional puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de la Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.



Diego Xavier Reyes Andrade

Certificación

Certifico que el presente trabajo fue desarrollado por Diego Xavier Reyes Andrade, bajo mi supervisión.

**SERGIO
ALEJANDRO
GONZALEZ
ANDRADE** Firmado
digitalmente por
SERGIO ALEJANDRO
GONZALEZ
ANDRADE
Fecha: 2022.03.15
10:18:34 -05'00'

Sergio Alejandro González Andrade, Ph.D.
Director

AGRADECIMIENTOS

A Dios y a la Virgen Dolorosa por su amor infinito, por todas las bendiciones que recibo cada día. También, por poner en mi vida a las personas más maravillosas que pueden existir.

A mis padres, Raúl Fernando y Elena Patricia, por ser el ejemplo perfecto de esfuerzo y sacrificio, por su amor incondicional y brindarme fuerzas cuando más las he necesitado. Sin ustedes simplemente, no estaría aquí, los amo.

A mis hermanos, Raúl Andrés y Pablo Martín, por ser los mejores hermanos del mundo y alegrar mis días con sus chistes y anécdotas. Gracias por estar a mi lado tanto en lo bueno como en lo malo, siempre están ahí para apoyarme.

A mi tía, mi abuelita, mis tíos y mis primos, por su inmenso cariño y gran apoyo. Soy afortunado de tener personas como ustedes en mi vida.

A mi tutor, Sergio, por el tiempo y esfuerzo dedicado en este proyecto. También, por convertirse en un modelo a seguir en el ámbito profesional y mostrarme las maravillas de la investigación científica.

A mis hermanos de otra madre, Francisco y Roberth, por enseñarme la verdadera amistad y permitirme disfrutar de momentos maravillosos a su lado.

A mi mejor amiga, Gaby, por brindarme mucho cariño así como apoyarme en cada instante. Gracias por ser parte de mi vida, con personas como tú, tengo la dicha de poder lograr todo lo que me propongo.

A la Banda de los Gatos, por ser ese grupo de amigos incondicionales que me han acompañado en mi carrera universitaria. Nunca olvidaré todas las risas, logros y festejos que hemos compartido juntos.

A mi perrhijo, Rocket Samuel, por permitirme amarlo y mimarlo cada día. Aunque hayan días en donde me sienta cansado, siempre tendré tiempo para ti porque tú siempre estas para mí. ¡Gracias por ser mi compañía de cuatro patas!

Diego

DEDICATORIA

Al Proyecto EC-RRHA.

Gracias por ayudarme a conseguir mis objetivos personales y profesionales.

Los amo de aquí al ∞ .

Índice general

Resumen	XI
Abstract	XII
1. Introducción	1
2. Preliminares	3
2.1. Conceptos y Resultados Generales	3
2.1.1. Notaciones preliminares	3
2.1.2. Espacios de funciones	3
2.1.3. Funcionales Convexos	5
2.2. Flujo viscoplástico de Herschel-Bulkley y su problema de optimización asociado	6
2.2.1. Modelo constitutivo de Herschel-Bulkley	7
2.2.2. Formulación como un problema de optimización	8
2.3. Métodos Quasi-Newton	9
2.3.1. Actualización BFGS	11
2.4. Introducción a la Optimización No Diferenciable	12
3. Métodos Bundle	15
3.1. Método General Bundle	15
3.2. Método Bundle Quasi-Newton	20
3.2.1. Regularización de Moreau-Yosida	20
3.2.2. Subalgoritmo Bundle para la aproximación de la regularización de Moreau-Yosida	24

4. Algoritmo no suave para la resolución del problema de flujo	30
4.1. Estudio analítico del problema de flujo	30
4.2. Discretización del problema por elementos finitos	37
4.2.1. Discretización del subdiferencial del funcional asociado al problema de flujo	41
4.3. Algoritmo tipo Bundle no diferenciable	42
4.3.1. Regularización de Moreau-Yosida del funcional discretizado	43
4.3.2. Método Bundle para la resolución del problema auxiliar	43
4.3.3. Actualización de tipo Newton para el problema regularizado	47
4.3.4. Búsqueda lineal de la longitud de paso	47
4.4. Análisis del algoritmo	52
4.4.1. Convergencia del método hacia al mínimo	52
4.4.2. Orden de convergencia del algoritmo	55
5. Resultados	59
5.1. Ensamble del algoritmo	59
5.2. Experimentos y resultados numéricos	62
5.2.1. Caso $1 < p < 2$	62
5.2.2. Caso $p > 2$	71
Conclusiones y Recomendaciones	81
Referencias	83

Índice de figuras

2.1. Bosquejo del comportamiento de distintos tipos de fluidos	6
2.2. Bosquejo del modelo de Herschel-Bulkley	8
3.1. Regularización de Moreau-Yosida de $f(x) = x $	21
4.1. Triangulación regular de $B_1(0,0) \subset \mathbb{R}^2$	38
5.1. Gráfica de la función $ \nabla\psi(x,y) $	61
5.2. Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo del diámetro de la tubería (derecha). Parámetros: $p = 1.75$, $g = 0.2$ y $M = A$. 63	
5.3. Valores de ν obtenidos en cada iteración del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = A$	64
5.4. Velocidad u calculada para $p = 1.5$ y $g = 0.1$ (superior izquierda), $g = 0.2$ (superior derecha) y $g = 0.25$ (inferior izquierda). Perfiles de las velocidades calculadas a lo largo del diámetro de la tubería (inferior derecha).	65
5.5. Valores calculados de ν por el Algoritmo 8. Parámetros: $p = 1.5$ y $M = A$. 66	
5.6. Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo de la diagonal de la tubería (derecha). Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$. 67	
5.7. Valores calculados de ν por el Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$	68
5.8. Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo del diámetro de la tubería (derecha). Parámetros: $p = 1.75$, $g = 0.2$, $M = I$ y $\delta_0 = 0.25$	68
5.9. Valores de ν calculados por el Algoritmo 8 para cada δ_0 . Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$	70

5.10. Velocidad u calculada para $p = 4$ y $g = 0.1$ (superior izquierda), $g = 0.2$ (superior derecha) y $g = 0.3$ (inferior izquierda). Perfiles de las velocidades calculadas a lo largo del diámetro de la tubería (inferior derecha).	72
5.11. Valores de ν obtenidos en cada iteración del Algoritmo 8. Parámetros: $p = 4$ y $M = A$	73
5.12. Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo de la diagonal de la tubería (derecha). Parámetros: $p = 4$, $g = 0.2$ y $M = I$.	74
5.13. Valores de ν obtenidos en cada iteración del Algoritmo 8. Parámetros: $p = 4$, $g = 0.2$ y $M = I$	75
5.14. Velocidad u calculada para $p = 8$ (superior izquierda) y $p = 10$ (inferior izquierda). Perfiles de las velocidades calculadas a lo largo del diámetro de la tubería con $p = 8$ (superior derecha) y $p = 10$ (inferior derecha). Parámetros: $g = 0.2$ y $M = A$	76
5.15. Valores de ν obtenidos por el Algoritmo 8 para $p = 8$ (izquierda) y $p = 10$ (derecha). Parámetros: $g = 0.2$ y $M = A$	77
5.16. Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo del diámetro de la tubería (derecha). Parámetros: $p = 4$, $g = 0.2$, $M = I$ y $\delta_0 = 0.25$	78
5.17. Valores de ν calculados por el Algoritmo 8 para cada δ_0 . Parámetros: $p = 4$, $g = 0.2$ y $M = I$	80

Índice de cuadros

5.1. Comportamiento general del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = A$	63
5.2. Comportamiento general del Algoritmo 8. Parámetros: $p = 1.5$ y $M = A$	66
5.3. Comportamiento general del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$	67
5.4. Comportamiento del Algoritmo 8 ante distintos δ_0 . Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$	69
5.5. Comportamiento general del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$, $M = I$ y $\delta_0 = 0.75$	70
5.6. Comportamiento general del Algoritmo 8. Parámetros: $p = 4$ y $M = A$	72
5.7. Comportamiento general del Algoritmo 8. Parámetros: $p = 4$, $g = 0.2$ y $M = I$	74
5.8. Comportamiento general del Algoritmo 8. Parámetros: $g = 0.2$ y $M = A$	77
5.9. Comportamiento general del Algoritmo 8 ante distintos δ_0 . Parámetros: $p = 4$, $g = 0.2$ y $M = I$	79
5.10. Comportamiento general del Algoritmo 8. Parámetros: $p = 4$, $g = 0.2$, $M = I$ y $\delta_0 = 1$	79

Índice de algoritmos

1.	Método Quasi-Newton	11
2.	Algoritmo BFGS	12
3.	Método Bundle	19
4.	Esquema de Algoritmo tipo Quasi-Newton	23
5.	Subalgoritmo Bundle	27
6.	Algoritmo tipo Bundle Quasi-Newton	28
7.	Algoritmo de búsqueda lineal backtracking	50
8.	Algoritmo tipo Bundle no diferenciable aplicado al problema de flujo . .	51

Resumen

En el presente trabajo, se desarrolla un método de optimización no suave aplicado al modelo de flujo de Herschel-Bulkley. El movimiento de estos fluidos se caracteriza como un problema de minimización de un funcional convexo y no diferenciable. Para la resolución de este, seguimos el enfoque “discretizar, luego optimizar”. Usamos el método de elementos finitos para obtener una aproximación del problema de interés, luego, por medio de la regularización de Moreau-Yosida, obtenemos un problema de optimización diferenciable que es equivalente a minimizar el funcional no suave. Aplicar este tipo de regularización es una tarea desafiante, pues, es necesario resolver un problema de optimización auxiliar en cada iteración. Siguiendo las ideas de los métodos Bundle, desarrollamos una estrategia que permite resolver aproximadamente este problema auxiliar. Construimos un método que sigue principalmente el esquema de la actualización BFGS, y que además, resuelve el problema auxiliar en cada iteración como una subrutina. Se demuestra que el algoritmo propuesto converge globalmente hacia el mínimo, y en adición, posee un orden de convergencia superlineal para el caso de fluidos espesantes. Finalmente, se presenta una serie de experimentos numéricos que permiten corroborar la eficiencia del algoritmo y sus propiedades de convergencia.

Palabras claves: Optimización no suave · Modelo de Herschel-Bulkley · Regularización de Moreau-Yosida · Métodos Bundle · Actualización BFGS

Abstract

In the present work, a nonsmooth optimization method applied to the Herschel-Bulkley flow model is developed. The motion of these fluids is characterized as a minimization problem of a convex and nonsmooth functional. To solve it, we follow the “discretize, then optimize” approach. We use the finite element method to obtain an approximation of the problem of interest, then, by means of the Moreau-Yosida regularization, we obtain a smooth optimization problem that is equivalent to minimizing the nonsmooth functional. Applying this type of regularization is a challenging task, since, it is necessary to solve an auxiliary optimization problem at each iteration. Following the ideas of Bundle methods, we develop a strategy that allows to solve this auxiliary problem, approximately. We build a method that mainly follows the BFGS update scheme, and furthermore, solves the auxiliary problem at each iteration as a subroutine. It is shown that the proposed algorithm converges globally to the minimum, in addition, it possesses a superlinear convergence rate for the case of shear-thickening fluids. Finally, several numerical experiments are presented to verify the efficiency of the algorithm and its convergence properties.

Keywords: Nonsmooth optimization · Herschel-Bulkley model · Moreau-Yosida regularization · Bundle methods · BFGS update

Capítulo 1

Introducción

El modelamiento matemático del flujo de materiales viscoplásticos como son los fluidos de Herschel-Bulkley, se centra en la resolución del problema de optimización no diferenciable

$$\min_{u \in W_0^{1,p}(\Omega)} J(u) := \frac{1}{p} \int_{\Omega} |\nabla u|^p dx + g \int_{\Omega} |\nabla u| dx - \int_{\Omega} f u dx, \quad (1.1)$$

con $1 < p < \infty$, $g > 0$ y $f \in L^{p'}(\Omega)$.

Estos fluidos se pueden encontrar en la naturaleza, por ejemplo, en las arenas movedizas o la sangre; o son el resultado de una mezcla de ciertos materiales destinados a crear un elemento o producto, tales como la salsa de tomate o las suspensiones industriales de arcillas. Están muy presentes en la industrias alimentaria, química y petrolera. Por tanto, comprender su comportamiento para un uso y tratamiento adecuado permitirá la potencialización de ciertos productos que involucran a este tipo de fluidos.

Los métodos que se han desarrollado para abordar el problema (1.1) han sido mediante el esquema de los lagrangianos aumentados propuesto por Huilgol y You en [15] y de un algoritmo de descenso preconditionado como el de González-Andrade en [13].

Ahora bien, al tratarse de un problema no diferenciable se desea explorar la teoría y métodos que se han construido para problemas de este tipo y proponer un nuevo algoritmo eficiente y robusto, basado en la optimización no suave, para la resolución de (1.1).

Para aplicar un método de optimización no diferenciable seguiremos el enfoque “discretizar, luego optimizar”. Para ello, se estudia y se adapta el esquema del método no suave propuesto por Mifflin, Sun y Qi en [22] para la resolución numérica del problema

de estudio.

Este trabajo se presenta en cinco capítulos. En el primer capítulo se presenta una breve introducción y los antecedentes del problema. A continuación, el segundo capítulo revisa conceptos y resultados que serán de utilidad para mostrar ciertas propiedades del funcional en (1.1). Se describe el modelo de Hershel-Bulkley y su problema de optimización asociado. Además, se revisan métodos de optimización suave, y finalmente, se aborda una breve introducción a la optimización no suave.

En el tercer capítulo se examinan los métodos Bundle para la resolución de problemas no diferenciables. Particularmente, se describe el algoritmo Bundle Quasi-Newton desarrollado en [16, 21, 22], se estudia la derivación del mismo así como las propiedades de convergencia que posee. Adicionalmente, se introduce la regularización de Moreau-Yosida para una función convexa.

En el capítulo 4 se estudia de manera analítica el problema (1.1), se muestra la existencia y unicidad de la solución. Posteriormente, se caracteriza el subdiferencial de J para luego construir un algoritmo no suave aplicado al problema de flujo, el cual está basado en el método Bundle y la regularización de Moreau-Yosida. Se analizan las propiedades de convergencia hacia el mínimo que este posee.

En el capítulo 5 se describen varios aspectos para la implementación del método en un software numérico. Se reportan los resultados de experimentos que simulan el comportamiento del flujo de un material de Herschel-Bulkley bajo diferentes condiciones como regiones, variación del índice de flujo y del umbral de plasticidad. Por último, se informa el desenvolvimiento del algoritmo propuesto y se corroboran las propiedades que este posee. Finalmente, se presentan las conclusiones y recomendaciones de la investigación.

Capítulo 2

Preliminares

2.1. Conceptos y Resultados Generales

En esta sección revisaremos conceptos y resultados principales que serán empleados para establecer las propiedades que goza el funcional objetivo del problema de optimización de nuestro interés. Para mayor detalle de estos conceptos y la demostración de los resultados referimos al lector [2, cap. 2] y [5, cap. 9].

2.1.1. Notaciones preliminares

El símbolo Ω denotará siempre un dominio abierto de \mathbb{R}^n . El espacio \mathbb{R}^n está dotado de la medida de Lebesgue dx , y todas las integrales se considerarán en el sentido de Lebesgue. La medida de un conjunto $E \subset \mathbb{R}^n$ se denotará por $|E|$. Diremos que una propiedad se tiene en casi todo punto (c.t.p.) en Ω si existe un subconjunto $E \subset \Omega$ de medida nula tal que la propiedad se cumple en cada punto de $\Omega \setminus E$.

Si $u : \Omega \rightarrow \mathbb{R}$ es diferenciable, el gradiente de u en $x = (x_1, \dots, x_n)$ es el vector $\nabla u(x) = \left(\frac{\partial u}{\partial x_1}(x), \dots, \frac{\partial u}{\partial x_n}(x) \right)$.

2.1.2. Espacios de funciones

Se introducen los siguientes espacios de funciones:

- $C^k(\Omega)$, con $k = 1, 2, \dots$ es el espacio de funciones $u : \Omega \rightarrow \mathbb{R}$ que son k veces diferenciables en Ω y cuya k -ésima derivada es continua en Ω .
- $C^\infty(\Omega)$ es el espacio de funciones $u : \Omega \rightarrow \mathbb{R}$ que son infinitamente diferenciables

en Ω .

- $C_c^\infty(\Omega)$ es el subespacio de $C^\infty(\Omega)$ que consiste de aquellas funciones con soporte compacto en Ω . El soporte de una función (continua) $u : \Omega \rightarrow \mathbb{R}$, denotado por $\text{supp } u$ es la clausura del conjunto $\{x \in \Omega : u(x) \neq 0\}$.
- $L^p(\Omega)$, para $p \in [1, +\infty)$, es el espacio de Lebesgue de funciones medibles $u : \Omega \rightarrow \mathbb{R}$ tales que $\int_\Omega |u(x)|^p dx < +\infty$. $L^p(\Omega)$ es un espacio de Banach dotado de la norma

$$\|u\|_{L^p} = \left(\int_\Omega |u|^p dx \right)^{1/p}.$$

- $W^{1,p}(\Omega)$, con $p \in [1, +\infty)$, es el espacio de Sobolev definido por

$$W^{1,p}(\Omega) = \left\{ u \in L^p(\Omega) \left| \begin{array}{l} \exists g_1, g_2, \dots, g_n \in L^p(\Omega) \text{ tal que} \\ \int_\Omega u \frac{\partial \varphi}{\partial x_i} = - \int_\Omega g_i \varphi \quad \forall \varphi \in C_c^\infty(\Omega), \quad \forall i = 1, 2, \dots, n \end{array} \right. \right\}.$$

Para $u \in W^{1,p}(\Omega)$ se define $\frac{\partial u}{\partial x_i} := g_i$ (ver [5, pág. 263]). $W^{1,p}(\Omega)$ es un espacio de Banach dotado de la siguiente norma

$$\|u\|_{W^{1,p}} = \left(\|u\|_{L^p} + \sum_{i=1}^n \left\| \frac{\partial u}{\partial x_i} \right\|_{L^p}^p \right)^{1/p}.$$

- $W_0^{1,p}(\Omega)$ es la clausura de $C_c^\infty(\Omega)$ en la norma de $W^{1,p}(\Omega)$. $W_0^{1,p}(\Omega)$ es un espacio de Banach dotado de la norma de $W^{1,p}(\Omega)$.

A continuación, estudiamos algunos resultados clásicos para este tipo de espacios funcionales.

Teorema 2.1.1. *Sea $\Omega \subset \mathbb{R}^n$ un conjunto abierto y $\{u_k\}$ una sucesión en $L^p(\Omega)$, con $p \in [0, +\infty]$, tal que $u_k \rightarrow u$ en $L^p(\Omega)$ cuando $k \rightarrow \infty$. Entonces existe una subsucesión $\{u_{k_j}\}$ y una función $v \in L^p(\Omega)$ tales que*

(I) $u_{k_j}(x) \rightarrow u(x)$ c.t.p. en Ω cuando $j \rightarrow \infty$;

(II) para todo j , $|u_{k_j}(x)| \leq v(x)$ c.t.p en Ω .

Teorema 2.1.2 (Inmersiones de Sobolev). *Sea $\Omega \subset \mathbb{R}^n$ un conjunto abierto, acotado y con frontera suficientemente suave. Sea $p \geq 1$.*

- Si $p < n$, entonces $W^{1,p}(\Omega) \hookrightarrow L^q(\Omega)$ para cada $q \in [1, \frac{np}{n-p}]$, la inyección es compacta para $q \in [1, \frac{np}{n-p})$.

- Si $p = n$, entonces $W^{1,p}(\Omega) \hookrightarrow L^q(\Omega)$ para cada $q \in [1, +\infty]$, la inyección es compacta.
- Si $p > n$, entonces $W^{1,p}(\Omega) \hookrightarrow L^\infty(\Omega)$, la inyección es compacta.

Teorema 2.1.3 (Desigualdad de Poincaré para $W_0^{1,p}(\Omega)$). Sea $\Omega \subset \mathbb{R}^n$ un conjunto abierto y acotado. Entonces existe una constante $C > 0$, que depende únicamente de Ω , tal que

$$\int_{\Omega} |u|^p dx \leq C \int_{\Omega} |\nabla u|^p dx, \quad \forall u \in W_0^{1,p}(\Omega).$$

Observación 2.1.1. Gracias a este teorema, si Ω es un conjunto abierto y acotado, entonces

$$\left(\int_{\Omega} |\nabla u|^p dx \right)^{\frac{1}{p}}$$

es una norma en $W_0^{1,p}(\Omega)$, equivalente a la norma usual.

2.1.3. Funcionales Convexos

Definición 2.1.1. Un funcional $I : X \rightarrow \mathbb{R}$ sobre un espacio vectorial X se dice convexo si para cada $u, v \in X$ y cada real $\lambda \in [0, 1]$ se tiene

$$I(\lambda u + (1 - \lambda)v) \leq \lambda I(u) + (1 - \lambda)I(v).$$

El funcional I se dice estrictamente convexo si para cada $u, v \in X$, $u \neq v$, y cada real $\lambda \in [0, 1]$ se tiene

$$I(\lambda u + (1 - \lambda)v) < \lambda I(u) + (1 - \lambda)I(v).$$

Teorema 2.1.4. Sea $I : X \rightarrow \mathbb{R}$ un funcional continuo y convexo sobre un espacio de Banach X . Entonces I es débilmente semicontinuo inferior. En particular, para cada sucesión $\{u_k\}$ en X que converge débilmente a $u \in X$, se cumple que

$$I(u) \leq \liminf_{k \rightarrow \infty} I(u_k).$$

Definición 2.1.2. Un funcional $I : X \rightarrow \mathbb{R}$ sobre un espacio de Banach X se dice coercivo si, para cada sucesión $\{u_k\}$ en X ,

$$\|u_k\|_X \rightarrow +\infty \quad \text{implica} \quad I(u_k) \rightarrow +\infty.$$

Teorema 2.1.5. Sea X un espacio de Banach reflexivo y sea $I : X \rightarrow \mathbb{R}$ un funcional continuo, convexo y coercivo. Entonces I tiene un mínimo global. Si en adición, I es estrictamente convexo entonces el mínimo es único.

2.2. Flujo viscoplástico de Herschel-Bulkley y su problema de optimización asociado

Los fluidos de Herschel-Bulkley son fluidos no newtonianos, estos no obedecen la ley de acción y reacción. En su lugar, estos fluidos siguen un modelo matemático de ley de potencias (*power-law*) cuya formulación viene dada por el operador p -laplaciano y la norma L^1 del gradiente. Este modelo es usado para simular el flujo de ciertos materiales.

Estos fluidos están caracterizados por dos parámetros: umbral de plasticidad e índice de flujo. Este último mide el grado en que el fluido se adelgaza (*shear-thinning behaviour*) o se espesa (*shear-thickening behaviour*). El umbral de plasticidad, por otro lado, cuantifica la tensión a la que el material puede ser sometido para comportarse como un sólido rígido o un fluido. Cuando una fuerza o tensión está por debajo de este límite, el material toma el comportamiento de sólido rígido y si esta sobrepasa el umbral de plasticidad entonces se comporta como un fluido.

Para el caso de los fluidos adelgazantes, entre mayor sea la tensión, el material se comporta como un fluido, como por ejemplo la salsa de tomate, pinturas y barnices. En cuanto a los fluidos espesantes, entre mayor sea la tensión estos se comportan como un sólido rígido, por ejemplo, *Oobleck* (mezcla de agua con maicena).

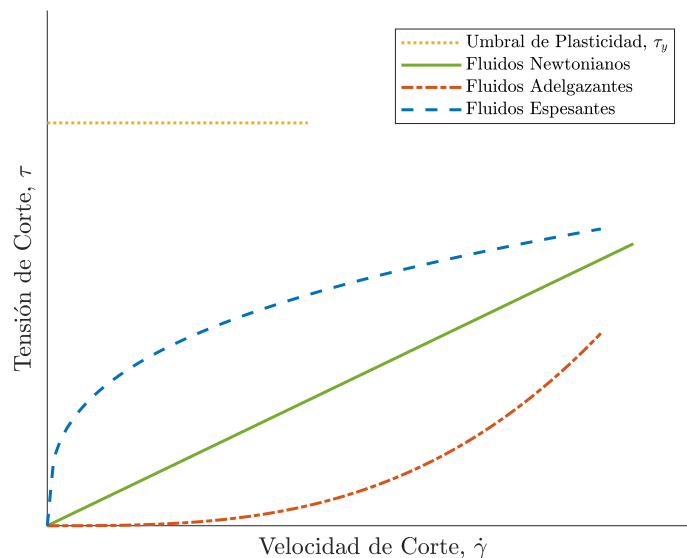


Figura 2.1: Bosquejo del comportamiento de distintos tipos de fluidos

La gráfica anterior ilustra la clasificación de los fluidos con la tensión de corte como

función de la velocidad de corte. Se puede observar que los fluidos newtonianos siguen un modelo lineal, mientras que los fluidos adelgazantes y espesantes se comportan acorde a un modelo de ley de potencias.

A continuación describimos en detalle el modelo para este tipo de materiales desarrollado por Huilgol y You en [15].

2.2.1. Modelo constitutivo de Herschel-Bulkley

Sea τ_y el parámetro que cuantifica el umbral de plasticidad. Cuando un fluido de Herschel-Bulkley ha cedido, se tiene que la tensión de corte τ verifica la siguiente relación

$$\tau = \tau_y + k_H |\dot{\gamma}|^{n-1} \dot{\gamma},$$

donde k_H es el parámetro de viscosidad, $\dot{\gamma}$ es la velocidad de corte y $n > 0$.

Supongamos que el fluido se encuentra estacionario en una tubería cuya longitud característica es L , que puede ser el radio de una sección transversal circular o la mitad de la longitud de la arista de una sección transversal cuadrada. Tomando una presión constante y positiva $c = -dp/dz$, y la velocidad característica $L (cL/k_H)^{1/n}$, el modelo matemático de Herschel-Bulkley está descrito por (ver [15, sec. 2])

$$\begin{cases} \tau_{ij} = \frac{Od}{\dot{\gamma}} \dot{\gamma}_{ij} + \dot{\gamma}^{n-1} \dot{\gamma}_{ij}, & \text{si } |\tau_{ij}| > Od, \\ \dot{\gamma}_{ij} = 0, & \text{si } |\tau_{ij}| > Od, \end{cases} \quad (2.1)$$

donde τ_{ij} es el tensor de fuerza, $\dot{\gamma}_{ij}$ un tensor que mide la tasa de deformación, $n > 0$ el índice de flujo y Od el número de Oldroyd que está dado por

$$Od = \frac{\tau_y}{cL},$$

el cual representa una proporción entre el umbral de plasticidad y la tensión viscosa. Para $n < 1$, el modelo describe el comportamiento de fluidos adelgazantes. Por otro lado, para $n > 1$ se modela el comportamiento de los fluidos espesantes.

En la figura 2.2 se observa el modelo de Herschel-Bulkley en relación de la velocidad de corte con la tensión de corte. Se ilustra el caso de fluidos adelgazantes ($n < 1$) como espesantes ($n > 1$). Además, se presenta el modelo de Bingham que puede ser visto como caso particular del modelo de Herschel-Bulkley cuando $n = 1$ (ver [13, 15]).

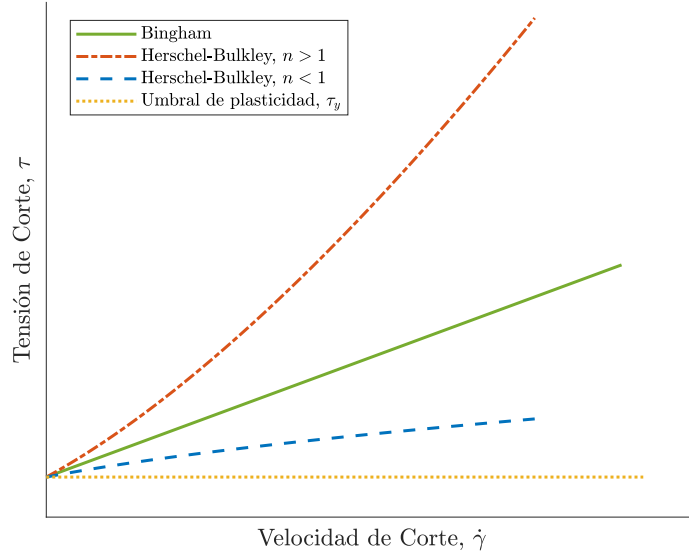


Figura 2.2: Bosquejo del modelo de Herschel-Bulkley

2.2.2. Formulación como un problema de optimización

En esta sección presentamos el modelo de flujo de Herschel-Bulkley como un problema de optimización en espacios funcionales. Esto nos permitirá resolver este problema aplicando un método de optimización adecuado.

Supongamos nuevamente que el fluido se encuentra en estado estacionario dentro de una tubería. Sea $\Omega \subset \mathbb{R}^2$ un dominio abierto, acotado y suficientemente regular, que representa la sección transversal de la tubería. La tensión de corte está dada por

$$\tau = \dot{\gamma}^{p-2} \dot{\gamma} + Od,$$

con $p > 1$. Suponga además que la velocidad del fluido se desvanece en la frontera y en adición posee una componente no nula en la dirección axial únicamente. Cuando el fluido ha cedido, su velocidad u satisface la siguiente igualdad

$$-\nabla \cdot \left[\dot{\gamma}^{p-2} \nabla u + \frac{\tau_y}{\dot{\gamma}} \nabla u \right] = f, \quad (2.2)$$

donde $f \in L^{p'}(\Omega)$, es un parámetro que mide el decaimiento de la presión en la tubería, y la tasa de corte $\dot{\gamma} = |\nabla u| \geq 0$.

Reemplazando τ_y , $\dot{\gamma}$ en (2.2) y añadiendo las condiciones de borde se tiene el si-

guiente problema con condiciones en la frontera

$$\begin{cases} -\nabla \cdot \left[|\nabla u|^{p-2} \nabla u + \frac{Od}{|\nabla u|} \nabla u \right] = f, & \text{en } \Omega \\ u = 0, & \text{sobre } \partial\Omega. \end{cases} \quad (2.3)$$

Siguiendo las ideas propuestas en [15], $u \in W_0^{1,p}(\Omega)$ es solución de (2.3) si y sólo si satisface la siguiente desigualdad variacional

$$\begin{aligned} \int_{\Omega} |\nabla u|^{p-2} (\nabla u, \nabla v) \, dx + Od \int_{\Omega} |\nabla v| \, dx - Od \int_{\Omega} |\nabla u| \, dx \\ \geq \int_{\Omega} f(v - u) \, dx, \quad \forall v \in W_0^{1,p}(\Omega). \end{aligned} \quad (2.4)$$

González-Andrade menciona en [13] que esta desigualdad variacional representa una condición de optimalidad necesaria para el siguiente problema de optimización

$$\min_{u \in W_0^{1,p}(\Omega)} J(u) := \frac{1}{p} \int_{\Omega} |\nabla u|^p \, dx + Od \int_{\Omega} |\nabla u| \, dx - \int_{\Omega} f u \, dx. \quad (2.5)$$

Por consiguiente, resolver (2.5) es equivalente a hallar la velocidad de un fluido de Herschel-Bulkley estacionario en una tubería.

Al problema (2.5) lo denominaremos el problema de flujo. En el capítulo 4, se analizan las propiedades del funcional J , y en adición estudiaremos la existencia y unicidad de este problema.

2.3. Métodos Quasi-Newton

A mediados de los años 50, el físico norteamericano W.C. Davidon desarrolló el primer algoritmo Quasi-Newton que resultó una de las ideas más creativas en la optimización no lineal. Posteriormente, Fletcher y Powell demostraron que este nuevo algoritmo era mucho más rápido y confiable que otros existentes. En los siguientes veinte años fueron propuestas muchas variaciones de este método y numerosos artículos se dedicaron al estudio de las mismas.

Los métodos Quasi-Newton requieren únicamente que el gradiente de la función objetivo esté disponible en cada iteración. Estos métodos generan una serie de aproximaciones de la Hessiana de la función, y al mismo tiempo poseen una tasa de convergencia rápida. Para profundizar en la temática se refiere al lector los textos [24, cap. 6] y [27, cap. 5].

Ahora, describimos la construcción del método Quasi-Newton. Sea $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una función dos veces continuamente diferenciable en una vecindad $V \subset \mathbb{R}^n$. Gracias al teorema de Taylor se tiene la siguiente expansión cuadrática de f en x_{k+1} ,

$$f(x) \approx f(x_{k+1}) + \nabla f(x_{k+1})^T (x - x_{k+1}) + \frac{1}{2} (x - x_{k+1})^T \nabla^2 f(x_{k+1}) (x - x_{k+1}).$$

Si derivamos esta última igualdad se tiene que

$$\nabla f(x) \approx \nabla f(x_{k+1}) + \nabla^2 f(x_{k+1}) (x - x_{k+1}).$$

Considerando esta aproximación en x_k y fijando $\delta_k = x_{k+1} - x_k$ y $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$, tenemos que

$$(\nabla^2 f(x_{k+1}))^{-1} y_k \approx \delta_k.$$

Esta igualdad se verifica exactamente para funciones cuadráticas.

Por tanto, exigimos a nuestras matrices H_k que satisfagan esta condición, es decir,

$$H_{k+1} y_k = \delta_k. \quad (2.6)$$

A esta condición se la conoce como ecuación Quasi-Newton o condición Quasi-Newton. Nótese que, entonces, vamos a construir una sucesión de matrices $\{H_k\}$ que aproximan $(\nabla^2 f(x_k))^{-1}$ en cada x_k .

Por otro lado, supongamos que tenemos el siguiente modelo para f en x_{k+1}

$$m_{k+1}(x) = f(x_{k+1}) + \nabla f(x_{k+1})^T (x - x_{k+1}) + \frac{1}{2} (x - x_{k+1})^T B_{k+1} (x - x_{k+1}),$$

el cual satisface las siguientes condiciones de interpolación

$$m_{k+1}(x_{k+1}) = f(x_{k+1}) \quad \text{y} \quad \nabla m_{k+1}(x_{k+1}) = \nabla f(x_{k+1}),$$

y además, suponga que $B_{k+1} = H_{k+1}^{-1}$ es una aproximación de $\nabla^2 f(x_{k+1})$. Si exigimos que el modelo cumpla

$$\nabla m_{k+1}(x_k) = \nabla f(x_k),$$

se tiene que

$$\nabla f(x_k) = \nabla f(x_{k+1}) + B_{k+1} (x_k - x_{k+1}),$$

de donde, se sigue que

$$B_{k+1} \delta_k = y_k. \quad (2.7)$$

Esta última condición es llamada condición de Quasi-Newton en la Hessiana. Tanto (2.6) como (2.7) se conocen como condiciones de la secante.

Luego, multiplicando (2.7) por δ_k^T se sigue que

$$\delta_k^T B_{k+1} \delta_k = \delta_k^T y_k,$$

lo que implica que B_{k+1} es definida positiva si

$$\delta_k^T y_k > 0, \quad \forall k.$$

A esta expresión se la conoce generalmente como condición de curvatura.

La discusión anterior nos dice que la clave de los métodos Quasi-Newton es determinar H_{k+1} o B_{k+1} usando ciertos métodos tales que (2.6) o (2.7) se verifiquen. Ahora, describimos el algoritmo Quai-Newton general.

Algoritmo 1: Método Quasi-Newton

Paso 1. Fijar $x_0 \in \mathbb{R}^n$, $H_0 \in \mathbb{R}^{n \times n}$ o $B_0 \in \mathbb{R}^{n \times n}$, $0 \leq \varepsilon \lll 1$ y $k = 0$.

Paso 2. Si $\|\nabla f(x_k)\| \leq \varepsilon$, **PARAR**

Paso 3. Calcular $\delta_k = -H_k \nabla f(x_k)$ o $B_{k+1} \delta_k = -\nabla f(x_k)$.

Paso 4. Usar búsqueda lineal para calcular el paso λ_k .

Paso 5. Actualizar $x_{k+1} = x_k + \lambda_k \delta_k$, $H_k \rightarrow H_{k+1}$ tal que (2.6) se cumpla o actualizar $B_k \rightarrow B_{k+1}$ tal que (2.7) se cumpla.

Paso 6. $k = k + 1$ e ir al paso 2.

2.3.1. Actualización BFGS

El método Quasi-Newton más popular es la actualización BFGS, nombrada así por sus descubridores Broyden, Fletcher, Goldfarb, y Shanno. Este método es una actualización de rango 2 que se obtiene usando la ecuación (2.7)

$$B_{k+1} \delta_k = y_k.$$

Así, BFGS genera una sucesión de matrices $\{B_k\}$ que aproximan $\nabla^2 f(x_k)$.

Planteamos la siguiente actualización

$$B_{k+1} = B_k + a u u^T + b v v^T,$$

con $a, b \in \mathbb{R}$ y $u, v \in \mathbb{R}^n$ a ser determinados. En este caso, esperamos que

$$B_k \delta_k + a u u^T \delta_k + b v v^T \delta_k = y_k.$$

Es conocido que u y v no son únicos, en el sentido que existen varias opciones de estos

vectores para que se cumpla esta ecuación. Así, se puede tomar

$$u = y_k, \quad v = B_k \delta_k, \quad a = \frac{1}{y_k^T \delta_k}, \quad \text{y} \quad b = \frac{1}{\delta_k^T B_k \delta_k}.$$

Con esta selección se tiene que

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T \delta_k} + \frac{B_k \delta_k \delta_k^T B_k}{\delta_k^T B_k \delta_k}. \quad (2.8)$$

Entonces, exponemos el siguiente algoritmo

Algoritmo 2: Algoritmo BFGS

Paso 1. Fijar $x_0 \in \mathbb{R}^n$, $B_0 \in \mathbb{R}^{n \times n}$ simétrica definida positiva, $0 \leq \varepsilon \lll 1$ y $k = 0$.

Paso 2. Si $\|\nabla f(x_k)\| \leq \varepsilon$, **PARAR**

Paso 3. Calcular $B_k \rho_k = -\nabla f(x_k)$.

Paso 4. Usar búsqueda lineal para calcular el paso λ_k .

Paso 5. Actualizar $x_{k+1} = x_k + \lambda_k \rho_k$, $\delta_k = x_{k+1} - x_k$,

$y_{k+1} = \nabla f(x_{k+1}) - \nabla f(x_k)$ y B_{k+1} usando (2.8).

Paso 6. $k = k + 1$ e ir al paso 2.

La actualización BFGS preserva la propiedad de definida positiva para todo B_k si y sólo si $\delta_k^T y_k > 0$. En adición, este algoritmo converge global y superlinealmente hacia el mínimo (ver [24, pág. 131]).

2.4. Introducción a la Optimización No Diferenciable

En esta sección revisamos brevemente la teoría de optimización no suave. Se introducirán algunos conceptos y se describirá una condición de optimalidad para este tipo de problemas. Para estudiar más a fondo este tema referimos al lector el libro *Introduction to Nonsmooth Optimization* escrito por A. Bagirov, N. Karmitza y M. Mäkelä.

Considere el siguiente problema de optimización

$$\min_{x \in \mathbb{R}^n} f(x), \quad (2.9)$$

donde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ es una función convexa no suave. A (2.9) se lo conoce como problema convexo.

Definición 2.4.1. Se dice que \bar{x} es un mínimo global de (2.9) si verifica que

$$f(\bar{x}) \leq f(x), \quad \forall x \in \mathbb{R}^n.$$

Es conocido que una condición de optimalidad para hallar un punto mínimo de una función diferenciable es $\nabla f(x) = 0$. Puesto que f es una función no suave, no es posible asegurar la existencia de ∇f en cada punto, por lo que, surge la necesidad de definir una estructura análoga al gradiente de una función para obtener una condición similar para el problema convexo.

Definición 2.4.2 (Subdiferencial). *El subdiferencial de una función convexa $f : \mathbb{R}^n \rightarrow \mathbb{R}$ en el punto $x \in \mathbb{R}^n$ se define como*

$$\partial f(x) := \{ \xi \in \mathbb{R}^n : f(y) - f(x) \geq \xi^T(y - x) \quad \forall y \in \mathbb{R}^n \}.$$

Al vector $\xi \in \mathbb{R}^n$ que pertenece a $\partial f(x)$ se lo conoce como subgradiente de f en x .

Observación 2.4.1. Si f es diferenciable se tiene que $\partial f(x) = \{ \nabla f(x) \}$ para todo $x \in \mathbb{R}^n$.

El siguiente resultado establece la existencia de la derivada direccional de una función convexa.

Teorema 2.4.1. *Sea $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una función convexa. Entonces la derivada direccional $f'(x; d)$ existe en cualquier dirección $d \in \mathbb{R}^n$, y además verifica que*

$$f'(x; d) = \liminf_{t > 0} \frac{f(x + td) - f(x)}{t}.$$

Demostración. Sea $d \in \mathbb{R}^n$ una dirección arbitraria. Definimos $\varphi : \mathbb{R} \rightarrow \mathbb{R}$ como

$$\varphi(t) := \frac{f(x + td) - f(x)}{t}.$$

Empezaremos mostrando que φ es acotada inferiormente cuando $t \downarrow 0$. Sea $\varepsilon > 0$ y t_1, t_2 constantes tales que $0 < t_1 < t_2 < \varepsilon$. Puesto que f es convexa se tiene que

$$\begin{aligned} \varphi(t_2) - \varphi(t_1) &= \frac{1}{t_1 t_2} [t_1 f(x + t_2 d) - t_2 f(x + t_1 d) + (t_2 - t_1) f(x)] \\ &= \frac{1}{t_1} \left\{ \left(\frac{t_1}{t_2} f(x + t_2 d) + \left(1 - \frac{t_1}{t_2} \right) f(x) \right) \right. \\ &\quad \left. - f \left(\frac{t_1}{t_2} (x + t_2 d) + \left(1 - \frac{t_1}{t_2} \right) x \right) \right\} \\ &\geq 0, \end{aligned}$$

así, $\varphi(t)$ es decreciente cuando $t \downarrow 0$. Por consiguiente, para todo $0 < t < \varepsilon$ se sigue

que

$$\begin{aligned}
\varphi(t) - \varphi(-\varepsilon/2) &= \frac{\frac{1}{2}f(x+td) + \frac{1}{2}f(x) + \frac{t}{\varepsilon}f\left(x - \frac{\varepsilon}{2}d\right) + \left(1 - \frac{t}{\varepsilon}\right)f(x) - 2f(x)}{t/2} \\
&\geq \frac{\frac{1}{2}f\left(x + \frac{t}{2}d\right) + \frac{1}{2}f\left(x - \frac{t}{2}d\right) - f(x)}{t/4} \\
&\geq \frac{f(x) - f(x)}{t/4} = 0,
\end{aligned}$$

lo que implica que la función φ es acotada inferiormente para $0 < t < \varepsilon$. Así, deducimos que el límite de φ cuando $t \downarrow 0$ existe. Y además se cumple que:

$$\lim_{t \downarrow 0} \varphi(t) = f'(x; d), \quad \forall d \in \mathbb{R}^n.$$

En adición, como $\varphi(t)$ decrece cuando $t \downarrow 0$ se concluye que

$$f'(x; d) = \inf_{t > 0} \frac{f(x+td) - f(x)}{t}.$$

□

A continuación se introduce el concepto de punto estacionario para el problema (2.9).

Definición 2.4.3. *Sea $x \in \mathbb{R}^n$ tal que $0 \in \partial f(x)$, entonces x se dice punto estacionario de f .*

El próximo teorema brinda una condición de optimalidad necesaria y suficiente para el problema convexo.

Teorema 2.4.2. *Si $f : \mathbb{R}^n \rightarrow \mathbb{R}$ es una función convexa, entonces esta alcanza un mínimo global en \bar{x} si y sólo si*

$$0 \in \partial f(\bar{x}) \quad \text{o} \quad f'(\bar{x}; d) \geq 0, \quad \forall d \in \mathbb{R}^n.$$

La demostración de este resultado puede ser estudiado en [3, cap. 4].

Capítulo 3

Métodos Bundle

En el año 1975 surgieron los primeros métodos Bundle como algoritmos para la resolución numérica de problemas de optimización no diferenciable. Los pioneros en proponer estos procedimientos son los matemáticos Philip Wolfe y Claude Lemaréchal. Durante las décadas de los ochenta y noventa, el interés sobre el estudio y la aplicación de estos métodos fue creciendo, lo que representó un gran avance para la optimización no suave.

En este capítulo, se describe el método general Bundle y el algoritmo tipo Bundle Quasi-Newton, estos dos constituyen una base fundamental y muy útil para el desarrollo del algoritmo propuesto en el presente trabajo.

3.1. Método General Bundle

En la presente sección estudiamos el método general Bundle, haciendo énfasis en la construcción, el funcionamiento y las propiedades de convergencia que presenta el modelo. Con referencia a lo mencionado, basándonos en los trabajos de Bagirov et al. [3], Bonnans et al. [4], Hiriart-Urruty y Lemaréchal [14] y Mäkelä [19]; se describe detalladamente el algoritmo Bundle.

Considere el siguiente problema de minimización

$$\min_{x \in \mathbb{R}^n} f(x), \tag{3.1}$$

donde la función objetivo $f : \mathbb{R}^n \rightarrow \mathbb{R}$ es convexa, es decir,

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y),$$

para cualesquier $x, y \in \mathbb{R}^n$ y $\lambda \in [0, 1]$. En adición, suponga que para todo $x \in \mathbb{R}^n$ es posible calcular el valor de la función $f(x)$ y al menos un subgradiente $\xi(x)$ perteneciente al subdiferencial de f en x , que se define por

$$\partial f(x) = \{ \xi(x) \in \mathbb{R}^n : f(y) - f(x) \geq \xi(x)^T (y - x), \text{ para todo } y \in \mathbb{R}^n \}.$$

Para continuar con la descripción del método Bundle es necesario introducir las siguientes definiciones.

Definición 3.1.1. Sea $x \in \mathbb{R}^n$ arbitrario, al algoritmo que devuelve $(f(x), \xi(x))$ (con $\xi(x) \in \partial f(x)$), se lo denomina oráculo.

Definición 3.1.2. En la k -ésima iteración de un método de minimización, sea

$$\{y^j \in \mathbb{R}^n : j = 1, \dots, k\},$$

el conjunto de puntos que han llamado al oráculo desde el inicio del método. Se define el bundle como el conjunto de tripletas

$$\mathcal{B} := \{(y^j, f(y^j), \xi(y^j) \in \partial f(y^j)) : j = 1, \dots, k\} \quad (3.2)$$

que almacena toda la información disponible desde el inicio del método.

El método Bundle almacena en un arreglo (*bundle*) toda la información posible de iteraciones pasadas que han sido generadas por el algoritmo de minimización y, de esta manera, construir un modelo (\check{f}_k) que aproxime a f . Con esta información extra siendo almacenada a lo largo de las iteraciones, el algoritmo es capaz de generar dos sucesiones: La primera sucesión está constituida por los puntos que son usados para modelar \check{f}_k , y se los denomina *puntos candidatos* (*trial points*), denotados por y^k . La segunda sucesión consiste de aquellos puntos que han generado un decrecimiento suficiente de la función objetivo f en el sentido de (3.7), y son llamados *centros de estabilidad* (*stability centers*), denotados por x^k .

Observación 3.1.1. Nótese que $\{x^k\}$ es una subsucesión de $\{y^k\}$.

Suponemos que a lo largo de todas las iteraciones, toda la información disponible es recolectada y almacenada en un arreglo. Es decir, en la k -ésima iteración del algoritmo, se dispone del centro de estabilidad x^k y en adición puntos candidatos $y^j \in \mathbb{R}^n$ (de iteraciones pasadas) y subgradientes $\xi^j := \xi(y^j) \in \partial f(y^j)$, con $j = 1, \dots, k$.

El método Bundle construye un modelo que aproxima a f usando una función lineal a trozos, dicho de otra manera, se reemplaza f por un modelo de plano-cortante

(*cutting plane model*), que se define de la siguiente manera

$$\check{f}_k(x) = \max_{j=1,\dots,k} \{f(y^j) + (\xi^j)^T(x - y^j)\}. \quad (3.3)$$

\check{f}_k puede ser reescrito como

$$\check{f}_k(x) = \max_{j=1,\dots,k} \{f(x^k) + (\xi^j)^T(x - x^k) - \alpha_k^j\},$$

donde α_k^j es denominado *error de linealización*, definido por

$$\alpha_k^j := f(x^k) - f(y^j) - (\xi^j)^T(x^k - y^j), \quad \forall j = 1, \dots, k.$$

Puesto que f es convexa, entonces $\alpha_k^j \geq 0$ para cada $j = 1, \dots, k$, y $\check{f}_k(x) \leq f(x)$ para todo $x \in \mathbb{R}^n$ (ver [3, 19]).

Observación 3.1.2.

- El modelo de plano cortante \check{f}_k es una aproximación por abajo de f .
- El error de linealización α_k^j mide cuán buena es la aproximación del modelo al problema original (ver Bagirov et al. (2014) [3, cap. 12]).

El siguiente punto candidato está definido por

$$y^{k+1} = x^k + d^k, \quad (3.4)$$

con d^k la dirección de búsqueda que es calculada por

$$d^k = \arg \min_{d \in \mathbb{R}^n} \left\{ \check{f}_k(x^k + d) + \frac{1}{2} d^T M_k d \right\}, \quad (3.5)$$

donde el término estabilizante $\frac{1}{2} d^T M_k d$ garantiza la existencia de la solución d^k , y la matriz simétrica definida positiva M_k acumula cierta información de segundo orden de f en una vecindad de x_k (ver Mäkelä (2002) [19]).

Para la actualización del próximo centro de estabilidad se utiliza un procedimiento al estilo de búsqueda lineal. Sea $m_L \in (0, 1/2)$ un parámetro de búsqueda lineal fijado por el usuario.

Se actualiza x^k a x^{k+1} de la siguiente manera:

$$x^{k+1} := y^{k+1}, \quad (3.6)$$

si la siguiente condición de descenso se verifica

$$f(y^{k+1}) \leq f(x^k) + m_L v_k, \quad (3.7)$$

donde v_k es el descenso previsto de f en x^k , es decir

$$v_k = \check{f}_k(y^{k+1}) - f(x^k) < 0.$$

Caso contrario, si (3.7) no se verifica, se actualiza x^{k+1} como

$$x^{k+1} = x^k. \quad (3.8)$$

Observación 3.1.3. En la literatura, la actualización (3.6) se la denomina *paso serio* y a (3.8) se la conoce como *paso nulo*.

Posteriormente, se llama al oráculo con $x = y^{k+1}$, se agrega al bundle la nueva tripleta $(y^{k+1}, f(y^{k+1}), \xi^{k+1})$, y de esta manera se crea un nuevo modelo de plano cortante \check{f}_{k+1} . Se actualiza $k = k + 1$ y se regresa a resolver (3.5).

El algoritmo termina si

$$v_k \geq -\varepsilon,$$

con $\varepsilon > 0$ un parámetro de precisión suficientemente pequeño ingresado por el usuario.

Como se menciona en Bagirov et al. (2014), para el caso que la función objetivo es convexa, el algoritmo Bundle converge globalmente hacia el mínimo y además su orden de convergencia es lineal (primer orden).

Por otro lado, nótese que el subproblema de minimización (3.5) sigue siendo un problema de minimización no diferenciable. Sin embargo, es posible transformarlo a un subproblema de minimización cuadrática que es diferenciable. Se procede de la siguiente manera: el subproblema de minimización (3.5) tiene la forma

$$\begin{aligned} & \min_{d \in \mathbb{R}^n} \left\{ \max_{j=1, \dots, k} \{f(x^k) + (\xi^j)^T d - \alpha_k^j\} + \frac{1}{2} d^T M_k d \right\}, \\ & \Leftrightarrow f(x^k) + \min_{d \in \mathbb{R}^n} \left\{ \max_{j=1, \dots, k} \{(\xi^j)^T d - \alpha_k^j\} + \frac{1}{2} d^T M_k d \right\}. \end{aligned} \quad (3.9)$$

Sea $\nu = \max_{j=1, \dots, k} \{(\xi^j)^T d - \alpha_k^j\}$, entonces el problema de minimización en (3.9) puede ser escrito como

$$\begin{aligned} & \min_{(d, \nu)} \quad \nu + \frac{1}{2} d^T M_k d \\ & \text{s.a.} \quad -\alpha_k^j + (\xi^j)^T d \leq \nu, \quad j = 1, \dots, k. \end{aligned} \quad (3.10)$$

El siguiente teorema tiene como objetivo garantizar la existencia y unicidad de la solución del problema (3.10).

Teorema 3.1.1. *El problema de minimización (3.10) tiene una única solución (d^k, ν^k) , las condiciones de optimalidad implican la existencia de multiplicadores $\{\lambda_k^j\}_{j=1}^k$ tales*

que

$$M_k d^k = \sum_{j=1}^k \lambda_k^j \xi^j,$$

$$\nu^k = -\frac{1}{2} (d^k)^T M_k d^k - \sum_{j=1}^k \lambda_k^j \alpha_k^j.$$

Demostración. El lector puede encontrar la demostración de este teorema en Nonsmooth Optimization de Mäkelä y Neittaanmäki (1992) en las páginas 115-117. \square

A continuación se presenta el algoritmo general bundle en manera de pseudocódigo

Algoritmo 3: Método Bundle

Datos: $x^1 \in \mathbb{R}^n$, $m_L \in (0, 1/2)$, $M_1 \in \mathbb{R}^{n \times n}$ simétrica definida positiva, $\varepsilon \geq 0$ y $v_0 \leq -\varepsilon$;
 Fijar $k = 1$;
 Llamar al oráculo para obtener $f(x^1)$ y $\xi^1 \in \partial f(x^1)$;
 Inicializar el bundle $\mathcal{B} = \{(x^1, f(x^1), \xi^1)\}$;
mientras $|v_{k-1}| \geq \varepsilon$ **hacer**
 Calcular la dirección de búsqueda
 $d^k = \arg \min_{d \in \mathbb{R}^n} \{\check{f}_k(x^k + d) + \frac{1}{2} d^T M_k d\}$;
 Fijar $y^{k+1} = x^k + d^k$;
 Llamar al oráculo para obtener $f(y^{k+1})$ y $\xi^{k+1} \in \partial f(y^{k+1})$;
 Calcular $v_k = \check{f}(y^{k+1}) - f(x^k)$;
 si $f(y^{k+1}) \leq f(x^k) + m_L v_k$ **entonces**
 | $x^{k+1} = y^{k+1}$; /* Paso Serio */
 en otro caso
 | $x^{k+1} = x^k$; /* Paso Nulo */
 fin
 Actualizar: $\mathcal{B} = \mathcal{B} \cup \{(y^{k+1}, f(y^{k+1}), \xi^{k+1})\}$, M_k acorde aciertos criterios de actualización y $k = k + 1$;
fin
Resultado: Solución final x^k .

Puesto que el método Bundle tiene como propiedad un orden de convergencia lineal, se desea estudiar métodos de optimización no diferenciable que posean un orden de convergencia superior, por ejemplo, Lukšan y Vlček en [18] discuten técnicas de precondicionamiento para mejorar el método Bundle. Además, se muestra que el método propuesto tiene convergencia global para funciones localmente Lipschitz continuas con semi-regularidad superior y, bajo hipótesis adicionales de convexidad fuerte, se tiene un orden de convergencia superlineal.

En la siguiente sección se describe un método de optimización para funciones no suaves que posee un orden de convergencia mayor y hace uso del método general Bundle.

3.2. Método Bundle Quasi-Newton

Entre las ideas más atractivas para tener un método de optimización no suave cuyo orden de convergencia sea mayor, es combinar el método Bundle con el método de Newton o con métodos de tipo Quasi-Newton, pues, estos dos suelen presentar una tasa de convergencia de segundo orden.

En la presente sección se presenta el método Bundle Quasi-Newton, el cual ha sido desarrollado, descrito y estudiado en los trabajos de Lemaréchal y Sagastizábal [17], Mifflin [21] y Mifflin, Sun y Qi [22]. Estas publicaciones se basan en hacer uso de la *regularización de Moreau-Yosida*, obteniendo un problema de minimización diferenciable el cual es equivalente a minimizar el problema original.

Se hará énfasis en el método Bundle Quasi-Newton propuesto por Mifflin, Sun y Qi. El mismo fue adaptado por nosotros para la resolución del problema de optimización asociado al flujo viscoplástico de Herschel-Bulkley.

Iniciaremos esta sección describiendo la regularización de Moreau-Yosida.

3.2.1. Regularización de Moreau-Yosida

Notación 3.2.1. Denotaremos por $\|\cdot\|$ la norma euclídeana en \mathbb{R}^n . Sea $M \in \mathbb{R}^{n \times n}$ una matriz simétrica definida positiva. Denotaremos por $\|\cdot\|_M$, la norma en \mathbb{R}^n inducida por M , es decir, para cualquier $x \in \mathbb{R}^n$, $\|x\|_M^2 = x^T M x$.

La regularización de Moreau-Yosida es una de las tantas maneras que existen para suavizar una función no suave. Como veremos más adelante, hacer uso de esta regularización permite la conexión entre el método Bundle y métodos que siguen el esquema Quasi-Newton para obtener un algoritmo de optimización no suave que posee un determinado orden de convergencia hacia el mínimo.

Sea $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una función convexa no necesariamente diferenciable y $M \in \mathbb{R}^{n \times n}$ una matriz simétrica definida positiva.

Definición 3.2.1. La regularización de Moreau-Yosida de f asociada a M , se define por

$$F_M(x) := \min_{y \in \mathbb{R}^n} \left\{ f(y) + \frac{1}{2} \|y - x\|_M^2 \right\}. \quad (3.11)$$

Observación 3.2.1. Gracias a la Proposición 3.a en [23] el problema de minimización del lado derecho de (3.11) posee un mínimo y es único, es decir, F_M está bien definida para cada $x \in \mathbb{R}^n$.

Notación 3.2.2. Notaremos por $p_M(x)$ al mínimo de (3.11), y este suele ser llamado punto proximal de x .

EJEMPLO 1. Sea $f(x) = |x|$, entonces la regularización de Moreau-Yosida de f asociada a $\mu > 0$ es

$$F_\mu(x) = \min_{y \in \mathbb{R}} \left\{ |y| + \frac{\mu}{2}(y - x)^2 \right\} = \begin{cases} \frac{\mu}{2}x^2, & \text{si } |x| \leq 1/\mu, \\ |x| - \frac{1}{2\mu} & \text{si } |x| > 1/\mu. \end{cases}$$

En la figura 3.1 es posible observar la regularización de Moreau-Yosida para diferentes valores de μ .

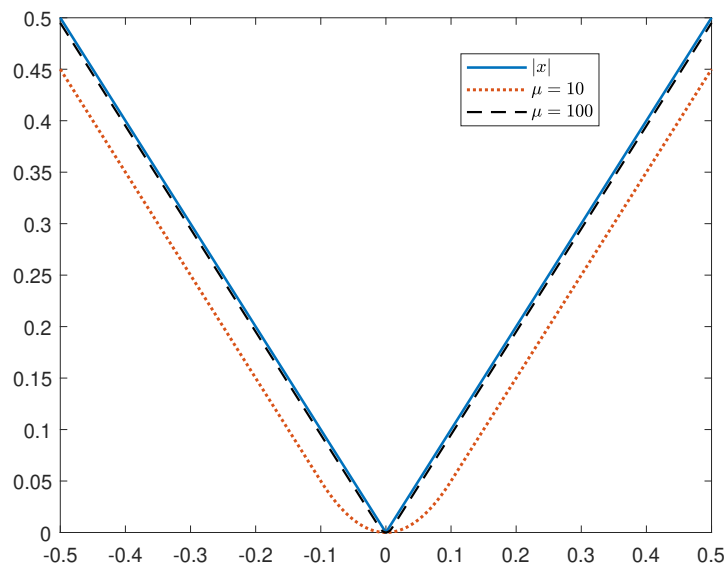


Figura 3.1: Regularización de Moreau-Yosida de $f(x) = |x|$.

Nótese que mientras el valor de μ es más grande, entonces F_μ se aproxima mejor a $f(x)$. En este caso, la regularización de Moreau-Yosida coincide con la regularización de Huber. Esta técnica fue empleada por González-Andrade en [13] en la construcción de un algoritmo de descenso preconditionado para la resolución numérica del problema de flujo.

A continuación se enlista algunas de las propiedades que presenta la regularización de Moreau-Yosida.

Proposición 3.2.1. F_M satisface las siguientes propiedades:

(I) F_M es una función convexa y continuamente diferenciable definida en \mathbb{R}^n . El gradiente de F_M en el punto x está definido por

$$\nabla F_M(x) = M(x - p(x)) \in \partial f(p(x)),$$

donde ∂f es el subdiferencial de f .

(II) ∇F_M es globalmente Lipschitz continuo con módulo $\|M\|$.

(III) El conjunto de mínimos de F_M es exactamente el conjunto de mínimos de f , y x^* minimiza f si y sólo si $\nabla F_M(x^*) = 0$ y $p(x^*) = x^*$.

Demostración. La demostración de este resultado se profundiza en [16, 23]. □

Observación 3.2.2. El tercer numeral de la Proposición 3.2.1 brinda una gran ventaja para construir un algoritmo que minimice f . En lugar de minimizar f , se busca minimizar F_M aplicando métodos de optimización suave que han sido estudiados y se conoce de antemano sus buenas propiedades de convergencia al mínimo.

Nótese que para minimizar F_M es necesario resolver un problema de minimización auxiliar, es decir, suponiendo que la iteración actual del método de minimización es x^k , entonces se debe resolver

$$\min_{y \in \mathbb{R}^n} \left\{ f(y) + \frac{1}{2} \|y - x^k\|_M^2 \right\}.$$

En consecuencia, se puede pensar que minimizar F_M presenta un costo computacional más alto que el aplicar un método directo para minimizar f . Sin embargo, la similitud de esta última ecuación con (3.5) nos permite apreciar el rol importante de las ideas del método Bundle para la resolución de este problema de minimización auxiliar. Más adelante, se describirá en detalle la manera en la que algoritmo Bundle interviene sobre este problema auxiliar para resolverlo de manera aproximada.

Ahora bien, dado que el objetivo es obtener un método de optimización cuyo orden de convergencia sea superior al de primer orden y dado que F_M es una función suave, resulta tentador aplicar el método de Newton (que tiene un orden de convergencia cuadrático) para minimizar F_M . En [16], Lemaréchal y Sagastizábal estudian información de segundo orden que puede tener F_M , es decir, la existencia de $\nabla^2 F_M$. Ellos demuestran que si f satisface una condición de crecimiento en el punto x_0 , y además $\nabla f(x_0)$ existe, entonces se garantiza la existencia de $\nabla^2 F_M(x_0)$.

En general, se está tratando con funciones no diferenciables, es así que, exigir que ∇f exista en ciertos puntos de su dominio representa una hipótesis restrictiva. Más aún, por la Proposición 3.2.1, se conoce que F_M es convexa, ∇F_M es Lipschitz continuo y además es dado explícitamente. Tomando en cuenta esto, Lemaréchal y Sagastizábal en [17], y Mifflin, Sun y Qi en [22], consideran que emplear un método Quasi-Newton es una idea natural y más viable que aplicar directamente el método de Newton.

Como en la sección anterior, se tiene los siguientes supuestos

Hipótesis 1. *Considere el siguiente problema de minimización*

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

con $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una función convexa, finitamente valuada, y además para todo $x \in \mathbb{R}^n$ es posible calcular el valor de la función $f(x)$ y al menos un subgradiente $\xi(x)$ perteneciente al subdiferencial de f en x ($\partial f(x)$). Sea $M \in \mathbb{R}^{n \times n}$ una matriz simétrica definida positiva y F_M , definida como en (3.11), la regularización de Moreau-Yosida de f asociada a M .

Suponiendo que por el momento se dispone de algún mecanismo que permite calcular rápidamente F_M y ∇F_M , se brinda un algoritmo conceptual cuya k -ésima iteración es como sigue

Algoritmo 4: Esquema de Algoritmo tipo Quasi-Newton

Paso 1. Dados x_k , $H_k \approx \nabla^2 F_M(x_k)$ y $p_M(x_k)$ solución de (3.11) con $x = x_k$ se asume disponible. Calcule $\nabla F_M(x_k) = M(x_k - p_M(x_k))$.

Paso 2. Realice la actualización de tipo Quasi-Newton

$$x_{k+1} = x_k - H_k^{-1} \nabla F_M(x_k) = x_k - H_k^{-1} M(x_k - p_M(x_k)), \quad (3.12)$$

posiblemente con algún método de búsqueda lineal para asegurar convergencia global.

Paso 3. Calcule $p_M(x_{k+1})$ de (3.11) y $\nabla F_M(x_{k+1}) = M(x_{k+1} - p_M(x_{k+1}))$.

Paso 4. Actualice H_k mediante una fórmula Quasi-Newton usando el par $(\Delta x, \Delta G)$:

$$\Delta x = x_{k+1} - x_k, \quad \Delta G = M(x_{k+1} - p_M(x_{k+1}) - x_k + p_M(x_k)).$$

En el paso 4, haciendo uso del par (u, v) debe cumplirse que la matriz actualizada satisfaga la ecuación Quasi Newton

$$H_{k+1}u = v.$$

Desde un punto de vista teórico el Algoritmo 4 converge globalmente hacia al mínimo (es suficiente con que ∇F_M es Lipschitz continuo) y posee un orden de convergencia superlineal si F_M goza de una cierta regularidad de segundo orden (ver [17] y las referencias en él).

Como fue mencionado anteriormente, la regularidad de segundo orden de F_M no es un asunto sencillo y por el momento no se desea imponer más condiciones sobre f para que el Algoritmo 4 tenga un orden de convergencia superlineal.

En resumen, nos resta abordar dos aspectos importantes del método que minimiza F_M . El primero, estudiar la metodología que se ha empleado para resolver el problema auxiliar en (3.11). En el segundo aspecto, analizar el orden de convergencia del método.

A continuación, nos adentramos a describir detalladamente el método propuesto por Mifflin, Sun y Qi en [22]. En este artículo se abarcan los dos aspectos mencionados anteriormente. En primer lugar, los autores derivan un subalgoritmo Bundle que resuelve aproximadamente el problema auxiliar (3.11) en un número finito de pasos. Posteriormente, proponen un método que sigue el esquema del algoritmo Quasi-Newton (Algoritmo 4) con una actualización de tipo BFGS. Además, demuestran que si se cumplen algunos supuestos de regularidad de segundo orden de F_M en el mínimo, entonces el método converge global y superlinealmente hacia la solución.

En primera instancia, se describirá la utilidad del método Bundle para resolver de manera aproximada (3.11), considere entonces que se tienen los supuestos de la hipótesis 1, y además, se tiene disponible el oráculo que devuelve $(f(x), \xi(x))$ con $\xi(x) \in \partial f(x)$, para cualquier $x \in \mathbb{R}^n$.

3.2.2. Subalgoritmo Bundle para la aproximación de la regularización de Moreau-Yosida

Sea $x \in \mathbb{R}^n$ cualquiera. Las ideas presentadas en el método general Bundle juegan un rol importante para la aproximación de $F_M(x)$ y $\nabla F_M(x)$ de acuerdo al desarrollo en [21] y [22]. Sea $d = y - x$ en (3.11), minimizando sobre d en lugar de y se tiene

$$F_M(x) = \min_{d \in \mathbb{R}^n} \left\{ f(x + d) + \frac{1}{2} d^T M d \right\}, \quad (3.13)$$

y $p_M(x) = x + d^*$, donde $d^* = \arg \min_{d \in \mathbb{R}^n} \{ f(x + d) + \frac{1}{2} d^T M d \}$. Ahora considere aproximar $f(x + d)$ por un modelo de plano cortante, esto es

$$\check{f}(x + d) = \max_{i=1, \dots, j} \{ f(u^i) + (\xi^i)^T (x + d - u^i) \},$$

donde las tripletas $(u^i, f(u^i), \xi^i)$ con $\xi^i = \xi(u^i) \in \partial f(u^i)$, constituyen un arreglo (*bundle*) \mathcal{B} que ha sido generado secuencialmente partiendo desde x y $\xi(x) \in \partial f(x)$ y, posiblemente, algún subconjunto del bundle anterior usado para generar x . Nuevamente, gracias a que f es convexa se tiene que

$$f(x+d) \geq \check{f}(x+d). \quad (3.14)$$

Se define el siguiente error de linealización

$$e(x, u^i) = f(x) - f(u^i) - (\xi^i)^T(x - u^i),$$

entonces $\check{f}(x+d)$ puede ser escrito como

$$\check{f}(x+d) = f(x) + \max_{i=1, \dots, j} \{(\xi^i)^T d - e(x, u^i)\}. \quad (3.15)$$

Considerando la forma de $\check{f}(x+d)$ como en (3.15), se define \check{F}_M de la siguiente manera

$$\begin{aligned} \check{F}_M(x) &:= \min_{d \in \mathbb{R}^n} \left\{ \check{f}(x+d) + \frac{1}{2} d^T M d \right\} \\ &= f(x) + \min_{d \in \mathbb{R}^n} \left\{ \max_{i=1, \dots, j} \{(\xi^i)^T d - e(x, u^i)\} + \frac{1}{2} d^T M d \right\}. \end{aligned} \quad (3.16)$$

De (3.14) y la definición de $F_M(x)$, se tiene que

$$\check{F}_M(x) \leq F_M(x),$$

así, $\check{F}_M(x)$ es una subaproximación del valor aún desconocido $F_M(x)$.

Nótese que el problema de minimización en el lado derecho de (3.16) tiene similitud con el problema de minimización en (3.9), en consecuencia, es posible transformar este problema de minimización en un problema de minimización cuadrática. Sea $\nu = \max_{i=1, \dots, j} \{(\xi^i)^T d(x) - e(x, u^i)\}$, se tiene el siguiente problema de minimización

$$\begin{aligned} \min_{(d, \nu)} \quad & \nu + \frac{1}{2} d^T M d \\ \text{s.a.} \quad & -e(x, u^i) + (\xi^i)^T d \leq \nu, \quad i = 1, \dots, j. \end{aligned} \quad (3.17)$$

Las condiciones de optimalidad del problema (3.17) implican la existencia de multipli-

cadores no negativos $\lambda_i(x)$ que suman uno y verifican

$$Md(x) = - \sum_{i=1}^j \lambda_i(x) \xi^i,$$

$$\nu(x) = (\xi^i)^T d(x) - e(x, u^i), \text{ si } \lambda_i(x) > 0,$$

con $(d(x), \nu(x))$ la única solución de (3.17) (ver [21, sec. 5]).

Sea $(d(x), \nu(x))$ la solución de (3.17), se sigue que

$$\check{F}_M(x) = f(x) + \nu(x) + \frac{1}{2} (d(x))^T Md(x).$$

Por otro lado, sea $a(x) = x + d(x)$ una aproximación de $p_M(x)$ y sea

$$\hat{F}_M(x) = f(a(x)) + \frac{1}{2} (d(x))^T Md(x),$$

puesto que $p_M(x)$ es el mínimo de (3.11) y es único (ver observación 3.2.1), entonces

$$F_M(x) \leq \hat{F}_M(x),$$

la igualdad se consigue si y sólo si $a(x) = p_M(x)$.

Como consecuencia de estas conclusiones, se tiene el siguiente resultado

Lema 3.2.1.

- (I) $\check{F}_M(x) \leq F_M(x) \leq \hat{F}_M(x)$,
- (II) $F_M(x) = \hat{F}_M(x)$ si y sólo si $a(x) = p_M(x)$.

Este lema juega un rol importante en el diseño del método. Sea

$$\varepsilon(x) = \hat{F}_M(x) - \check{F}_M(x). \tag{3.18}$$

Se acepta que $a(x)$ es una aproximación de $p_M(x)$ si se cumple que

$$\varepsilon(x) \leq \delta(x) \text{ mín } \{ (d(x))^T Md(x), N \}, \tag{3.19}$$

donde $\delta(x)$ y N son números positivos dados, y $\delta(x)$ permanece fijo durante el subproceso Bundle. Si (3.19) no se verifica, entonces sean $u^{j+1} = x + d(x)$ y $\xi^{j+1} \in \partial f(u^{j+1})$, agregue el término $(\xi^{j+1})^T d - e(x, u^{j+1})$ a (3.15). Reemplace $j = j + 1$, actualice el arreglo $\mathcal{B} = \mathcal{B} \cup \{(u^{j+1}, f(u^{j+1}), \xi^{j+1})\}$. Entonces, resuelva el nuevo subproblema en (3.17) que determina un nuevo $d(x)$ y un nuevo $\varepsilon(x)$ a ser probados en (3.19).

En el algoritmo subsiguiente, se describe el subproceso Bundle que retorna una buena aproximación de $p_M(x)$.

Algoritmo 5: Subalgoritmo Bundle

Paso 0. Llamar al oráculo para obtener $f(x)$ y $\xi \in \partial f(x)$, inicializar el arreglo $\mathcal{B} = \{(x, f(x), \xi)\}$, o bien, usar el arreglo anterior que guarda información de iteraciones pasadas y ha sido empleado para generar x .

Paso 1. Hallar $d(x)$ solución de (3.17), y fijar $a(x) = x + d(x)$, llamar al oráculo para obtener $(f(a(x)), \xi^a)$.

Paso 2. Calcular $\check{F}_M(x)$ y $\hat{F}_M(x)$, y determinar $\varepsilon(x)$ como en (3.18).

Paso 3. Determinar si (3.19) se satisface, si es así **PARAR**. Caso contrario, sea $w^{j+1} = a(x)$ y $\xi^{j+1} = \xi^a$, agregue el término $(\xi^{j+1})^T d - e(x, w^{j+1})$ a (3.15). Actualizar $j = j + 1$ y el arreglo $\mathcal{B} = \mathcal{B} \cup \{(w^{j+1}, f(w^{j+1}), \xi^{j+1})\}$; y regresar al paso 1.

Resultado: $a(x)$ aproximación de $p_M(x)$.

Ahora, sea $G_M(x) := \nabla F_M(x)$, se define $\tilde{G}_M(x)$ de la siguiente manera

$$\tilde{G}_M(x) = M(x - a(x)) = -Md(x).$$

A continuación, se presenta dos lemas importantes para el desarrollo del algoritmo general. El primero, establece una relación de aproximación entre $G_M(x)$ y $\tilde{G}_M(x)$, consecuentemente, es posible aproximar el gradiente de $F_M(x)$ gracias al subproceso Bundle. El segundo lema señala que este subalgoritmo finaliza en un número finito de pasos.

Lema 3.2.2.

$$\begin{aligned} \|G_M(x) - \tilde{G}_M(x)\|_{M^{-1}} &= \|p(x) - a(x)\|_M \leq \sqrt{2\varepsilon(x)}, \\ \|G_M(x) - \tilde{G}_M(x)\| &\leq \sqrt{2\varepsilon(x)}\|M\|. \end{aligned}$$

Lema 3.2.3. *Si x no minimiza f , entonces, después de un número finito de pasos del subalgoritmo Bundle, es posible encontrar una solución $d(x)$ tal que (3.19) se verifica.*

Las demostraciones de estos dos lemas se encuentran de manera detallada en [22].

Observación 3.2.3. Gracias al lema 3.2.3 es posible usar el subproceso Bundle (Algoritmo 5) para hallar un vector $d(x)$ tal que (3.19) se verifica si x no la solución óptima. Esto resulta esencial para el algoritmo Bundle Quasi-Newton, pues, es posible aproximar $F_M(x)$ y $\nabla F_M(x)$, siendo que x es arbitrario.

El siguiente resultado ayuda a obtener un criterio de parada práctico para que el

algoritmo Bundle Quasi-Newton se detenga. Su demostración puede ser estudiada en [21].

Teorema 3.2.1. *Sea $(d(x), \nu(x))$ solución de (3.17), para cualquier $z \in \mathbb{R}^n$ se cumple que*

$$f(z) \geq f(x) + \nu(x) + d(x)^T M d(x) - (z - x)^T M d(x),$$

y

$$\nu(x) \leq -d(x)^T M d(x) \leq 0.$$

Gracias a este teorema es posible notar que si $\nu(x) = 0$ entonces x minimiza f . Así, el algoritmo termina si el subproceso Bundle arroja una solución $\nu(x)$ tal que

$$|\nu(x)| \leq \text{tol},$$

donde tol es un parámetro de precisión positivo suficientemente pequeño.

Tomando en cuenta todo lo anterior, se describe el algoritmo tipo Bundle Quasi-Newton.

Notación 3.2.3. Se denota $\varepsilon_k = \varepsilon(x^k)$, $a^k = a(x^k)$, $d^k = (x^k)$, etc.

Algoritmo 6: Algoritmo tipo Bundle Quasi-Newton

Datos: Sean σ, ρ, N y tol números positivos tales que $\sigma < 1/2$, $\rho < 1$ y $\text{tol} \lll 1$. Sea $\{\delta_k\}$ una sucesión de números positivos tales que $\sum_{k=0}^{\infty} \delta_k < \infty$. Sea $x^0 \in \mathbb{R}^n$ un candidato inicial, $M, B_0 \in \mathbb{R}^{n \times n}$ simétricas definidas positivas y sea \mathcal{B} un bundle.

Paso 0. (Inicialización) Fijar $k := 0$, hallar (d^0, ν^0) y ε_0 mediante el subalgoritmo bundle (Algoritmo 5) tales que

$$\varepsilon_0 \leq \delta^0 \{ (d^0)^T M d^0, N \},$$

por ejemplo, inicializando el bundle $\mathcal{B} = \{(x^0, f^0, \xi^0)\}$.

Paso 1. (Calcular la dirección de búsqueda) Si $|\nu^k| < \text{tol}$ parar con x^k optimal, caso contrario calcule

$$s^k = -B_k^{-1} \tilde{G}_M(x^k). \quad (3.20)$$

Paso 2. (Búsqueda Lineal) Iniciando con $i_k = 0$, sea i_k el entero no negativo más pequeño tal que

$$\tilde{F}_M(x^k + \rho^{i_k} s^k) \leq \hat{F}_M(x^k) + \sigma \rho^{i_k} (s^k)^T \tilde{G}_M(x^k), \quad (3.21)$$

donde $\check{F}_M(x^k + \rho^{i_k} s^k)$ es la subaproximación de F_M en $x^k + \rho^{i_k} s^k$ y cumple que

$$\begin{aligned} \hat{F}_M(x^k + \rho^{i_k} s^k) - \check{F}_M(x^k + \rho^{i_k} s^k) \\ \leq \delta_{k+1} \min \left\{ (d(x^k + \rho^{i_k} s^k))^T M d(x^k + \rho^{i_k} s^k), N \right\}. \end{aligned} \quad (3.22)$$

Fijar $\tau_k := \rho^{i_k}$ y $x^{k+1} = x^k + \tau_k s^k$.

Paso 3. (Actualización de la matriz Quasi-Newton) Sea $\Delta x = x^{k+1} - x^k$ y $\Delta y^k = \tilde{G}_M(x^{k+1}) - \tilde{G}_M(x^k)$. Si $(\Delta x^k)^T \Delta y^k > 0$, actualizar B_k a B_{k+1} tal que B_{k+1} es simétrica definida positiva y satisface la ecuación Quasi-Newton

$$B_{k+1} \Delta x^k = \Delta y^k.$$

Caso contrario, fijar $B_{k+1} = M$. Actualizar $k := k + 1$ y regresar al paso 1.

Observación 3.2.4.

- Si $x^k + \rho^{i_k} s^k$ no minimiza f , gracias al lema 3.2.3 es posible hallar un vector $d(x^k + \rho^{i_k} s^k)$ que satisface (3.22) luego de un número finito de pasos del subproceso Bundle. Así, el paso 2 se procede de la manera siguiente: Primero calcule $d(x^k + \rho^{i_k} s^k)$ y luego determine si (3.21) se cumple. De no ser así, incremente i_k en 1 y repita este proceso con el nuevo punto $x^k + \rho^{i_k} s^k$; caso contrario fijar $\tau_k = \rho^{i_k}$ y $x^{k+1} = x^k + \tau_k s^k$, y continúe al paso 3.
- Los autores muestran que i_k está bien definido en cada iteración del algoritmo (ver [22], teorema 3.1), de este modo es posible encontrar τ_k tal que (3.21) y (3.22) se satisfacen en cada iteración del Algoritmo 6.

En cuanto a los aspectos de convergencia del algoritmo tipo Bundle Quasi-Newton (ver [22], secciones 4 y 5), se muestra que si f es acotada inferiormente y si se supone que las sucesiones de matrices $\{B_k\}$ y $\{B_k^{-1}\}$ (generadas por el método) permanecen acotadas. Entonces el Algoritmo 6 converge globalmente hacia el mínimo de f .

Adicionalmente, si la actualización Quasi-Newton es realizada mediante la actualización BFGS, y suponiendo que además, f es fuertemente convexa, G_M es Fréchet diferenciable en \bar{x} y su derivada direccional es radialmente Lipschitz continua en \bar{x} , donde \bar{x} es el mínimo de f . Entonces, el algoritmo tipo Bundle BFGS (algoritmo tipo Bundle Quasi-Newton empleando la actualización Quasi-Newton BFGS) converge superlinealmente hacia el mínimo.

Capítulo 4

Algoritmo no suave para la resolución del problema de flujo

Este capítulo está dedicado al desarrollo de un algoritmo no diferenciable para la resolución numérica del problema de flujo de un fluido de Herschel-Bulkley. En primer lugar estudiaremos el problema de optimización de interés. Posteriormente, empleando los conceptos revisados en el Capítulo 3, construiremos el método de minimización no suave que resuelve el problema de flujo. Finalmente, mostraremos que este converge globalmente hacia el mínimo y analizaremos su orden de convergencia.

4.1. Estudio analítico del problema de flujo

Empezaremos analizando el problema de optimización de interés, discutiremos la existencia y unicidad de la solución, las propiedades geométricas y de diferenciabilidad que presenta el funcional a minimizar.

Introducimos la notación que se usará a lo largo de todo el capítulo

Notación 4.1.1. Notaremos por $\Omega \subset \mathbb{R}^n$ un conjunto abierto, acotado y suficientemente regular. El producto escalar en \mathbb{R}^n y la norma euclídeana serán denotados por (\cdot, \cdot) y $|\cdot|$, respectivamente. El producto en dualidad de un espacio de Banach V y su espacio dual V^* es representado por $\langle \cdot, \cdot \rangle$, y $\|\cdot\|_V$ es la norma de V . Dado $1 < p < \infty$, su exponente conjugado es denotado por p' ($1/p + 1/p' = 1$). Usaremos la notación clásica para el espacio de Sobolev $W_0^{1,p}(\Omega)$ y la notación $W^{-1,p'}(\Omega)$ para su espacio dual. Finalmente, notaremos $\mathbf{L}^p(\Omega) := L^p(\Omega) \times L^p(\Omega)$, para $1 < p < \infty$.

Nuestro interés es resolver numéricamente el problema de flujo, que es el siguiente

$$\min_{u \in W_0^{1,p}(\Omega)} J(u) := \frac{1}{p} \int_{\Omega} |\nabla u|^p dx + g \int_{\Omega} |\nabla u| dx - \int_{\Omega} f u dx, \quad (4.1)$$

con $1 < p < \infty$, $g > 0$ y $f \in L^{p'}(\Omega)$.

Para garantizar la existencia y unicidad de la solución de (4.1), necesitamos exhibir las siguientes propiedades que posee el funcional J .

Proposición 4.1.1. *El funcional J definido en (4.1) es continuo, estrictamente convexo y coercivo.*

Demostración. Comenzaremos probando la continuidad de J , sea $\{u_k\}$ una sucesión en $W_0^{1,p}(\Omega)$ tal que $u_k \rightarrow u$ en $W_0^{1,p}(\Omega)$, cuando $k \rightarrow \infty$. Mostremos que $J(u_k) \rightarrow J(u)$, cuando $k \rightarrow \infty$. En efecto, gracias al Teorema 2.1.1, existe una subsucesión $\{u_k\}$ (denotada de la misma manera) y una función $w \in L^p(\Omega)$ tal que

- $\nabla u_k \rightarrow \nabla u$ en $L^p(\Omega)$, cuando $k \rightarrow \infty$,
- $\nabla u_k(x) \rightarrow \nabla u(x)$ c.t.p. en Ω , cuando $k \rightarrow \infty$,
- Para todo k , $|\nabla u_k(x)| \leq w(x)$ c.t.p. en Ω .

Por otro lado, notemos lo siguiente

$$\begin{aligned} |J(u_k) - J(u)| &= \left| \frac{1}{p} \int_{\Omega} |\nabla u_k|^p - |\nabla u|^p dx + g \int_{\Omega} |\nabla u_k| - |\nabla u| dx - \int_{\Omega} f(u_k - u) dx \right|, \\ &\leq \frac{1}{p} \int_{\Omega} ||\nabla u_k|^p - |\nabla u|^p| dx + g \int_{\Omega} ||\nabla u_k| - |\nabla u|| dx \\ &\quad + \int_{\Omega} |f(u_k - u)| dx, \\ &\leq \frac{1}{p} \int_{\Omega} ||\nabla u_k|^p - |\nabla u|^p| dx + g \int_{\Omega} |\nabla u_k - \nabla u| dx \\ &\quad + \|f\|_{L^{p'}} \|u_k - u\|_{L^p} dx, \end{aligned}$$

la última desigualdad se obtiene haciendo uso de la desigualdad triangular inversa y la desigualdad de Hölder para espacios L^p . Ahora, note que el término derecho de la desigualdad puede ser reescrito como

$$\frac{1}{p} \int_{\Omega} ||\nabla u_k|^p - |\nabla u|^p| dx + g \|u_k - u\|_{W_0^{1,1}(\Omega)} + \|f\|_{L^{p'}} \|u_k - u\|_{L^p},$$

en virtud de los Teoremas 2.1.2 y 2.1.3, existe $C > 0$ tal que

$$|J(u_k) - J(u)| \leq C \left(\int_{\Omega} ||\nabla u_k|^p - |\nabla u|^p| dx + \|u_k - u\|_{W_0^{1,p}} \right). \quad (4.2)$$

Por el segundo ítem podemos inferir que $|\nabla u_k(x)|^p \rightarrow |\nabla u(x)|^p$ c.t.p en Ω , cuando $k \rightarrow \infty$; y además, gracias al tercer ítem, para todo k se sigue que

$$||\nabla u_k(x)|^p - |\nabla u(x)|^p| \leq |\nabla u_k(x)|^p + |\nabla u(x)|^p \leq w(x)^p + |\nabla u(x)|^p \in L^1(\Omega),$$

en consecuencia, gracias al Teorema de convergencia dominada de Lebesgue se deduce que

$$\int_{\Omega} ||\nabla u_k|^p - |\nabla u|^p| dx \xrightarrow[k \rightarrow \infty]{} 0.$$

Así, tomando el límite en (4.2) cuando $k \rightarrow \infty$ se concluye que $J(u_k) \rightarrow J(u)$. Entonces J es continuo.

Probemos ahora que J es estrictamente convexo. Siguiendo la definición 2.1.1, sean $u, v \in W_0^{1,p}(\Omega)$, $u \neq v$, y $\lambda \in [0, 1]$ arbitrarios pero fijos. Notemos lo siguiente

$$\begin{aligned} J(\lambda u + (1 - \lambda)v) &= \frac{1}{p} \int_{\Omega} |\nabla(\lambda u + (1 - \lambda)v)|^p dx + g \int_{\Omega} |\nabla(\lambda u + (1 - \lambda)v)| dx \\ &\quad - \int_{\Omega} f(\lambda u + (1 - \lambda)v) dx, \\ &= \frac{1}{p} \int_{\Omega} |\lambda \nabla u + (1 - \lambda) \nabla v|^p dx + g \int_{\Omega} |\lambda \nabla u + (1 - \lambda) \nabla v| dx \\ &\quad - \lambda \int_{\Omega} f u dx - (1 - \lambda) \int_{\Omega} f v dx. \end{aligned}$$

Dado que $1 < p < \infty$, entonces la función $\mathbb{R}^N \ni x \mapsto |x|^p$ es estrictamente convexa (ver [2, pág. 91]). Por consiguiente, vemos que

$$\begin{aligned} J(\lambda u + (1 - \lambda)v) &< \frac{1}{p} \int_{\Omega} \lambda |\nabla u|^p + (1 - \lambda) |\nabla v|^p dx \\ &\quad + g \int_{\Omega} \lambda |\nabla u| + (1 - \lambda) |\nabla v| dx - \lambda \int_{\Omega} f u dx - (1 - \lambda) \int_{\Omega} f v dx, \\ &= \lambda \left(\frac{1}{p} \int_{\Omega} |\nabla u|^p dx + g \int_{\Omega} |\nabla u| dx - \int_{\Omega} f u dx \right) \\ &\quad + (1 - \lambda) \left(\frac{1}{p} \int_{\Omega} |\nabla v|^p dx + g \int_{\Omega} |\nabla v| dx - \int_{\Omega} f v dx \right), \\ &= \lambda J(u) + (1 - \lambda) J(v). \end{aligned}$$

De donde,

$$J(\lambda u + (1 - \lambda)v) < \lambda J(u) + (1 - \lambda) J(v),$$

puesto que u, v y λ son arbitrarios se sigue que J es estrictamente convexo.

Finalmente, demostremos que J es coercivo. Sea $\{u_k\}$ una sucesión en $W_0^{1,p}(\Omega)$ tal

que $\|u_k\|_{W_0^{1,p}(\Omega)} \rightarrow \infty$ cuando $k \rightarrow \infty$. Observemos lo siguiente

$$\begin{aligned}
J(u_k) &= \frac{1}{p} \int_{\Omega} |\nabla u_k|^p dx + \underbrace{g \int_{\Omega} |\nabla u_k| dx}_{\geq 0} - \int_{\Omega} f u_k dx, \\
&\geq \frac{1}{p} \int_{\Omega} |\nabla u_k|^p dx - \int_{\Omega} f u_k dx \\
&\geq \frac{1}{p} \int_{\Omega} |\nabla u_k|^p dx - \|f\|_{L^{p'}} \|u_k\|_{L^p} \\
&\geq \frac{1}{p} \|u_k\|_{W_0^{1,p}(\Omega)}^p - C \|f\|_{L^{p'}} \|u_k\|_{W_0^{1,p}(\Omega)} \\
&\geq C \left(\|u_k\|_{W_0^{1,p}(\Omega)}^p - \|u_k\|_{W_0^{1,p}(\Omega)} \right).
\end{aligned}$$

Puesto que $p > 1$, entonces el último término de la desigualdad converge a ∞ cuando $k \rightarrow \infty$. Esto nos permite concluir que

$$J(u_k) \xrightarrow[k \rightarrow \infty]{} \infty.$$

Entonces J es coercivo. □

Observación 4.1.1. Gracias al Teorema 2.1.4, se tiene que J es débilmente semicontinuo inferior.

Por las propiedades mostradas en la Proposición anterior se tiene como consecuencia el siguiente Teorema.

Teorema 4.1.1. *El problema de optimización (4.1) tiene solución y además es única.*

Demostración. Es una consecuencia directa ya que, gracias a la Proposición 4.1.1, las hipótesis del Teorema 2.1.5 se verifican. □

Observación 4.1.2. Sea $\bar{u} \in W_0^{1,p}(\Omega)$ tal que

$$J(\bar{u}) = \min_{u \in W_0^{1,p}(\Omega)} J(u),$$

entonces

$$J(u) \geq J(\bar{u}) \quad \forall u \in W_0^{1,p}(\Omega).$$

Esta última desigualdad nos dice que J es acotado inferiormente.

Nos resta analizar la diferenciabilidad de J . Siguiendo las ideas de Gabay y Mercier en [11], expresamos el funcional J de la siguiente manera

$$J(u) = J_1(u) + gJ_2(u),$$

donde

$$J_1(u) = \frac{1}{p} \int_{\Omega} |\nabla u|^p dx - \int_{\Omega} fu dx \quad \text{y} \quad J_2(u) = \int_{\Omega} |\nabla u| dx.$$

En consecuencia, el subdiferencial de J en el punto u queda determinado por

$$\partial J(u) = \partial J_1(u) + g\partial J_2(u),$$

esta igualdad es cierta puesto que J_1 y J_2 son funcionales débiles semicontinuos inferiores (ver [11]). Así, vamos a estudiar los subdiferenciales asociados a J_1 y J_2 , respectivamente. Y, de esta manera, lograr caracterizar el subdiferencial de J .

Se tiene que J_1 es un funcional Fréchet diferenciable en $W_0^{1,p}(\Omega)$ y su diferencial asociada está dada por (ver [2, pág. 91])

$$J'_1(u)v = \int_{\Omega} |\nabla u|^{p-2} (\nabla u, \nabla v) dx - \int_{\Omega} fv dx \quad (4.3)$$

Por esta razón, $\partial J_1(u) = \{J'_1(u)\}$ para todo $u \in W_0^{1,p}(\Omega)$.

Por otro lado, J_2 no es diferenciable. En efecto, nótese que J_2 es equivalente a

$$J_2(u) = \|\nabla u\|_{L^1(\Omega)}.$$

Como es conocido, la norma en $L^1(\Omega)$ no es diferenciable en cero. En consecuencia, J_2 no es diferenciable para $u \in W_0^{1,p}(\Omega)$ tal que $\nabla u = 0$. Sin embargo, gracias al Teorema 2.1.3 es posible deducir que J_2 no es diferenciable en cero.

La siguiente Proposición nos brinda información de la diferenciable de J_2 en los puntos que son distintos de cero.

Proposición 4.1.2. *El funcional J_2 es Fréchet diferenciable para todo $u \in W_0^{1,p}(\Omega)$ tal que $u \neq 0$. Su diferencial asociada está determinada por*

$$J'_2(u)v = \int_{\Omega} \frac{(\nabla u, \nabla v)}{|\nabla u|} dx,$$

para todo $v \in W_0^{1,p}(\Omega)$.

Demostración. Sea $u \in W_0^{1,p}(\Omega)$ tal que $u \neq 0$ y sea $v \in W_0^{1,p}(\Omega)$ arbitrario pero fijo. Empecemos considerando la función $\mathbb{R}^N \ni x \mapsto \varphi(x) = |x|$, la cual es continuamente diferenciable en todo punto de \mathbb{R}^N a excepción de 0. Así, para todo $x, y \in \mathbb{R}^N$ con $x \neq 0$, se tiene que

$$\lim_{t \rightarrow 0} \frac{\varphi(x + ty) - \varphi(x)}{t} = \frac{(x, y)}{|x|}.$$

De donde, se obtiene que

$$\lim_{t \rightarrow 0} \frac{|\nabla u(x) + t\nabla v(x)| - |\nabla u(x)|}{t} = \frac{(\nabla u(x), \nabla v(x))}{|\nabla u(x)|}, \quad \text{c.t.p. en } \Omega.$$

Ahora, gracias al Teorema del valor medio, existe $\theta \in \mathbb{R}$ tal que $|\theta| \leq |t|$ y además

$$\begin{aligned} \left| \frac{|\nabla u(x) + t\nabla v(x)| - |\nabla u(x)|}{t} \right| &\leq \left| \frac{(\nabla u + \theta\nabla v, \nabla v)}{|\nabla u + \theta\nabla v|} \right| \\ &\leq |\nabla v| \in L^1(\Omega). \end{aligned}$$

Así, por el Teorema de convergencia dominada de Lebesgue se sigue que

$$\lim_{t \rightarrow 0} \frac{J_2(u + tv) - J_2(u)}{t} = \lim_{t \rightarrow 0} \int_{\Omega} \frac{|\nabla u + t\nabla v| - |\nabla u|}{t} dx = \int_{\Omega} \frac{(\nabla u, \nabla v)}{|\nabla u|} dx.$$

Puesto que v es arbitrario se concluye que J_2 es Gâteaux diferenciable en u y

$$D_G J_2(u)v = \int_{\Omega} \frac{(\nabla u, \nabla v)}{|\nabla u|} dx.$$

Mostremos ahora que $D_G J_2 : W_0^{1,p}(\Omega) \rightarrow W^{-1,p'}(\Omega)$ es continuo. Consideremos una sucesión $\{u_k\}$ en $W_0^{1,p}(\Omega)$ tal que $u_k \rightarrow u \in W_0^{1,p}(\Omega)$, más aún, en virtud del Teorema 2.1.1 (tomando subsucesiones) es posible asumir que

- $\nabla u_k \rightarrow \nabla u$ en $L^1(\Omega)$, cuando $k \rightarrow \infty$,
- $\nabla u_k(x) \rightarrow \nabla u(x)$ c.t.p. en Ω , cuando $k \rightarrow \infty$.

Notemos lo siguiente

$$(D_G J_2(u_k) - D_G J_2(u))v = \int_{\Omega} \left(\frac{\nabla u_k}{|\nabla u_k|} - \frac{\nabla u}{|\nabla u|}, \nabla v \right) dx.$$

En adición

$$\begin{aligned} &\left| \int_{\Omega} \left(\frac{\nabla u_k}{|\nabla u_k|} - \frac{\nabla u}{|\nabla u|}, \nabla v \right) dx \right| \\ &\leq \left(\int_{\Omega} \left| \frac{\nabla u_k}{|\nabla u_k|} - \frac{\nabla u}{|\nabla u|} \right|^{\frac{p}{p-1}} dx \right)^{\frac{p-1}{p}} \left(\int_{\Omega} |\nabla v|^p dx \right)^{1/p} \\ &\leq \left(\int_{\Omega} \left| \frac{\nabla u_k}{|\nabla u_k|} - \frac{\nabla u}{|\nabla u|} \right|^{\frac{p}{p-1}} dx \right)^{\frac{p-1}{p}} \|v\|_{W_0^{1,p}(\Omega)}. \end{aligned}$$

Por tanto

$$\begin{aligned} \|D_G J_2(u_k) - D_G J_2(u)\|_{W^{-1,p'}(\Omega)} &= \sup_{\|v\|=1} \{|(D_G J_2(u_k) - D_G J_2(u))v|\} \\ &\leq \left(\int_{\Omega} \left| \frac{\nabla u_k}{|\nabla u_k|} - \frac{\nabla u}{|\nabla u|} \right|^{\frac{p}{p-1}} dx \right)^{\frac{p-1}{p}}. \end{aligned}$$

Ahora, sabemos que

$$\frac{\nabla u_k}{|\nabla u_k|} \rightarrow \frac{\nabla u}{|\nabla u|},$$

cuando $k \rightarrow \infty$, para c.t.p. en Ω . Además, nótese que

$$\left| \frac{\nabla u_k}{|\nabla u_k|} - \frac{\nabla u}{|\nabla u|} \right|^{\frac{p}{p-1}} \leq 2^{\frac{p}{p-1}} \in L^1(\Omega).$$

Así, por el Teorema de convergencia dominada de Lebesgue, se tiene que

$$\left(\int_{\Omega} \left| \frac{\nabla u_k}{|\nabla u_k|} - \frac{\nabla u}{|\nabla u|} \right|^{\frac{p}{p-1}} dx \right)^{\frac{p-1}{p}} \xrightarrow[k \rightarrow \infty]{} 0.$$

En consecuencia, $\|D_G J_2(u_k) - D_G J_2(u)\|_{W^{-1,p'}(\Omega)} \rightarrow 0$ cuando $k \rightarrow \infty$.

Nótese que esto se verifica para una subsucesión de la sucesión original $\{u_k\}$. En virtud de lo expuesto en [2, pág. 20], es posible concluir que $D_G J_2(u_k) \rightarrow D_G J_2(u)$ en $W^{-1,p'}(\Omega)$. Por consiguiente, se deduce que J_2 es Fréchet diferenciable para $u \neq 0$ y además

$$J'_2(u)v = \int_{\Omega} \frac{(\nabla u, \nabla v)}{|\nabla u|} dx.$$

□

Por otra parte, puesto que J_2 es convexo (la norma es una función convexa) entonces el subdiferencial de J_2 en 0 está definido por

$$\begin{aligned} \partial J_2(0) &= \left\{ \psi \in W^{-1,p'}(\Omega) : J_2(v) - J_2(0) \geq \langle \psi, v - 0 \rangle, \quad \forall v \in W_0^{1,p}(\Omega) \right\}, \\ &= \left\{ \psi \in W^{-1,p'}(\Omega) : J_2(v) \geq \langle \psi, v \rangle \quad \forall v \in W_0^{1,p}(\Omega) \right\}. \end{aligned} \quad (4.4)$$

Por ende, es posible caracterizar el subdiferencial de J_2 en 0 como sigue

$$\psi \in \partial J_2(0) \Leftrightarrow \langle \psi, v \rangle \leq J_2(v), \quad \forall v \in W_0^{1,p}(\Omega). \quad (4.5)$$

Observación 4.1.3. Puesto que $C_c^\infty(\Omega) \subset W_0^{1,p}(\Omega) \hookrightarrow W^{-1,p'}(\Omega)$ (ver [5, pág. 291]),

para $\psi \in C_c^\infty(\Omega)$, se deduce la siguiente caracterización

$$\psi \in \partial J_2(0) \Leftrightarrow \int_{\Omega} (\nabla \psi, \nabla v) \, dx \leq \int_{\Omega} |\nabla v| \, dx.$$

Por Ejemplo, si ψ es tal que $|\nabla \psi(x)| \leq 1$ para todo $x \in \Omega$, entonces $\psi \in \partial J_2(0)$.

Tomando todo lo anterior en cuenta, es posible expresar el subdiferencial de J_2 como

$$\partial J_2(u) = \begin{cases} \{J_2'(u)\} & \text{si } u \neq 0, \\ \{\psi \in W^{-1,p'}(\Omega) : \psi \text{ verifica (4.5)}\} & \text{si } u = 0. \end{cases}$$

Por tanto, el subdiferencial de J está constituido por

$$\partial J(u) = \begin{cases} \{J_1'(u) + gJ_2'(u)\} & \text{si } u \neq 0, \\ \{J_1'(0) + g\psi : \psi \in \partial J_2(0)\} & \text{si } u = 0. \end{cases} \quad (4.6)$$

Observación 4.1.4. De la forma que tiene el subdiferencial de J en (4.6) es posible observar que este funcional no es diferenciable en 0. No obstante, es diferenciable en $W_0^{1,p}(\Omega) \setminus \{0\}$.

EJEMPLO 2. Tomando $\psi \in C_c^\infty(\Omega) \cap \partial J_2(0)$. Por la caracterización vista en la Observación 4.1.3, es posible expresar (4.6) en su forma funcional, resultando lo siguiente

$$\partial J(u)v = \begin{cases} \int_{\Omega} |\nabla u|^{p-2} (\nabla u, \nabla v) \, dx + g \int_{\Omega} \frac{(\nabla u, \nabla v)}{|\nabla u|} \, dx - \int_{\Omega} f v \, dx, & \text{si } u \neq 0, \\ g \int_{\Omega} (\nabla \psi, \nabla v) \, dx - \int_{\Omega} f v \, dx, & \text{si } u = 0. \end{cases} \quad (4.7)$$

Para cualquier $v \in W_0^{1,p}(\Omega)$.

4.2. Discretización del problema por elementos finitos

El siguiente paso para la resolución numérica de (4.1) será discretizar este problema de optimización mediante el método de elementos finitos (FEM) y, de esta manera, obtener un problema de optimización que aproxima a (4.1).

Recordemos el problema de minimización que se busca resolver

$$\min_{u \in W_0^{1,p}} J(u) = \frac{1}{p} \int_{\Omega} |\nabla u|^p \, dx + g \int_{\Omega} |\nabla u| \, dx - \int_{\Omega} f u \, dx.$$

Siguiendo las ideas propuestas por González-Andrade [13], vamos a discretizar este problema usando una aproximación de elementos finitos de primer orden.

Consideremos $\Omega \subset \mathbb{R}^2$ no vacío, abierto, acotado y suficientemente regular. Sea T^h una triangulación regular de Ω en el sentido de Ciarlet (ver [8, pág. 38]). Seguidamente, sea Ω^h una aproximación polinomial de Ω dada por

$$\Omega^h = \bigcup_{\tau \in T^h} \bar{\tau},$$

donde τ son los triángulos que son disjuntos uno a uno y cuyo diámetro está acotado por h . Más aún, para cualquier par de triángulos, sus clausuras son disjuntas o bien tienen un vértice en común o una arista en común. Finalmente, sea $\{P_j\}_{j=1}^N$ los vértices (nodos) asociados a la triangulación T^h . De aquí en adelante asumimos que $P_j \in \partial\Omega^h$ implica que $P_j \in \partial\Omega$ y que $\Omega^h \subset \Omega$.

En el siguiente gráfico es posible observar una discretización por elementos finitos de la bola unitaria en \mathbb{R}^2 , se observa además que los triángulos y los nodos en la frontera verifican las propiedades de construcción antes mencionadas.

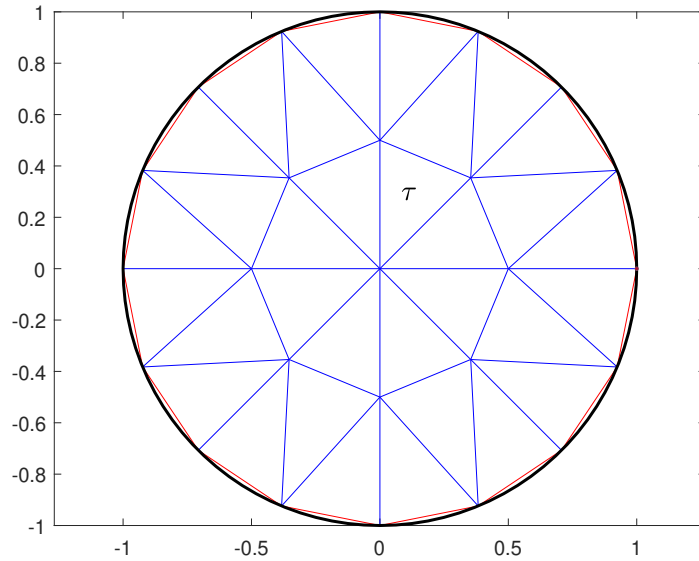


Figura 4.1: Triangulación regular de $B_1(0,0) \subset \mathbb{R}^2$.

Tomando todo esto en cuenta, introducimos el siguiente espacio finito dimensional asociado a la triangulación T^h ,

$$W_0^h := \{v \in C(\bar{\Omega}^h) : v|_{\tau} \in \mathbb{P}_1, \forall \tau \in T^h \text{ y } v = 0 \text{ en } \partial\Omega^h\},$$

donde \mathbb{P}_1 es el espacio de polinomios con grado menor o igual que 1.

Junto con estas consideraciones realizadas es posible definir la siguiente aproxima-

ción del problema (4.1) como sigue

$$\min_{u^h \in W_0^h} J_h(u^h) := \frac{1}{p} \int_{\Omega^h} |\nabla u^h|^p dx + g \int_{\Omega^h} |\nabla u^h| dx - \int_{\Omega^h} f u^h dx. \quad (4.8)$$

En consecuencia se tiene la siguiente Proposición

Proposición 4.2.1. *El funcional J_h es continuo, convexo y coercivo en W_0^h . Además, el problema (4.8) tiene solución $\bar{u}^h \in W_0^h$ y es única.*

Demostración. Es una consecuencia directa del hecho que W_0^h es un subespacio cerrado de $W_0^{1,p}(\Omega)$ (ver [12, sec. 3.2]), y hemos mostrado todas estas propiedades en la Proposición 4.1.1. \square

Ahora, sean φ_j con $j = 1, \dots, n$, las funciones que forman una base de W_0^h . Así, para cada $u^h \in W_0^h$ se tiene que

$$u^h = \sum_{j=1}^n u^j \varphi_j,$$

con $u^j \in \mathbb{R}$ para cada $j = 1, \dots, n$.

Notación 4.2.1. Notaremos por \vec{u} a el vector que guarda los coeficientes asociados a las funciones aproximadas u^h , es decir, $\vec{u} = (u^1, \dots, u^n)^T$.

Suponiendo que $\text{card}(T^h) = m$ y siguiendo las ideas de discretización planteadas en [13], usamos la siguiente versión discreta del gradiente

$$\nabla^h := \begin{pmatrix} \partial_1^h \\ \partial_2^h \end{pmatrix} \in \mathbb{R}^{2m \times n}, \quad (4.9)$$

donde

$$\partial_1^h := \left. \frac{\partial \varphi_j(x)}{\partial x_1} \right|_{\tau_k} \quad \text{y} \quad \partial_2^h := \left. \frac{\partial \varphi_j(x)}{\partial x_2} \right|_{\tau_k}$$

para $j = 1, \dots, n$ y $\tau_k \in T^h$.

Notemos que $\left. \frac{\partial \varphi_j(x)}{\partial x_1} \right|_{\tau_k}$ y $\left. \frac{\partial \varphi_j(x)}{\partial x_2} \right|_{\tau_k}$ son valores constantes de $\frac{\partial \varphi_j}{\partial x_1}$ y $\frac{\partial \varphi_j}{\partial x_2}$ sobre cada triángulo τ_k , respectivamente. Esto indica que $\nabla^h \vec{u}$ es la aproximación de $\nabla u^h(x)$.

Ahora, definimos la función $\eta: \mathbb{R}^{2m} \rightarrow \mathbb{R}^m$ dada por

$$\eta(w)_k = |(w_k, w_{k+m})|^T, \quad k = 1, \dots, m.$$

En consecuencia, calculamos $|\nabla u^h(x)|$ mediante $\eta(\nabla^h \vec{u})$. Nótese que $\eta(\nabla^h \vec{u})_k$ representa el valor de $|\nabla u^h(x)|$ en cada triángulo $\tau_k \in T^h$.

Finalmente, vamos a discutir la implementación de (4.8). En primer lugar reescribamos el funcional J_h de la siguiente manera

$$J_h(u^h) = \frac{1}{p} \int_{\Omega} |\nabla u^h|^{p-2} (\nabla u^h, \nabla u^h) dx + g \int_{\Omega} |\nabla u^h| dx - \int_{\Omega} f u^h dx.$$

Haremos uso del método de Galerkin para aproximar el primer término en J_h . Por ende, obtenemos la siguiente aproximación

$$\frac{1}{p} \int_{\Omega} |\nabla u^h|^{p-2} (\nabla u^h, \nabla u^h) dx \approx \frac{1}{p} \sum_{i,j=1}^n u^i u^j \sum_{\tau_k \in T^h} \int_{\tau_k} (\eta(\nabla^h \vec{u})_k)^{p-2} (\nabla \varphi_i, \nabla \varphi_j) dx.$$

Nótese que $(\eta(\nabla^h \vec{u})_k)^{p-2}$ es constante en cada triángulo τ_k . Así, obtenemos la matriz $A_u^h \in \mathbb{R}^{n \times n}$ cuyas entradas están dadas por

$$(a_u)_{i,j} = \sum_{\tau_k \in T^h} (\eta(\nabla^h \vec{u})_k)^{p-2} \int_{\tau_k} (\nabla \varphi_i, \nabla \varphi_j) dx.$$

Por otro lado, aproximamos el segundo término como sigue

$$g \int_{\Omega} |\nabla u^h| dx \approx g \sum_{\tau_k \in T^h} \int_{\tau_k} \eta(\nabla^h \vec{u})_k dx =: I(u^h).$$

Finalmente, el último término es aproximado de la siguiente manera

$$\int_{\Omega} f^h \varphi_j dx \approx \sum_{\tau \in T^h} Q_{\tau}(f \varphi_j), \quad j = 1, \dots, n,$$

donde la regla de cuadratura Q_{τ} está dada por

$$Q_{\tau}(v) = \frac{1}{3} |\tau| \sum_{i=1}^3 v(a_i),$$

con a_i , $i = 1, 2, 3$, los vértices de $\tau \in T^h$.

Así obtenemos el vector $\vec{f}^h \in \mathbb{R}^n$ cuyas entradas son

$$f_j^h = \sum_{\tau \in T^h} Q_{\tau}(f \varphi_j), \quad j = 1, \dots, n.$$

Tenemos entonces la versión discreta del funcional J_h ,

$$J_h(u^h) = \frac{1}{p} \vec{u}^T A_u^h \vec{u} + I(u^h) + (\vec{f}^h)^T \vec{u}. \quad (4.10)$$

4.2.1. Discretización del subdiferencial del funcional asociado al problema de flujo

Con el fin de conectar con las ideas del método Bundle Newton necesitamos también discretizar el subdiferencial de J . Como se observó en el Ejemplo 2, consideremos $\psi \in C_c^\infty(\Omega) \cap \partial J_2(0)$. En este caso, tenemos que el subdiferencial de J_h en su forma funcional es

$$\partial J_h(u^h)v^h = \begin{cases} \int_{\Omega} |\nabla u^h|^{p-2} (\nabla u^h, \nabla v^h) dx + g \int_{\Omega} \frac{(\nabla u^h, \nabla v^h)}{|\nabla u^h|} dx - \int_{\Omega} f v^h dx & \text{si } u^h \neq 0, \\ g \int_{\Omega} (\nabla \psi, \nabla v^h) dx - \int_{\Omega} f v^h dx & \text{si } u^h = 0, \end{cases}$$

para todo $v^h \in W_0^h$.

Nuevamente haremos uso del método de Galerkin para aproximar los términos que aparecen en $\partial J_h(u^h)$. Analicemos por casos:

(I) Si $u^h \neq 0$, vamos a aproximar $\int_{\Omega} \frac{(\nabla u^h, \nabla v^h)}{|\nabla u^h|} dx$. Se tiene entonces que

$$\int_{\Omega} \frac{(\nabla u^h, \nabla v^h)}{|\nabla u^h|} dx \approx \sum_{i,j=1}^n u^i v^j \sum_{\tau_k \in T^h} \int_{\tau_k} (\eta(\nabla^h \vec{u})_k)^{-1} (\nabla \varphi_i, \nabla \varphi_j) dx.$$

De igual manera, $(\eta(\nabla^h \vec{u})_k)^{-1}$ es constante sobre cada triángulo τ_k . Así, se obtiene la matriz $A_{u,-1}^h \in \mathbb{R}^{n \times n}$ cuyos coeficientes están dados por

$$(a_{u,-1})_{i,j} = \sum_{\tau_k \in T^h} (\eta(\nabla^h \vec{u})_k)^{-1} \int_{\tau_k} (\nabla \varphi_i, \nabla \varphi_j) dx.$$

(II) Si $u^h = 0$, el término por aproximar es $\int_{\Omega} (\nabla \psi, \nabla v^h) dx$. Se tiene lo siguiente

$$\int_{\Omega} (\nabla \psi, \nabla v^h) dx \approx \sum_{i,j=1}^n \psi^i v^j \sum_{\tau_k \in T^h} \int_{\tau_k} (\nabla \varphi_i, \nabla \varphi_j) dx.$$

En consecuencia, se obtiene la matriz de rigidez $A \in \mathbb{R}^{n \times n}$ cuyas entradas son

$$a_{i,j} = \sum_{\tau_k \in T^h} \int_{\tau_k} (\nabla \varphi_i, \nabla \varphi_j) dx.$$

Observe que los términos restantes que forman parte de $\partial J_h(u^h)v^h$ son similares a los que aparecen en la discretización del funcional J_h . Por consiguiente, aproximamos estos términos de la misma manera para obtener la siguiente versión discreta del

subdiferencial de J_h .

$$\partial J_h(u^h)v^h = \begin{cases} \bar{u}^T A_u^h \bar{v} + \bar{u}^T A_{u,-1}^h \bar{v} - (\bar{f}^h)^T \bar{v}, & \text{si } u^h \neq 0, \\ \bar{\psi}^T A \bar{v} - (\bar{f}^h)^T \bar{v}, & \text{si } u^h = 0. \end{cases} \quad (4.11)$$

Como consecuencia, tenemos listos todos los ingredientes necesarios para proponer un método que resuelva numéricamente el siguiente problema de optimización

$$\min_{u^h \in W_0^h} J_h(u^h),$$

con J_h definido en (4.8).

Este tópico será nuestro objetivo en la siguiente sección.

4.3. Algoritmo tipo Bundle no diferenciable

En la presente sección construimos el método de optimización que resuelve numéricamente el problema de minimización aproximado. El método propuesto tiene su fundamento en el algoritmo tipo Bundle Quasi-Newton con actualización BFGS que ha sido descrito en el Capítulo 3, sección 3.2.

Considere el siguiente problema de optimización

$$\min_{u^h \in W_0^h} J_h(u^h), \quad (4.12)$$

con J_h definido como en (4.8), el funcional aproximado a J . Gracias a la Proposición 4.2.1 conocemos que J_h es un funcional continuo y estrictamente convexo. Y además, por la ecuación (4.11) conocemos explícitamente su subdiferencial en cada uno de sus puntos.

Observación 4.3.1. Puesto que se ha obtenido la versión discreta de J_h (ecuación (4.10)) es posible notar que el problema (4.12) puede ser transformado en un problema de minimización finito dimensional sobre \mathbb{R}^n . Usaremos esta discretización de J_h y, de aquí en adelante, nos enfocaremos en la resolución numérica de este funcional. Además, sin causar ninguna ambigüedad nos desharemos de la notación vectorial “ \rightarrow ” por simplicidad de escritura.

4.3.1. Regularización de Moreau-Yosida del funcional discretizado

Como se ha observado en la sección 3.2, el punto de partida para trabajar con un método Bundle Quasi-Newton es hacer uso de la regularización de Moreau-Yosida.

Sea $M \in \mathbb{R}^{n \times n}$ una matriz simétrica definida positiva. La regularización de Moreau-Yosida de J_h asociada a M está dada por

$$F_M(u) := \min_{v \in \mathbb{R}^n} \left\{ J_h(v) + \frac{1}{2} |v - u|_M^2 \right\}, \quad (4.13)$$

donde $|\cdot|_M$ denota la norma en \mathbb{R}^n inducida por M (ver Notación 3.2.1).

Notación 4.3.1. Continuaremos denotando por $p_M(u)$ al punto proximal de u , el cual es la única solución del problema de minimización en (4.13) (ver Observación 3.2.1).

Gracias a la Proposición 3.2.1 se sigue que F_M es una función convexa, continuamente diferenciable y definida en todo \mathbb{R}^n . Además, la derivada de F_M en el punto u está dada por

$$G_M(u) := \nabla F_M(u) = M(u - p_M(u)) \in \partial J_h(p_M(u)). \quad (4.14)$$

En adición, G_M es globalmente Lipschitz continuo con módulo $\|M\|_{\mathbb{R}^n \times \mathbb{R}^n}$.

Recordemos que minimizar F_M es equivalente a minimizar J_h , es decir,

$$\bar{u} = \arg \min_{u \in \mathbb{R}^n} J_h(u) \Leftrightarrow \bar{u} = \arg \min_{u \in \mathbb{R}^n} F_M(u).$$

Así, por el Teorema 2.4.2, \bar{u} minimiza J_h si y sólo si $G_M(\bar{u}) = 0$ y $p_M(\bar{u}) = \bar{u}$.

El siguiente paso es determinar la manera a resolver el problema de minimización auxiliar en (4.13) para calcular F_M y G_M . Como se ha observado en el trabajo de Mifflin, Sun y Qi [22], incorporaremos las ideas del método Bundle clásico para aproximar F_M y G_M .

4.3.2. Método Bundle para la resolución del problema auxiliar

Seguiremos exactamente el mismo procedimiento detallado en la Subsección 3.2.2. para la aproximación de F_M y G_M .

Sea $u \in \mathbb{R}^n$ cualquiera, y sea $w = v - u$ en (4.13). Minimizando sobre w en lugar de v se tiene que

$$F_M(u) = \min_{w \in \mathbb{R}^n} \left\{ J_h(u + w) + \frac{1}{2} w^T M w \right\}, \quad (4.15)$$

y $p_M(u) = u + \bar{w}$, donde $\bar{w} = \arg \min_{w \in \mathbb{R}^n} \{J_h(u + w) + \frac{1}{2}w^T M w\}$. Dado que J_h es convexo, se aproxima $J_h(u + w)$ mediante el siguiente modelo de plano cortante

$$\check{J}_h(u + w) = \max_{j=1, \dots, l} \{J_h(v^j) + (\xi^j)^T(u + w - v^j)\},$$

donde las tripletas $(v^j, J_h(v^j), \xi^j)$ con $\xi^j = \xi(v^j) \in \partial J_h(v^j)$, constituyen un bundle \mathcal{B} que ha sido generado secuencialmente partiendo desde u y $\xi(u) \in \partial J_h(u)$, y posiblemente, algún subconjunto del bundle anterior usado para generar u .

Proposición 4.3.1. *Puesto que J_h es convexo (estricto) se tiene que*

$$J_h(u + w) \geq \check{J}_h(u + w). \quad (4.16)$$

Demostración. Sea $m \in \{1, \dots, l\}$ tal que

$$\begin{aligned} J_h(v^m) + (\xi^m)^T(u + w - v^m) &= \max_{j=1, \dots, l} \{J_h(v^j) + (\xi^j)^T(u + w - v^j)\} \\ &= \check{J}_h(u + w). \end{aligned}$$

Puesto que $\xi^m \in \partial J_h(v^m)$, entonces satisface lo siguiente

$$J_h(z) \geq J_h(v^m) + (\xi^m)^T(z - v^m), \quad \forall z \in \mathbb{R}^n.$$

Tomando $z = u + w$ se sigue que

$$\begin{aligned} J_h(u + w) &\geq J_h(v^m) + (\xi^m)^T(u + w - v^m) \\ &= \check{J}_h(u + w). \end{aligned}$$

Lo que muestra el resultado deseado. □

Análogamente, se define el error de linearización como

$$e(u, v^j) = J_h(u) - J_h(v^j) - (\xi^j)^T(u - v^j),$$

entonces $\check{J}_h(u + w)$ puede ser escrito como

$$\check{J}_h(u + w) = J_h(u) + \max_{j=1, \dots, l} \{(\xi^j)^T w - e(u, v^j)\}. \quad (4.17)$$

Así, se define \check{F}_M de la siguiente manera

$$\begin{aligned} \check{F}_M(u) &:= \min_{w \in \mathbb{R}^n} \left\{ \check{J}_h(u + w) + \frac{1}{2}w^T M w \right\} \\ &= J_h(u) + \min_{w \in \mathbb{R}^n} \left\{ \max_{j=1, \dots, l} \{(\xi^j)^T w - e(u, v^j)\} + \frac{1}{2}w^T M w \right\}. \end{aligned} \quad (4.18)$$

En consecuencia, gracias (4.16) y la definición de F_M se tiene que

$$\check{F}_M(u) \leq F_M(u).$$

Por consiguiente, $\check{F}_M(u)$ es también una subaproximación del valor aún desconocido $F_M(u)$.

Ahora bien, transformamos el problema auxiliar en (4.18) en un problema de optimización cuadrática. Sea $\nu = \max_{j=1, \dots, l} \{(\xi^j)^T w - e(u, v^j)\}$, el siguiente problema de minimización es equivalente al problema auxiliar de (4.18)

$$\begin{aligned} \min_{(w, \nu)} \quad & \nu + \frac{1}{2} w^T M w \\ \text{s.a.} \quad & -e(u, v^j) + (\xi^j)^T w \leq \nu, \quad j = 1, \dots, l. \end{aligned} \tag{4.19}$$

Las condiciones de optimalidad del problema (4.19) implican la existencia de multiplicadores no negativos $\lambda_j(u)$ que suman uno y verifican

$$\begin{aligned} M w(u) &= - \sum_{j=1}^l \lambda_j(u) \xi^j, \\ \nu(u) &= (\xi^j)^T w(u) - e(u, v^j), \quad \text{si } \lambda_j(u) > 0, \end{aligned}$$

con $(w(u), \nu(u))$ la única solución de (3.17) (ver [21, sec. 5]).

Sea $(w(u), \nu(u))$ la solución de (4.19), entonces

$$\check{F}_M(u) = J_h(u) + \nu(u) + \frac{1}{2} (w(u))^T M w(u).$$

Por otro lado, sea $a(u) = u + w(u)$ una aproximación de $p_M(u)$ y sea

$$\hat{F}_M(u) = J_h(a(u)) + \frac{1}{2} (w(u))^T M w(u),$$

puesto que $p_M(u)$ es el mínimo de (4.13) y es único, entonces

$$F_M(u) \leq \hat{F}_M(u),$$

y la igualdad se consigue si y sólo si $a(u) = p_M(u)$.

Por tanto tenemos las mismas deducciones que en el lema 3.2.1 que se enlistan a continuación.

Lema 4.3.1.

$$(i) \quad \check{F}_M(u) \leq F_M(u) \leq \hat{F}_M(u),$$

(II) $F_M(u) = \hat{F}_M(u)$ si y sólo si $a(u) = p_M(u)$.

Como se ha notado, gracias a este lema se puede imponer una condición de parada para el subproceso Bundle. Sea

$$\varepsilon(u) = \hat{F}_M(u) - \check{F}_M(u). \quad (4.20)$$

Se acepta que $a(u)$ es una aproximación de $p_M(u)$ si se cumple que

$$\varepsilon(u) \leq \delta(u) \text{ mín } \{ (w(u))^T M w(u), N \}, \quad (4.21)$$

donde $\delta(u)$ y N son números positivos dados, y $\delta(u)$ permanece fijo durante el subproceso Bundle. Si (4.21) no se verifica, entonces sean $u^{l+1} = u + d(u)$ y $\xi^{l+1} \in \partial J_h(u^{l+1})$, agregue el término $(\xi^{l+1})^T d - e(x, u^{l+1})$ a (4.17). Reemplace $l = l + 1$, actualice el bundle $\mathcal{B} = \mathcal{B} \cup \{ (v^{l+1}, J_h(v^{l+1}), \xi^{l+1}) \}$. Entonces, resuelva el nuevo subproblema en (4.19) que determina un nuevo $w(u)$ y un nuevo $\varepsilon(u)$ a ser probados en (4.21).

Observamos que hasta ahora nos ha sido posible seguir las ideas del subproceso Bundle descrito en el capítulo anterior para la aproximación de F_M . Por consiguiente obtendremos dicho valor empleando el algoritmo 5, el cual calcula una aproximación de $p_M(u)$.

Ahora, denotaremos por $\tilde{G}_M(u)$ a

$$\tilde{G}_M(u) = M(u - a(u)) = -Md(u).$$

Tomando esto en cuenta sabemos que por el siguiente lema nos es posible aproximar $G_M(u)$ gracias al subproceso Bundle y su demostración se sigue análogamente a la realizada en [22].

Lema 4.3.2.

$$\begin{aligned} |G_M(u) - \tilde{G}_M(u)|_{M^{-1}} &= |p(u) - a(u)|_M \leq \sqrt{2\varepsilon(u)}, \\ |G_M(u) - \tilde{G}_M(u)| &\leq \sqrt{2\varepsilon(u)} \|M\|. \end{aligned}$$

En adición, se presenta otro lema que establece que el subproceso Bundle que estamos siguiendo termina en número finito de pasos, de igual manera su demostración puede ser estudiada en [22].

Lema 4.3.3. *Si u no minimiza J_h . Entonces, después de un número finito de pasos del subalgoritmo Bundle, es posible encontrar una solución $w(u)$ tal que (4.21) se verifica.*

Observación 4.3.2. Gracias al Lema 4.3.3 es posible usar el subproceso Bundle (Al-

goritmo 5) para hallar un vector $w(u)$ tal que (4.21) se verifica si u no es la solución óptima. Esto resulta esencial para el algoritmo Bundle no diferenciable, pues, es posible aproximar $F_M(u)$ y $\nabla F_M(u)$, para cualquier valor de u .

4.3.3. Actualización de tipo Newton para el problema regularizado

Para la actualización de tipo Quasi-Newton se propone realizar una actualización BFGS si ciertas condiciones se verifican, caso contrario, se tomará la matriz M para realizar la actualización tipo Newton y obtener una nueva dirección de descenso.

Dados los vectores Δu y Δy . La actualización BFGS de una matriz simétrica $B \in \mathbb{R}^{n \times n}$ sigue la siguiente ecuación

$$BFGS(B, \Delta u, \Delta y) := B + \frac{\Delta u \Delta y^T}{\Delta u^T \Delta y} - \frac{B \Delta u \Delta u^T B}{\Delta u^T B \Delta u}.$$

Si B es definida positiva y además $\Delta u^T \Delta y > 0$ entonces la matriz simétrica $B^+ := BFGS(B, \Delta u, \Delta y)$ es también definida positiva (ver [24, pág. 31]).

En nuestro método asumiremos que $B_0 = M$ y además $\sum_{k=1}^{\infty} \delta_k^{1/3} < \infty$. Durante cada iteración, si las siguientes dos condiciones se verifican, se actualizará B_k a $B_{k+1} = BFGS(B_k, \Delta u^k, \Delta y^k)$; caso contrario, se tomará $B_{k+1} := M$. Dados $\alpha_1 \in (0, +\infty)$ y $\alpha_2 \in (0, 1)$ estas dos condiciones son

$$|\Delta u^k|_M (\sqrt{2\varepsilon_k} + \sqrt{2\varepsilon_{k+1}}) \leq \alpha_1 (\Delta u^k) \Delta y^k, \quad (4.22)$$

y

$$2|\Delta y^k|_M (\sqrt{2\varepsilon_k} + \sqrt{2\varepsilon_{k+1}}) \leq \min\{\alpha_2, \delta_k^{1/3} + \delta_{k+1}^{1/3}\} |\Delta y^k|^2. \quad (4.23)$$

Con $\varepsilon_k = \varepsilon(u^k)$.

Como bien señalan Mifflin, Sun y Qi en [22, teo. 5.5], si se supone que F_M es fuertemente convexa, entonces, esta actualización permite que el método converja R -linealmente hacia el mínimo.

4.3.4. Búsqueda lineal de la longitud de paso

Como siguiente tema para la construcción del algoritmo nos enfocaremos en la implementación de un método de búsqueda lineal inexacta. Emplearemos el algoritmo de búsqueda lineal “backtracking” desarrollado por Dennis y Schnabel en [9, sección. 6.3.2].

Sea u^k la iteración actual del método y s^k la dirección de descenso. Dado $\sigma \in (0, 1)$, buscaremos $\lambda_k > 0$ que verifique la siguiente condición tipo Wolfe-Powell

$$F_M(u^k + \lambda_k s^k) \leq F_M(u^k) + \sigma \lambda_k (s^k)^T G_M(u^k). \quad (4.24)$$

La estrategia del algoritmo backtracking es tomar inicialmente $\lambda_k = 1$. Luego, si $u^k + s^k$ no es admisible ((4.24) no se satisface), entonces se reduce λ_k hasta que $u^k + \lambda_k s^k$ sea aceptable.

Como menciona González-Andrade en [13], el punto central en un algoritmo backtracking es cómo disminuir λ_k . Usualmente, este método es implementado tomando $\lambda_k = \frac{1}{2^k} \lambda_k$, así, λ_k es aminorado a la mitad en cada iteración. Sin embargo, este procedimiento puede resultar ineficiente debido a que se requiere de numerosas iteraciones para lograr convergencia, y en adición, los valores de λ_k pueden ser bastante pequeños.

Siguiendo las ideas presentadas en [9, sección. 6.3.2], proponemos una estrategia para la reducción de λ_k usando modelos polinomiales de F_M .

Iniciemos introduciendo la siguiente función

$$\varphi_k(\lambda) := F_M(u^k + \lambda s^k),$$

la restricción de F_M a la recta que pasa por u^k en la dirección s^k . Seguidamente, usando la información disponible de F_M , se toma λ_k como la aproximación del valor que minimiza $\varphi_k(\lambda)$, es decir, $\lambda_k \approx \arg \min \varphi_k(\lambda)$.

En primer lugar, nótese que la siguiente información sobre φ_k está disponible

$$\varphi_k(0) = F_M(u^k) \quad \text{y} \quad \varphi_k'(0) = (s^k)^T G_M(u^k). \quad (4.25)$$

Una vez que se ha calculado $F_M(u^k + s^k)$, también se tiene que

$$\varphi_k(1) = F_M(u^k + s^k). \quad (4.26)$$

En consecuencia, si $F_M(u^k + s^k)$ no satisface (4.24), (i.e. $\varphi_k(1) > \varphi_k(0) + \sigma \varphi_k'(0)$), construimos el siguiente modelo cuadrático de φ_k usando (4.25) y (4.26)

$$m_2(\lambda) := (\varphi_k(1) - \varphi_k(0) - \varphi_k'(0)) \lambda^2 + \varphi_k'(0) \lambda + \varphi_k(0).$$

No es difícil mostrar que

$$\tilde{\lambda}_2 = \frac{-\varphi_k'(0)}{2(\varphi_k(1) - \varphi_k(0) - \varphi_k'(0))}, \quad (4.27)$$

es un punto estacionario de m_2 , en consecuencia, $m_2'(\tilde{\lambda}_2) = 0$. Y en adición, se tiene

que

$$m_2''(\lambda) = 2(\varphi_k(1) - \varphi_k(0) - \varphi_k'(0)) > 0,$$

pues $\varphi_k(1) > \varphi_k(0) + \sigma\varphi_k'(0) > \varphi_k(0) + \varphi_k'(0)$. En consecuencia, $\tilde{\lambda}_2$ minimiza $m_2(\lambda)$. Además, dado que $\varphi_k'(0) < 0$ entonces $\tilde{\lambda}_2 > 0$. Por consiguiente se toma $\lambda_k := \tilde{\lambda}_2$.

Puesto que $\varphi_k(1) > \varphi_k(0) + \sigma\varphi_k'(0)$, de (4.26) se deduce que

$$\tilde{\lambda}_2 < \frac{1}{2(1-\sigma)}.$$

Así, si sucede que $\varphi_k(1) \geq \varphi_k(0)$, entonces $\tilde{\lambda}_2 \leq \frac{1}{2}$. Por tanto, (4.26) brinda una cota superior de aproximadamente $\frac{1}{2}$ en la primera reducción de λ_k .

Por otra parte, si $\varphi_k(1) \gg \varphi_k(0)$ entonces λ_k puede ser muy pequeño. Esto sugiere que $\varphi_k(\lambda)$ no está suficientemente bien modelada por un modelo cuadrático en esta región. En consecuencia, para evitar longitudes de paso muy pequeñas imponemos una cota inferior de $\frac{1}{10}$. Esto es, si en la primera reducción de cada iteración se tiene que $\tilde{\lambda}_2 \leq 0.1$, entonces el algoritmo intenta con $\lambda_k = \frac{1}{10}$.

Ahora, suponiendo que $\varphi_k(\tilde{\lambda}_2)$ no verifica (4.24), entonces λ_k debe reducirse nuevamente. En este caso, se tiene la siguiente información disponible: $\varphi_k(0)$, $\varphi_k'(0)$ y los dos últimos valores de $\varphi_k(\lambda)$. De aquí y en todas las reducciones siguientes usaremos un modelo cúbico de φ_k que usa estos cuatro valores disponibles, y tomamos λ_k como el mínimo de este modelo.

Este procedimiento se sostiene en la razón de que un modelo cúbico puede ser más preciso al modelar situaciones donde la curvatura de F_M es negativa, que habitualmente sucede cuando (4.24) no se verifica para dos valores positivos de λ . (Ver [9, pág. 128]).

Para calcular λ_k se procede de la siguiente manera, sea λ_p and λ_{2p} los dos valores anteriores de λ_k . El modelo cúbico que usa $\varphi_k(0)$, $\varphi_k'(0)$, $\varphi_k(\lambda_p)$, y $\varphi_k(\lambda_{2p})$ es

$$m_3(\lambda) = c\lambda^3 + d\lambda^2 + \varphi_k'(0)\lambda + \varphi_k(0),$$

donde

$$\begin{pmatrix} c \\ d \end{pmatrix} = \frac{1}{\lambda_p - \lambda_{2p}} \begin{pmatrix} \frac{1}{\lambda_p^2} & -\frac{1}{\lambda_{2p}^2} \\ -\frac{\lambda_{2p}}{\lambda_p^2} & \frac{\lambda_p}{\lambda_{2p}^2} \end{pmatrix} \begin{pmatrix} \varphi_k(\lambda_p) - \varphi_k(0) - \varphi_k'(0)\lambda_p \\ \varphi_k(\lambda_{2p}) - \varphi_k(0) - \varphi_k'(0)\lambda_{2p} \end{pmatrix}.$$

Además, el mínimo de m_3 está dado por

$$\tilde{\lambda}_3 = \frac{-d + \sqrt{d^2 - 3c\varphi_k'(0)}}{3c}. \quad (4.28)$$

En [9, pág. 128] los autores mencionan que si $\varphi_k(\lambda_p) \geq \varphi_k(0)$, entonces $\tilde{\lambda}_3 < \frac{2}{3}\lambda_p$. Sin embargo, esta reducción se considera bastante pequeña. Como resultado, nuevamente impondremos una cota superior de $1/2$, lo que implica que si $\tilde{\lambda}_3 > \frac{1}{2}\lambda_p$, tomamos $\lambda_k = \frac{1}{2}\lambda_p$. De igual manera, puesto que $\tilde{\lambda}_3$ puede resultar una pequeña fracción de λ_p , exigimos una cota inferior de $1/10$, es decir, si $\tilde{\lambda}_3 < \frac{1}{10}\alpha_p$, escogemos $\lambda_k = \frac{1}{10}\alpha_p$.

Resumiendo, se tiene el siguiente algoritmo:

Algoritmo 7: Algoritmo de búsqueda lineal backtracking

Datos: Dado $\sigma \in (0, 1/2)$ y $\lambda_0 = 1$.

Paso 1. Determine si (4.24) se verifica. Si es así **PARAR** y tomar $\lambda_k = \lambda_0$. De otro modo:

Paso 2. Decidir si la longitud de paso es demasiado pequeña. De ser el caso

PARAR y terminar el algoritmo: el programa ha fallado en localizar u^{k+1} suficientemente distinto de u^k . Caso contrario:

Paso 3. Reducir λ por un factor entre 0.1 y 0.5 de la siguiente manera:

- (a) En la primera reducción: tomar $\lambda_k := \tilde{\lambda}_2 = \arg \min m_2(\lambda)$, pero restringir el nuevo valor de λ_k a ser mayor que 0.1.
- (b) En las subsiguientes reducciones: tomar $\lambda_k := \tilde{\lambda}_3 = \arg \min m_3(\lambda)$, pero restringir el nuevo valor de λ_k a que pertenezca al intervalo $[0.1\lambda_p, 0.5\lambda_p]$.

Paso 4. Regresar al paso 1.

Usualmente, el parámetro σ se lo toma muy pequeño, como de un orden de 10^{-4} . Más aún, (4.28) no es imaginario si $\sigma < 1/4$ (ver [9, pág. 129]).

Observación 4.3.3. Nótese que en este método de búsqueda lineal hemos trabajado teóricamente con los valores exactos de F_M y G_M . Ahora por lo observado en la subsección 4.3.2 estos dos valores son aproximados a través del subalgoritmo Bundle. Por esta razón, implementaremos el algoritmo 7 haciendo uso de \check{F}_M y \check{G}_M las aproximaciones de F_M y G_M , respectivamente. Estos resultan ser valores disponibles a lo largo del método de minimización.

Finalmente, apoyándonos en el Teorema 3.2.1 se tiene el siguiente criterio de parada para el algoritmo en general: El método se detiene si el subalgoritmo Bundle genera $\nu(u)$ tal que verifica

$$|\nu(u)| \leq tol,$$

donde tol es un parámetro de precisión positivo y suficientemente pequeño.

Para finalizar esta sección, resumimos todo lo abordado en el siguiente esquema del método tipo Bundle no diferenciable con aplicación al problema de flujo.

Algoritmo 8: Algoritmo tipo Bundle no diferenciable aplicado al problema de flujo

Datos: Sean σ , N y tol números positivos tales que $\sigma < 1/2$ y $tol \lll 1$. Sea $\{\delta_k\}$ una sucesión de números positivos tales que $\sum_{k=0}^{\infty} \delta_k^{1/3} < \infty$. Sea $u^0 \in \mathbb{R}^n$ un candidato inicial, $M \in \mathbb{R}^{n \times n}$ simétrica definida positiva y sea \mathcal{B} un bundle.

Paso 0. (Inicialización) Fijar $k := 0$ y $B_0 := M$, hallar (w^0, ν^0) y ε_0 mediante el subalgoritmo bundle (algoritmo 5) tales que

$$\varepsilon_0 \leq \delta^0 \{(w^0)^T M w^0, N\}.$$

Por ejemplo, inicializando el bundle $\mathcal{B} = \{(u^0, J_h^0, \xi^0)\}$.

Paso 1. (Calcular la dirección de búsqueda) Si $|\nu^k| < tol$ parar con u^k optimal. Caso contrario, calcule

$$s^k = -B_k^{-1} \tilde{G}_M(u^k). \quad (4.29)$$

Paso 2. (Búsqueda Lineal) Iniciando con $\lambda_k = 1$, determinar si la siguiente condición se verifica

$$\tilde{F}_M(u^k + \lambda_k s^k) \leq \tilde{F}_M(u^k) + \sigma \lambda_k (s^k)^T \tilde{G}_M(u^k), \quad (4.30)$$

donde $\tilde{F}_M(u^k + \lambda_k s^k)$ es la subaproximación de F_M en $u^k + \lambda_k s^k$ y cumple que

$$\begin{aligned} & \hat{F}_M(u^k + \lambda_k s^k) - \tilde{F}_M(u^k + \lambda_k s^k) \\ & \leq \delta_{k+1} \min \left\{ (w(u^k + \lambda_k s^k))^T M w(u^k + \lambda_k s^k), N \right\}. \end{aligned} \quad (4.31)$$

Caso contrario, reduzca λ_k mediante el algoritmo de búsqueda lineal (algoritmo 7) hasta que (4.30) se satisfaga. Fijar $u^{k+1} = u^k + \lambda_k s^k$.

Paso 3. (Actualización de la matriz Quasi-Newton) Sea $\Delta u^k = u^{k+1} - u^k$ y $\Delta y^k = \tilde{G}_M(u^{k+1}) - \tilde{G}_M(u^k)$. Si (4.22) y (4.23) se satisfacen, actualizar B_k a B_{k+1} usando la actualización BFGS

$$B_{k+1} = BFGS(B_k, \Delta u^k, \Delta y^k).$$

Caso contrario, tomar $B_{k+1} = M$. Actualizar $k := k + 1$ y regresar al paso 1.

4.4. Análisis del algoritmo

Para concluir este capítulo estudiaremos las propiedades que presenta el Algoritmo 8, mostraremos que el mismo converge hacia el mínimo y además examinaremos el orden de convergencia hacia la solución.

Durante esta sección asumiremos que \bar{u} , la solución de (4.12), es distinta de 0. Esta suposición resulta razonable puesto que una velocidad de deformación nula no brindaría mucha información sobre el comportamiento del fluido en una tubería.

4.4.1. Convergencia del método hacia al mínimo

Mostraremos que el Algoritmo 8 converge globalmente hacia el mínimo de J_h . De aquí en adelante realizaremos la siguiente suposición sobre la sucesión generada por el método.

Hipótesis 2. *Asumiremos que el método propuesto no termina. En consecuencia, $\{u^k\}$ es una sucesión infinita.*

Puesto que $\sum_{k=0}^{\infty} \delta_k < \infty$. Entonces, existe $C > 0$ tal que

$$\sum_{k=0}^{\infty} \delta_k \leq C. \quad (4.32)$$

Ahora, sea D el siguiente conjunto de nivel definido por

$$D := \{u \in \mathbb{R}^N : F_M(u) \leq F_M(u^0) + NC\}.$$

El siguiente lema nos indica que todo elemento de la sucesión generada por el Algoritmo 8 pertenece a D y además nos brindará una estimación fundamental para conseguir la convergencia global del método.

Lema 4.4.1. *Para todo $k \geq 0$ se satisface que $u^k \in D$, y además*

$$F_M(u^{k+1}) \leq F_M(u^k) + N\delta_{k+1}. \quad (4.33)$$

Demostración. Sea $k \geq 0$ arbitrario pero fijo. Por el Lema 4.3.1 y las reglas de búsqueda

lineal del algoritmo (ecuaciones (4.30) y (4.31)) se tiene que

$$\begin{aligned}
F_M(u^{k+1}) &\leq \hat{F}_M(u^{k+1}), \\
&\leq \check{F}_M(u^{k+1}) + N\delta_{k+1}, \\
&\leq \check{F}_M(u^k) + \sigma\lambda_k(s^k)^T \tilde{G}_M(u^k) + N\delta_{k+1}, \\
&= \check{F}_M(u^k) - \sigma\lambda_k \tilde{G}_M(u^k)^T B_K^{-1} \tilde{G}_M(u^k) + N\delta_{k+1}.
\end{aligned}$$

Como B_k es definida positiva, entonces B_k^{-1} también lo es, en consecuencia

$$-\sigma\lambda_k \tilde{G}_M(u^k)^T B_K^{-1} \tilde{G}_M(u^k) < 0.$$

Tomando esto en cuenta, y nuevamente por el Lema 4.3.1 se sigue que

$$F_M(u^{k+1}) \leq F_M(u^k) + N\delta_{k+1}.$$

Puesto que k es arbitrario entonces, para todo $k \geq 0$, (4.33) se verifica y además

$$u^{k+1} \in D.$$

Finalmente, es claro que $u^0 \in D$, lo que concluye la demostración de este resultado. \square

En el siguiente teorema, bajo hipótesis de acotamiento sobre las sucesiones de las matrices $\{B_k\}$ y $\{B_k^{-1}\}$, garantizamos la convergencia global del método hacia el mínimo de J_h .

Teorema 4.4.1. *Suponga que existen dos constantes positivas c_1 y c_2 tales que $\|B_k\| \leq c_1$ y $\|B_k^{-1}\| \leq c_2$, para cada k . Entonces cualquier punto de acumulación de $\{u^k\}$ minimiza J_h .*

Demostración. Por el Lema 4.4.1 se tiene que $F_M(u^k)$ es acotada superiormente. Por otro lado, puesto que J_h es acotado inferiormente (ver Proposición 4.2.1), entonces F_M también lo es. Sea $F_M^* := \liminf_{k \rightarrow \infty} F_M(x^k)$, por (4.32) y (4.33) se tiene que

$$\lim_{k \rightarrow \infty} F_M(u^k) = F_M^*.$$

Ahora, puesto que $\delta_k \rightarrow 0$ cuando $k \rightarrow \infty$, de las reglas del algoritmo se sigue que $\varepsilon_k \rightarrow 0$ cuando $k \rightarrow \infty$, y en consecuencia

$$\lim_{k \rightarrow \infty} \check{F}_M(u^k) = \lim_{k \rightarrow \infty} \hat{F}_M(u^k) =: l.$$

Luego, por el lema 4.3.1 sabemos que

$$\check{F}_M(u^k) \leq F_M(u^k) \leq \hat{F}_M(u^k),$$

tomando límite cuando $k \rightarrow \infty$ en esta última ecuación, por el teorema del sánduche se deduce que

$$\lim_{k \rightarrow \infty} F_M(u^k) = l,$$

y por unicidad del límite $l = F_M^*$.

De donde,

$$\lim_{k \rightarrow \infty} \check{F}_M(u^k) = \lim_{k \rightarrow \infty} \hat{F}_M(u^k) = F_M^*.$$

Por otra parte, nuevamente de las reglas del algoritmo se sabe que

$$\check{F}_M(u^{k+1}) - \check{F}_M(u^k) \leq \sigma \lambda_k (s^k)^T \tilde{G}_M(u^k) < 0,$$

tomando límite cuando $k \rightarrow \infty$, y nuevamente, por el teorema del sánduche se deduce que

$$\lim_{k \rightarrow \infty} \lambda_k (s^k)^T \tilde{G}_M(u^k) = 0,$$

de donde,

$$\lim_{k \rightarrow \infty} \lambda_k |\tilde{G}_M(u^k)|^2 = 0. \quad (4.34)$$

En efecto, notemos lo siguiente

$$\begin{aligned} \lambda_k |\tilde{G}_M(u^k)|^2 &= \lambda_k \tilde{G}_M(u^k)^T \tilde{G}_M(u^k), \\ &= \lambda_k \tilde{G}_M(u^k)^T B_k B_k^{-1} \tilde{G}_M(u^k), \end{aligned}$$

y gracias al supuesto de que $\{B_k\}$ son acotadas, implica que

$$\lambda_k |\tilde{G}_M(u^k)|^2 \leq c_1 \lambda_k \tilde{G}_M(u^k)^T B_k^{-1} \tilde{G}_M(u^k) \xrightarrow{k \rightarrow \infty} 0,$$

entonces (4.34) se verifica.

Ahora, sea \bar{u} un punto de acumulación de $\{u^k\}$, así podemos tomar una subsucesión $\{u^k\}$ (denotada de la misma manera) tal que $u^k \rightarrow \bar{u}$ cuando $k \rightarrow \infty$. Por el Lema 4.3.2 se tiene que

$$0 \leq |G_M(xu^k) - \tilde{G}_M(u^k)|^2 \leq 2\|M\|\varepsilon_k,$$

puesto que $\varepsilon_k \rightarrow 0$, se sigue que

$$\lim_{k \rightarrow \infty} \tilde{G}_M(u^k) = \lim_{k \rightarrow \infty} G_M(u^k).$$

Como G_M es continuo, se deduce que

$$\lim_{k \rightarrow \infty} \tilde{G}_M(u^k) = G_M(\bar{u}). \quad (4.35)$$

Por el método de búsqueda lineal propuesto tenemos que $\liminf_{k \rightarrow \infty} \lambda_k > 0$. Así, gracias a (4.34) y (4.35) se sigue que

$$G_M(\bar{u}) = 0.$$

Entonces \bar{u} minimiza J_h . □

4.4.2. Orden de convergencia del algoritmo

Para finalizar el presente capítulo estudiamos el orden de convergencia del algoritmo Bundle no diferenciable (algoritmo 8).

Una de las hipótesis señaladas por los autores en [22] para que el método posea un orden de convergencia superlineal es requerir que F_M sea fuertemente convexa. En consecuencia de [16, teo. 2.2], F_M es fuertemente convexa si y sólo si J_h lo es. Por consiguiente, iniciaremos estudiando en la siguiente proposición el caso cuando J_h es fuertemente convexo.

Proposición 4.4.1. *Si $p \geq 2$ entonces el funcional J_h es fuertemente convexo.*

Demostración. Sea $p \geq 2$ cualquiera. Iniciemos reescribiendo J_h de la siguiente manera

$$J_h(u^h) = J_{1,h}(u^h) + J_{2,h}(u^h),$$

donde

$$J_{1,h}(u^h) = \int_{\Omega^h} |\nabla u^h|^p dx - \int_{\Omega^h} f u^h dx \quad \text{y} \quad J_{2,h}(u^h) = \int_{\Omega^h} |\nabla u^h| dx.$$

Como hemos observado en la Sección 4.1, $J_{1,h}$ es un funcional Fréchet diferenciable y su derivada está dada por

$$J'_{1,h}(u^h)v^h = \int_{\Omega^h} |\nabla u^h|^{p-2} (\nabla u^h, \nabla v^h) dx - \int_{\Omega^h} f v^h dx.$$

Ciarlet en [8, pág. 322] menciona que $J'_{1,h}$ es un operador fuertemente monótono, y en consecuencia de [10, teo. 3.4] se sigue que $J_{1,h}$ es fuertemente convexo.

Por otro lado, nótese que $J_{2,h}$ es convexo. Así, J_h es la suma de un funcional fuertemente convexo y uno convexo. Esto permite concluir que J_h es fuertemente convexo. □

Observación 4.4.1. En [7], Chiappinelli y Edmunds describen que para $1 < p < 2$ el

operador $J'_{1,h}$ no es fuertemente monótono, y nuevamente por [10, teo. 3.4] es posible deducir que J_h no es fuertemente convexo.

De aquí en adelante supondremos que $p > 2$. Consecuentemente F_M es fuertemente convexa.

Adicionalmente, otra de las hipótesis exigidas por Mifflin, Sun y Qi en [22, sec. 5], es regularidad de segundo orden de F_M en el mínimo y en una vecindad del mismo. Por consiguiente, debemos analizar la diferenciabilidad de segundo orden de F_M en \bar{u} y en una vecindad adecuada.

Para abordar este tema, nos ayudaremos del trabajo de Lemaréchal y Sagastizábal [16], en dicho artículo los autores exponen las condiciones necesarias para la existencia de $\nabla^2 F_M$. Particularmente, demuestran el siguiente resultado

Lema 4.4.2. *Sea f una función convexa, finitamente valuada y que posee Hessiana en $p_M(x)$. Entonces, la Hessiana de F_M en x existe y está dada por*

$$\nabla^2 F_M(x) = M - M[\nabla^2 f(p_M(x)) + M]^{-1}M. \quad (4.36)$$

Nótese que para $u^h \in W_0^h$ tal que $u^h \neq 0$ el funcional J_h es dos veces diferenciable y su diferencial de segundo orden está dada por

$$\begin{aligned} J_h''(u^h)(v^h, w^h) &= (p-2) \int_{\Omega} |\nabla u^h|^{p-4} (\nabla u^h, \nabla v^h) (\nabla u^h, \nabla w^h) dx \\ &\quad + \int_{\Omega} |\nabla u^h|^{p-2} (\nabla v^h, \nabla w^h) dx + g \int_{\Omega} \frac{(\nabla v^h, \nabla w^h)}{|\nabla u^h|} dx \\ &\quad - g \int_{\Omega} \frac{(\nabla u^h, \nabla v^h) (\nabla u^h, \nabla w^h)}{|\nabla u^h|^3} dx. \end{aligned}$$

Así, por medio del método de Galerkin es posible obtener J_h'' en su versión matricial. Y, la seguiremos denotando por J_h'' .

Puesto que $\bar{u} \neq 0$, entonces $J_h''(\bar{u})$ existe y está bien definida. Por tanto, gracias al Lema 4.4.2, la Hessiana de F_M en \bar{u} existe, y como $p_M(\bar{u}) = \bar{u}$, se sigue que

$$\nabla^2 F_M(\bar{u}) = M - M[J_h''(\bar{u}) + M]^{-1}M. \quad (4.37)$$

Ahora, analizaremos la regularidad de segundo orden de F_M en una vecindad de \bar{u} . Consideremos el siguiente conjunto $V := B_r(\bar{u})$, donde $r := \frac{|\bar{u}|}{\max\{2, L_p\}}$, con L_p la constante asociada a la Lipschitz continuidad del operador p_M (ver [16, prop. 3.2]). Es claro que V es una vecindad de \bar{u} , y además verifica la siguiente propiedad

Lema 4.4.3. *Para cada $u \in V$ se tiene que $p_M(u) \neq 0$.*

Demostración. Empecemos notando que $0 \notin B(\bar{u}, L_p r)$, pues $|\bar{u}| \geq L_p r$. Por otro lado, para cualquier $u \in V$ se tiene que $p_M(u) \in B(\bar{u}, L_p r)$. En efecto

$$\begin{aligned} |p_M(u) - \bar{u}| &= |p_M(u) - p_M(\bar{u})| \\ &\leq L_p |u - \bar{u}| \\ &< L_p r. \end{aligned}$$

Por tanto, se sigue el resultado deseado. \square

Como consecuencia de este lema se tiene que para cada $u \in V$, $J_h''(p_M(u))$ existe y está bien definida. Ahora, en vista del Lema 4.4.2 se deduce que F_M es dos veces diferenciable en V y su Hessiana está dada por (4.36).

Con todo esto en mente, el siguiente resultado establece que el método propuesto posee un orden de convergencia hacia el mínimo.

Teorema 4.4.2. *Suponga que $p > 2$, F_M es fuertemente convexa en D y F_M es dos veces diferenciable en \bar{u} . Entonces, la sucesión $\{u^k\}$ generada por el Algoritmo 8 converge Q -superlinealmente hacia \bar{u} . Más aún, las sucesiones de matrices $\{B_k\}$ y $\{B_k^{-1}\}$ son acotadas.*

Observación 4.4.2. Nótese que F_M es fuertemente convexa sobre D . En efecto, comencemos señalando que D es cerrado. Efectivamente, sea $\{v^k\} \subset D$ una sucesión tal que $v^k \rightarrow v$ cuando $k \rightarrow \infty$. Puesto que $v^k \in D$ para todo k , se tiene que

$$F_M(v^k) \leq F_M(u^0) + NC \quad \forall k.$$

Así, tomando el límite cuando $k \rightarrow \infty$, y por la continuidad de F_M se sigue que $v \in D$. Como la sucesión $\{v^k\}$ es arbitraria se concluye que D es cerrado.

Puesto que F_M es fuertemente convexa en todo su dominio y además D es cerrado, entonces F_M también lo es en D .

Demostración. (Teorema 4.4.2) Iniciaremos mostrando que existe una constante $L > 0$ tal que

$$\sup_{|d|=1} |G'_M(u; d) - G'_M(\bar{u}; d)| \leq L|u - \bar{u}|, \quad (4.38)$$

para todo $u \in V$. Sea $u \in V$ arbitrario pero fijo y sea $d \in \mathbb{R}^n$ cualquiera. Notemos lo siguiente

$$\begin{aligned} &|G'_M(u; d) - G'_M(\bar{u}; d)| \\ &= |M[J_h''(p_M(u)) + M]^{-1}Md - M[J_h''(p_M(\bar{u})) + M]^{-1}Md| \end{aligned}$$

$$\begin{aligned}
&\leq |M| \left| [J_h''(p_M(u)) + M]^{-1}Md - [J_h''(p_M(\bar{u})) + M]^{-1}Md \right| \\
&\leq |M| \left| [J_h''(p_M(u)) + M]^{-1}[J_h''(p_M(\bar{u})) + M]^{-1} (J_h''(p_M(\bar{u}))Md - J_h''(p_M(u))Md) \right| \\
&\leq |M| \left| [J_h''(p_M(u)) + M]^{-1} \right| \left| [J_h''(p_M(\bar{u})) + M]^{-1} \right| \left| (J_h''(p_M(\bar{u})) - J_h''(p_M(u)))Md \right| \\
&\leq |M| \left| [J_h''(p_M(u)) + M]^{-1} \right| \left| [J_h''(p_M(\bar{u})) + M]^{-1} \right| |J_h''(p_M(\bar{u})) - J_h''(p_M(u))| |M| |d| \\
&\leq |M|^2 \left| [J_h''(p_M(u)) + M]^{-1} \right| \left| [J_h''(p_M(\bar{u})) + M]^{-1} \right| |J_h''(p_M(\bar{u})) - J_h''(p_M(u))| |d| \\
&\leq L_1 \left(\sup_{u \in V} \left| [J_h''(p_M(u)) + M]^{-1} \right| \right) |J_h''(p_M(u)) - J_h''(p_M(\bar{u}))| |d| \\
&\leq L_2 |J_h''(p_M(u)) - J_h''(p_M(\bar{u}))| |d|.
\end{aligned}$$

Nótese que

$$\sup_{u \in V} \left| [J_h''(p_M(u)) + M]^{-1} \right|,$$

es finito, pues J_h'' es una aplicación localmente Lipschitz conitnua y además p_M es Lipschitz continua. En consecuencia, se sigue que

$$|G'_M(u; d) - G'_M(\bar{u}; d)| \leq L|u - \bar{u}||d|.$$

Como d es arbitrario, es posible concluir que

$$\sup_{|d|=1} |G'_M(u; d) - G'_M(\bar{u}; d)| \leq L|u - \bar{u}|.$$

Por tanto, (4.38) se verifica sobre V .

Ahora bien, gracias a [25, lem. 2.2], (4.38) implica que

$$\frac{|\Delta \bar{y}^k - G'_M(\bar{u})\Delta u^k|}{|\Delta u^k|} \leq L \max \{|u^{k+1} - \bar{u}|, |u^k - \bar{u}|\}, \quad (4.39)$$

para k suficientemente grande (u^k elementos de V), donde $\Delta \bar{y}^k = G_M(u^{k+1}) - G_M(u^k)$. Así, en virtud de [22, teo. 5.9], la ecuación (4.39) implica la acotación de las sucesiones de matrices $\{B_k\}$ y $\{B_k^{-1}\}$, y en adición se satisface que

$$\lim_{k \rightarrow \infty} \frac{|(B_k - G'_M(\bar{u}))\Delta u^k|}{|\Delta u^k|} = 0.$$

Esta última ecuación implica la convergencia Q -superlineal del Algoritmo 8 (ver [22, teo. 5.12]). \square

Capítulo 5

Resultados

En este capítulo reportamos una serie de experimentos que modelan el flujo de un fluido de Herschel-Bulkley estacionario en una tubería. En primer lugar, discutiremos algunos aspectos importantes para implementación numérica del Algoritmo 8. Posteriormente, analizaremos el comportamiento de este, tanto para fluidos adelgazantes como espesantes. Estos últimos tienen un interés especial puesto que, para $p > 2$, el algoritmo tipo Bundle no diferenciable converge superlinealmente al mínimo (ver Teorema 4.4.2), y se desea observar esta propiedad experimentalmente.

5.1. Ensamble del algoritmo

En esta sección discutiremos los aspectos requeridos para implementar el algoritmo Bundle no diferenciable para la resolución numérica del problema de minimización (4.12).

Para la discretización de Ω , se hizo uso de PDETOOLBOX que es una herramienta asociada a MATLAB para la resolución numérica de Ecuaciones Diferenciales Parciales. Esta aplicación permite discretizar el dominio en un número finito de triángulos obteniendo una triangulación T^h en el sentido de Ciarlet (ver figura 4.1). PDETOOLBOX devuelve la tripleta de arreglos \mathbf{p} e \mathbf{t} (*points, edges and triangles*), donde \mathbf{p} guarda las coordenadas de cada nodo, \mathbf{e} almacena parejas de nodos que se conectan mediante una arista y \mathbf{t} reúne tripletas de nodos asociados a cada uno de los triángulo.

Siguiendo las ideas desarrolladas en [1], es posible implementar las matrices que intervienen en la versión discreta del funcional (ver ecuación (4.10)) mediante el manejo de la estructura \mathbf{p} e \mathbf{t} identificando los nodos de cada triángulo por medio de \mathbf{t} y tomando las coordenadas correspondientes a cada uno a través de \mathbf{p} . Para esto,

consideremos un triángulo perteneciente a T^h , cuyas coordenadas de cada vértice son (x_1, y_1) , (x_2, y_2) y (x_3, y_3) ; y sean φ_1 , φ_2 y φ_3 las funciones base asociadas, es decir,

$$\varphi_j(x_k, y_k) = \delta_{j,k}, \quad j, k = 1, 2, 3.$$

Así, φ_j es de la forma

$$\varphi_j(x, y) = \det \begin{pmatrix} 1 & x & y \\ 1 & x_{j+1} & y_{j+1} \\ 1 & x_{j+2} & y_{j+2} \end{pmatrix} / \det \begin{pmatrix} 1 & x_j & y_j \\ 1 & x_{j+1} & y_{j+1} \\ 1 & x_{j+2} & y_{j+2} \end{pmatrix},$$

y

$$\nabla \varphi_j(x, y) = \frac{1}{2|\tau|} \begin{pmatrix} y_{j+1} - y_{j+2} \\ x_{j+2} - x_{j+1} \end{pmatrix}.$$

Aquí, los índices son tomados en módulo 3 y $|\tau|$ representa el área del triángulo τ , es decir

$$2|\tau| = \det \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix}.$$

De esta manera, la jk -ésima entrada correspondiente a la matriz de rigidez A es de la forma

$$a_{jk} = \sum_{\tau \in T^h} \int_{\tau} (\nabla \varphi_j, \nabla \varphi_k) = \sum_{\tau \in T^h} \frac{|\tau|}{(2|\tau|)^2} \begin{pmatrix} y_{j+1} - y_{j+2} \\ x_{j+1} - x_{j+2} \end{pmatrix}^T \begin{pmatrix} y_{k+1} - y_{k+2} \\ x_{k+1} - x_{k+2} \end{pmatrix},$$

con índices módulo 3. Ahora bien, mediante la aplicación de la versión discreta del gradiente ∇^h y la función η descritas en la Sección 4.2, es posible implementar análogamente las matrices A_u^h y $A_{u,-1}^h$.

El siguiente paso es tener un oráculo, esto es, dado u^k se busca desarrollar una rutina que nos permita obtener $J_h(u^k)$ y $\xi \in \partial J_h(u^k)$. Por la definición de $\partial J_h(\cdot)$, nuestro oráculo determinará si u^k es igual o distinto de cero. Para este último caso resulta sencillo, pues por la ecuación (4.11) el oráculo devolverá $J_h(u^k)$ y $J'_h(u^k)$. Por otra parte, si $u^k = 0$ entonces se debe calcular una función $\psi \in \partial J_h(0)$. Como se ve en la Observación 4.1.3, si $\psi \in C_c^\infty(\Omega)$ verifica que $|\nabla \psi| \leq 1$ en Ω , entonces $\psi \in \partial J_h(0)$.

Definimos la siguiente función

$$\psi(x, y) = \begin{cases} e^{1/(|(x,y)|^2-1)} & \text{si } |(x, y)| < 1, \\ 0 & \text{si } |(x, y)| \geq 1. \end{cases} \quad (5.1)$$

Es conocido que $\psi \in C_c^\infty(\mathbb{R}^N)$ (ver [5, pág. 108]), y además se tiene que

$$|\nabla\psi(x, y)| = \begin{cases} 2e^{1/(|(x,y)|^2-1)} \frac{|(x,y)|}{(|(x,y)|^2-1)^2} & \text{si } |(x, y)| < 1, \\ 0 & \text{si } |(x, y)| \geq 1. \end{cases}$$

En la siguiente ilustración se presenta la gráfica de $|\nabla\psi(x, y)|$, como se puede observar esta función toma valores entre 0 y 0.8, y en consecuencia $|\nabla\psi| \leq 1$ en Ω . Por consiguiente $\psi \in \partial J_h(0)$. Así, implementaremos ψ para que oráculo retorne esta función y $J_h(u^k)$, cuando $u^k = 0$.

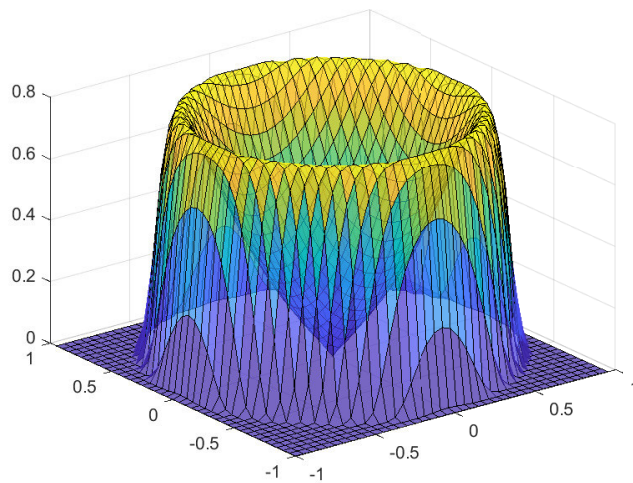


Figura 5.1: Gráfica de la función $|\nabla\psi(x, y)|$.

Como parte final para el ensamblaje del método abordaremos la resolución numérica del problema cuadrático (4.19) presente en el subalgoritmo Bundle. Haremos uso de la función de MATLAB `quadprog`, el cual es un solver para funciones cuadráticas con restricciones lineales. De esta manera, con ayuda del oráculo, nos será posible programar la función objetivo y las restricciones en (4.19). Para posteriormente, usar este solver para que retorne un nuevo par $(\nu(u), w(u))$.

Finalmente, todo el algoritmo Bundle no diferenciable fue programado en el software numérico MATLAB, en una computadora marca HP con 8 GB de RAM y con procesador Intel Core i7 de décima generación.

5.2. Experimentos y resultados numéricos

5.2.1. Caso $1 < p < 2$

En los siguientes experimentos observaremos el comportamiento del Algoritmo 8 para el problema de flujo en el caso de un fluido de Herschel-Bulkley cuyo comportamiento es adelgazante (*shear-thinning behaviour*), es decir $1 < p < 2$. Además, tomaremos $f = 1$, la constante que representa un decaimiento lineal de la presión en la tubería. Consideraremos g como la constante de Oldroyd la cual modela el umbral de plasticidad del material (ver [15]).

De aquí y en adelante, usaremos triangulaciones uniformes descritas por h , el radio de la circunferencias inscritas de los triángulos pertenecientes a la malla de discretización. Inicializaremos el Algoritmo 8 con la solución del problema de Poisson $-\Delta u_0^h = f^h$. A menos que se diga lo contrario, tomaremos $\delta_0 = 0.0001$ y $\delta_k = 1/2^k$ para $k = 1, \dots$. En adición, usaremos el criterio de parada obtenido gracias al Teorema 3.2.1, generalmente tomaremos $tol = 1e - 05$ o $tol = 1e - 06$.

Experimento 1

En este experimento tomamos Ω como la bola unitaria de \mathbb{R}^2 . Modelamos el flujo de un material con $p = 1.75$. Analizamos el comportamiento del Algoritmo 8 con $g = 0.2$ y una malla dada por $h \approx 0.0086$. Para la regularización de Moreau-Yosida (ver ecuación (4.15)) consideramos $M = A$, la matriz de rigidez obtenida por el método de elementos finitos.

En la Figura 5.2 se puede observar la función de velocidad obtenida y su perfil a lo largo del diámetro de la tubería. Estos gráficos ilustran las propiedades mecánicas esperadas del material, es decir, puesto que la tensión de corte transmitida mediante una capa del fluido disminuye hacia el centro de la tubería, el fluido de Herschel-Bulkley se mueve como un sólido en dicho sector. Este fenómeno describe el aplanamiento de la velocidad en el centro de la tubería.

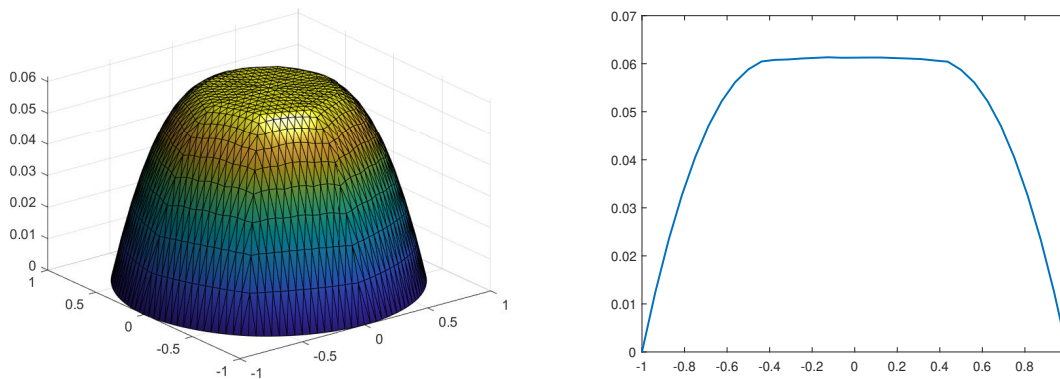


Figura 5.2: Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo del diámetro de la tubería (derecha). Parámetros: $p = 1.75$, $g = 0.2$ y $M = A$.

En el Cuadro 5.1 se presenta el comportamiento general del método en cada iteración hasta su convergencia. Para cada iteración se muestra el valor de J_h , la norma de u^k en W_0^h , el tipo de actualización para la matriz B_k , la iteraciones del subalgoritmo Bundle (Algoritmo 5), el valor de ν , la longitud de paso y las subiteraciones realizadas por el Algoritmo 7.

it.	$J_h(u^k)$	$\ u^k\ $	B_k	Its. Bundle	$ \nu $	λ_k	it. ls.
1	0.0812	0.6545	M	389	0.0914	1	0
2	-0.0102	0.3934	M	1	0.0189	1	0
3	-0.0252	0.2830	M	1	0.0045	1	0
4	-0.0293	0.2301	M	1	$3.3207e - 04$	1	0
5	-0.0296	0.2123	M	1	$8.2777e - 07$	1	0

Cuadro 5.1: Comportamiento general del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = A$.

Es posible observar que el valor del funcional J_h es reducido monótonamente en cada iteración. Además $\|u^k\| \neq 0$, lo cual nos indica que el Algoritmo 8 no pasó por el punto donde el funcional no es diferenciable. En cuanto a las iteraciones del subalgoritmo Bundle vemos que al inicio requiere de bastantes iteraciones para converger, posteriormente solo le toma realizar una iteración en lo que val del método. Esto sucede por la elección de $\{\delta_k\}$ que hemos realizado, pues en un inicio δ_0 es mucho más pequeño que los primeros δ_k , haciendo que el costo computacional sea mayor en la primera iteración. Por otra parte, nótese que el criterio de parada $|\nu|$, decrece muy rápido en las últimas iteraciones, sugiriéndonos que el Algoritmo 8 posee una tasa de convergencia rápida cerca del mínimo. Por último, la longitud de paso se mantiene invariante durante todo el proceso, así, el Algoritmo 7 no requiere realizar ninguna iteración.

A continuación, en la Figura 5.3 se observa el decrecimiento de los valores de ν en cada iteración, esto nos permite corroborar una convergencia rápida del Algoritmo 8.

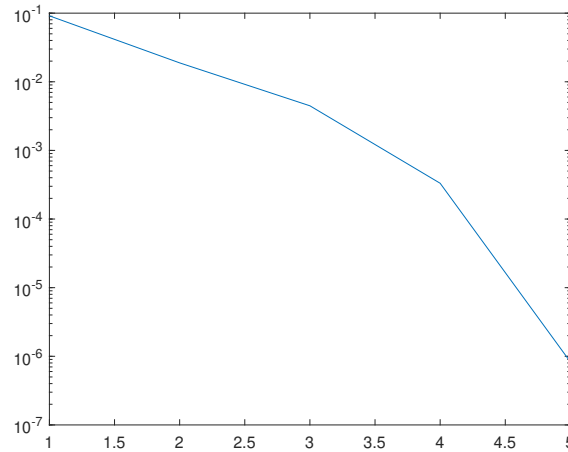


Figura 5.3: Valores de ν obtenidos en cada iteración del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = A$.

Finalmente, cabe mencionar que este experimento fue ejecutado por el computador en un tiempo total de 434.21 segundos.

Experimento 2

En el presente experimento mantenemos Ω como la bola unitaria de \mathbb{R}^2 y tomamos $p = 1.5$. Es conocido que para valores pequeños de p el funcional no es preciso dado que operador p -laplaciano se vuelve más degenerado, por consiguiente, la minimización del mismo es más complicada (ver [13]). Además, si g es más grande, entonces la contribución del funcional no diferenciable J_2 es mucho mayor, haciendo que la aproximación numérica de nuestro problema sea más complicada. Así, probamos el Algoritmo 8 para varios valores de g con el fin observar el comportamiento del mismo. Como en el experimento anterior, tomaremos $M = A$, la matriz de elementos finitos para la regularización de Moreau-Yosida.

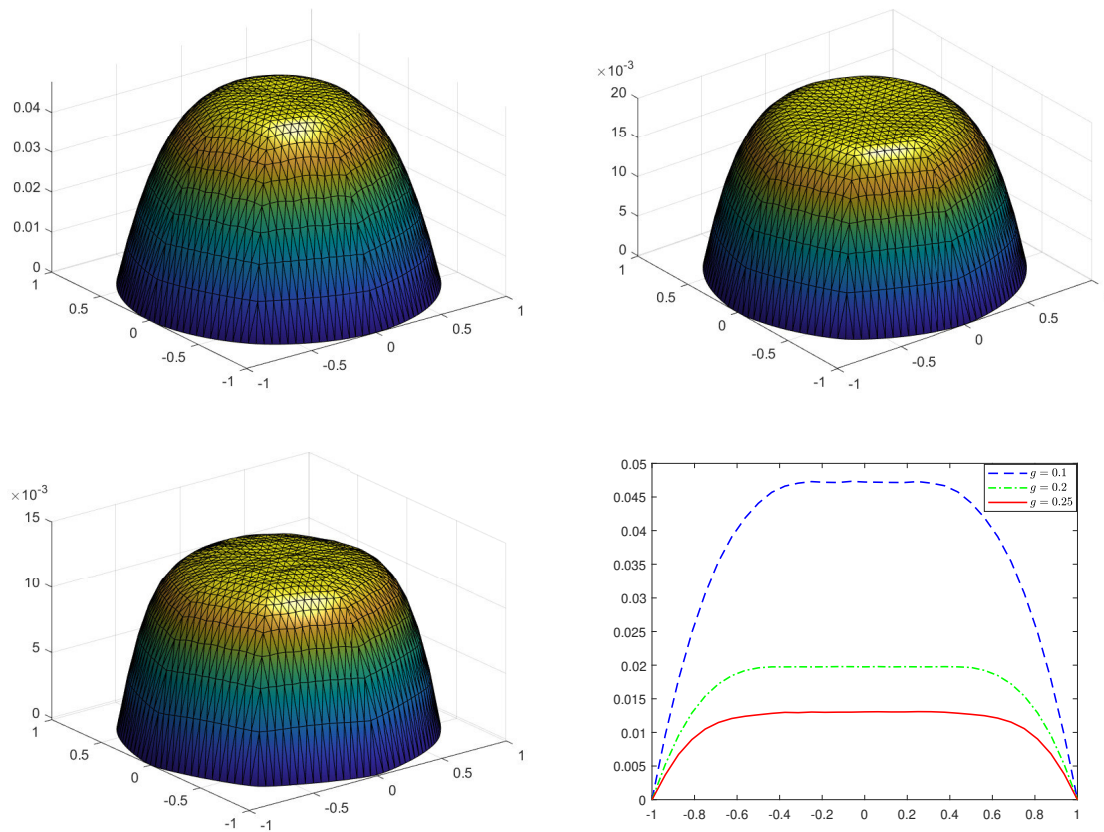


Figura 5.4: Velocidad u calculada para $p = 1.5$ y $g = 0.1$ (superior izquierda), $g = 0.2$ (superior derecha) y $g = 0.25$ (inferior izquierda). Perfiles de las velocidades calculadas a lo largo del diámetro de la tubería (inferior derecha).

Las funciones de velocidad resultantes y sus perfiles a lo largo del diámetro de la tubería se pueden observar en la Figura 5.4. Como en el caso anterior, la tensión de corte transmitida mediante una capa del fluido disminuye hacia el centro de la tubería, ocasionando que el fluido posea un comportamiento de sólido rígido en dicho sector. En adición, las figuras incluidas nos indican que el incremento de g genera que el fluido posea una mayor resistencia al movimiento, su velocidad sea casi nula y las zonas planas sean más grandes. Este comportamiento concuerda con la contribución de los experimentos en [13, 15].

En el Cuadro 5.2 se presenta el comportamiento del Algoritmo 8, se puede observar que para g más grande entonces la resolución del problema (4.12) requiere de más iteraciones, así como un mayor tiempo de ejecución. En cuanto a las iteraciones realizadas por el subalgoritmo Bundle se evidencia un gran incremento para el último caso, mostrando la que la resolución numérica de este es mucho más complicada. Por otro lado, para todos los valores de g , el algoritmo no ejecutó ninguna actualización de tipo BFGS. Además, se comprobó que las velocidades calculadas son distintas de 0.

g	$J(\bar{u})$	$\ \bar{u}\ $	Nro. its.	Its. Bundle	Its. BFGS	Tiempo
0.1	-0.0259	0.1712	5	343	0	381.87 s
0.2	-0.0091	0.5019	5	335	0	467.32 s
0.25	-0.0046	0.0551	6	409	0	762.54 s

Cuadro 5.2: Comportamiento general del Algoritmo 8. Parámetros: $p = 1.5$ y $M = A$.

Finalmente, en la Figura 5.5 se presenta el comportamiento de convergencia del Algoritmo 8 a través de los valores de ν en cada iteración, para cada uno de los casos de g . Se puede observar un decrecimiento monótono y en adición un gran decaimiento en las iteraciones finales, esto nos dice que el método converge rápidamente cuando está cerca del mínimo.

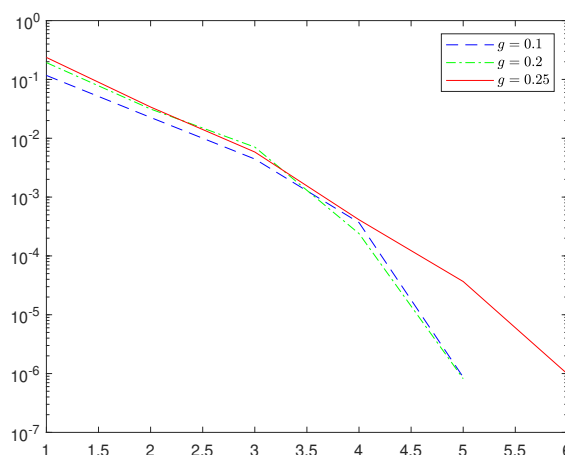


Figura 5.5: Valores calculados de ν por el Algoritmo 8. Parámetros: $p = 1.5$ y $M = A$.

Experimento 3

En este experimento fijamos $\Omega \subset \mathbb{R}^2$ como el cuadrado $[-1, 1] \times [-1, 1]$. Modelamos el flujo de un material con $p = 1.75$. Analizamos el comportamiento del Algoritmo 8 con $g = 0.2$ y una malla dada por $h \approx 0.0047$. Para la regularización de Moreau-Yosida consideramos $M = I$, la matriz identidad en \mathbb{R}^n .

La función de velocidad resultante y su perfil a lo largo de la diagonal de la tubería se exhiben en la Figura 5.6. Así como en los experimentos previos, se ilustran las propiedades mecánicas esperadas del material. Como la tensión de corte transmitida mediante una capa del fluido disminuye hacia el centro de la tubería, es ahí donde el fluido se mueve como un sólido.

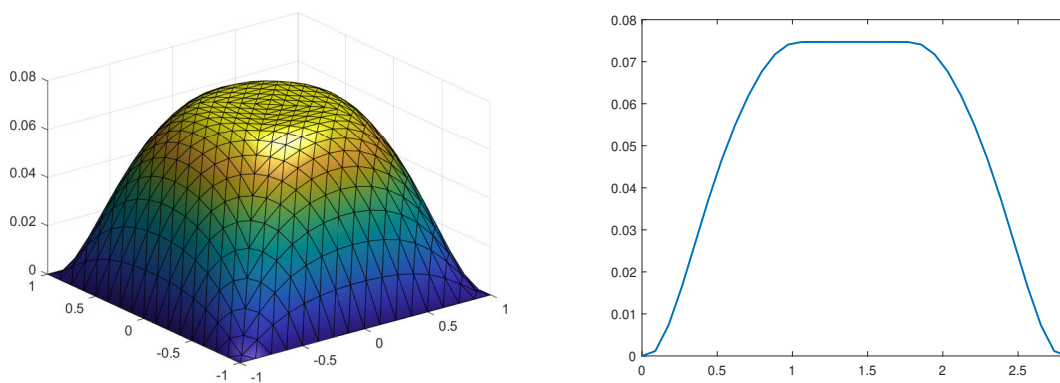


Figura 5.6: Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo de la diagonal de la tubería (derecha). Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$.

it.	$J_h(u^k)$	$\ u^k\ $	B_k	Its. Bundle	$ \nu $	λ_k	it. ls.
1	0.1163	0.8133	M	57	0.0156	1	0
10	0.0300	0.6382	M	1	0.0058	1	0
19	-0.0104	0.5047	M	1	0.0034	1	0
38	-0.0421	0.2858	M	1	$6.0815e - 05$	1	0
39	-0.0422	0.2800	M	1	$2.9216e - 05$	1	0
40	-0.0422	0.2773	M	1	$1.1508e - 05$	0.4973	1
41	-0.0422	0.2745	M	1	$3.2250e - 06$	0.4973	0

Cuadro 5.3: Comportamiento general del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$.

En el Cuadro 5.3 observamos el comportamiento del método. En comparación con los experimentos anteriores se nota que existe un gran incremento en el número de iteraciones realizadas hasta su convergencia. Otro aspecto que resalta este experimento es el decrecimiento del funcional J_h , durante las últimas iteraciones su valor no presenta variación alguna. Por otra parte, se puede notar que $\|u^k\| \neq 0$ lo cual nos permite deducir que el Algoritmo 8 no pasó por el punto donde J_h es no diferenciable. Durante la ejecución de este experimento las matrices B_k se mantuvieron constantes. Además, a partir de la segunda iteración del método, el subalgoritmo Bundle realizó exactamente una iteración, nuevamente, esto se atribuye a la elección de $\{\delta_k\}$ realizada. Se observa también que los valores de ν presentan un decrecimiento monótono. En cuanto a la longitud de paso λ_k se puede notar que fue constante hasta la penúltima iteración, el Algoritmo 7 realizó una subiteración para converger, se observa que λ_k fue reducido más o menos a la mitad.

En la Figura 5.7 se presenta el decrecimiento de los valores de ν de en cada iteración.

Se puede notar que, a partir de la iteración 31 el descenso es más pronunciado. Esto nos dice nuevamente que, cuando el Algoritmo 8 está cerca del mínimo, entonces este tiene una tasa de convergencia bastante rápida.

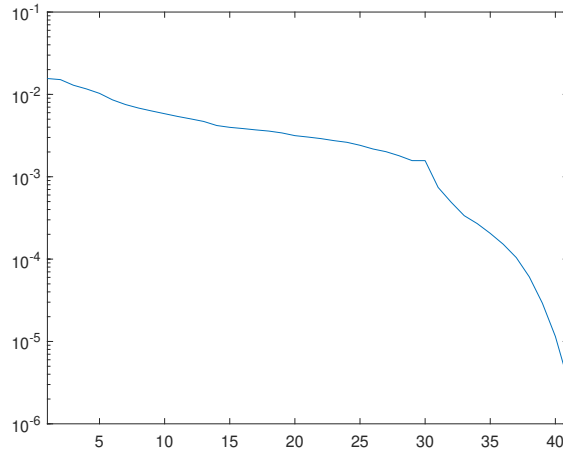


Figura 5.7: Valores calculados de ν por el Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$.

Finalmente, la ejecución total de este experimento tuvo una duración de 5868.73 segundos.

Experimento 4

En este experimento se analiza el comportamiento del Algoritmo 8 ante la variación del parámetro de inicialización δ_0 . Fijamos Ω como la bola unitaria de \mathbb{R}^2 . Modelamos el flujo de un material con $p = 1.75$, $g = 0.2$ y una malla dada por $h \approx 0.0386$. Para la matriz asociada a la regularización de Moreau-Yosida consideramos $M = I$. Inicializaremos el Algoritmo 8 con δ_0 igual a 1, 0.75, 0.50 y 0.25.

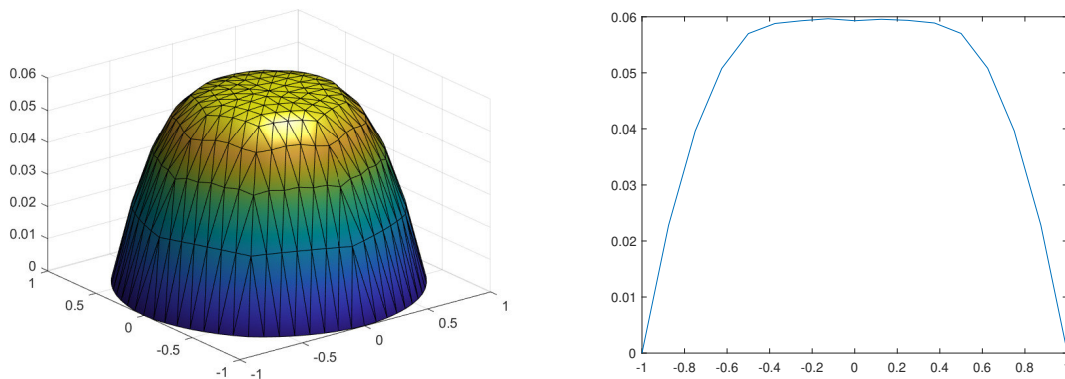


Figura 5.8: Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo del diámetro de la tubería (derecha). Parámetros: $p = 1.75$, $g = 0.2$, $M = I$ y $\delta_0 = 0.25$.

En la Figura 5.8 se presenta la función de velocidad y su perfil a lo largo del diámetro de la tubería. Como en el primer experimento, estos gráficos ilustran las mismas propiedades mecánicas esperadas del material.

El comportamiento general del método se exhibe en el Cuadro 5.4 para cada valor de δ_0 . Como se puede observar en los valores de $J_h(\bar{u})$ y $\|\bar{u}\|$, si δ_0 es mayor, entonces el método es menos preciso en calcular la solución aproximada. Cuando $\delta_0 = 1$, las iteraciones realizadas por el Algoritmo 8 incrementan de forma abrupta; para el resto de valores se observa un número menor de iteraciones realizadas. En cuanto al comportamiento del subalgoritmo Bundle cabe mencionar que, por cada iteración del algoritmo general, este método realizó en promedio unas 5 iteraciones para converger. Para cada δ_0 , se presenta el número total de iteraciones realizadas por el subproceso Bundle.

En este experimento se pudo observar que el Algoritmo 8 sí realizó actualizaciones de tipo BFGS. Se puede notar que, para δ_0 pequeño, el método realiza menos iteraciones con actualización de este tipo. Este comportamiento también se observa en los experimentos previos. En dichos casos se inicializó con $\delta_0 = 0.0001$, provocando que el Algoritmo 8 no realice actualizaciones BFGS en su ejecución total.

δ_0	$J_h(\bar{u})$	$\ \bar{u}\ $	Nro. its.	Its. Bundle	Its. BFGS	Tiempo
1	-0.0283	0.2074	108	436	24	8145.27 s
0.75	-0.0286	0.2096	38	161	4	787.35 s
0.50	-0.0288	0.2109	18	105	3	495.73 s
0.25	-0.0290	0.2115	26	253	3	624.65 s

Cuadro 5.4: Comportamiento del Algoritmo 8 ante distintos δ_0 . Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$.

Ahora, el Cuadro 5.5 recopila la información del desenvolvimiento del Algoritmo 8 con $\delta_0 = 0.75$ como parámetro de inicialización. Se observa que en las iteraciones con actualización BFGS el valor del funcional decae más rápido que en las futuras iteraciones donde se toma la matriz de actualización $B_k = M$. Nótese que el subalgoritmo Bundle en general no tiene un coste computacional alto, en las últimas iteraciones del método se observa que este subproceso requiere realizar más iteraciones para converger, sin embargo, no excede en gran número como se ha visto en los experimentos previos. Además, se observa los valores de la longitud de paso así como el número de iteraciones realizadas por el algoritmo de búsqueda lineal en cada iteración.

it.	$J_h(u^k)$	$\ u^k\ $	B_k	Its. Bundle	$ \nu $	λ_k	it. ls.
1	0.0862	0.6516	<i>BFGS</i>	2	0.0576	1	0
2	0.0398	0.5522	<i>BFGS</i>	2	0.0316	1	0
3	0.0278	0.3400	<i>BFGS</i>	6	0.0210	1	0
5	0.0256	0.3361	<i>BFGS</i>	5	0.0538	0.0046	4
15	0.0246	0.3314	<i>M</i>	1	0.0501	0.0046	0
25	0.0223	0.3280	<i>M</i>	1	0.0478	0.0046	0
30	-0.0265	0.2361	<i>M</i>	3	$4.0416e - 04$	0.5000	1
35	-0.0271	0.2117	<i>M</i>	14	$2.1106e - 05$	0.5000	0
38	-0.0286	0.2065	<i>M</i>	9	$2.7607e - 07$	0.2500	1

Cuadro 5.5: Comportamiento general del Algoritmo 8. Parámetros: $p = 1.75$, $g = 0.2$, $M = I$ y $\delta_0 = 0.75$.

En la Figura 5.9 se observa la convergencia del método a través de los valores de ν obtenidos en cada iteración. Para $\delta_0 = 1$, se puede observar un decaimiento monótono hasta la décima iteración, posteriormente, el parámetro ν oscila hasta el final de las iteraciones, requiriendo que la ejecución total del método tome una mayor cantidad de tiempo.

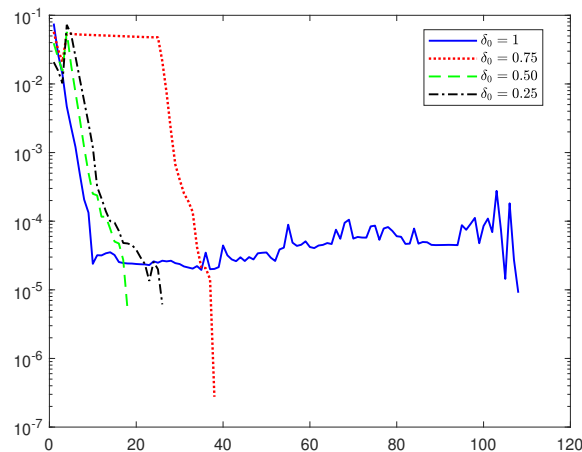


Figura 5.9: Valores de ν calculados por el Algoritmo 8 para cada δ_0 . Parámetros: $p = 1.75$, $g = 0.2$ y $M = I$.

Para el resto de valores de δ_0 , se observa que el método presenta un comportamiento de convergencia similar. En general, se nota un decaimiento monótono de los valores de ν , en el caso $\delta_0 = 0.75$ el descenso es más profundo. Para estos casos, el tiempo de ejecución del método es mucho más rápido que el tiempo requerido para $\delta_0 = 1$.

5.2.2. Caso $p > 2$

En los siguientes experimentos observaremos el comportamiento del Algoritmo 8 para el problema de flujo para un fluido de Herschel-Bulkley cuyo comportamiento es espesante (*shear-thickening behaviour*), es decir $p > 2$. Como en la sección anterior, tomaremos $f = 1$, la constante que representa un decaimiento lineal de la presión en la tubería. Consideraremos g como la constante de Oldroyd la cual modela el umbral de plasticidad del material (ver [15]).

De igual manera, usaremos triangulaciones uniformes descritas por h , el radio de la circunferencias inscritas de los triángulos pertenecientes a la malla de discretización. Inicializaremos el Algoritmo 8 con la solución del problema de Poisson $-\Delta u_0^h = f^h$. A menos que se diga lo contrario, tomaremos $\delta_0 = 0.0001$ y $\delta_k = 1/2^k$ para $k = 1, \dots$. En adición, usaremos el criterio de parada obtenido gracias al Teorema 3.2.1, generalmente fijaremos $tol = 1e - 05$ o $tol = 1e - 06$.

Experimento 1

En este experimento tomamos Ω como la bola unitaria en \mathbb{R}^2 . Modelamos el flujo de un material con $p = 4$. Analizamos el comportamiento del Algoritmo 8 con $g = 0.1$, $g = 0.2$ y $g = 0.3$. Usamos una malla dada por $h \approx 0.0086$. Para la regularización de Moreau-Yosida consideramos $M = A$, la matriz de rigidez obtenida por el método de elementos finitos.

En la Figura 5.10 se aprecian las velocidades calculadas por el Algoritmo 8 así como los perfiles de las mismas a lo largo del diámetro de la tubería. Los gráficos ilustran las propiedades mecánicas esperadas del material, esto es, la viscosidad de los fluidos espesantes incrementa con la tasa del esfuerzo de corte. Puesto que la tensión de corte transmitida mediante una capa del fluido decrece hacia el centro de la tubería, la velocidad toma una forma cónica con una parte plana exactamente al centro del dominio. Además, se puede observar que para un valor de g mayor entonces el tamaño de estas zonas planas incrementa.

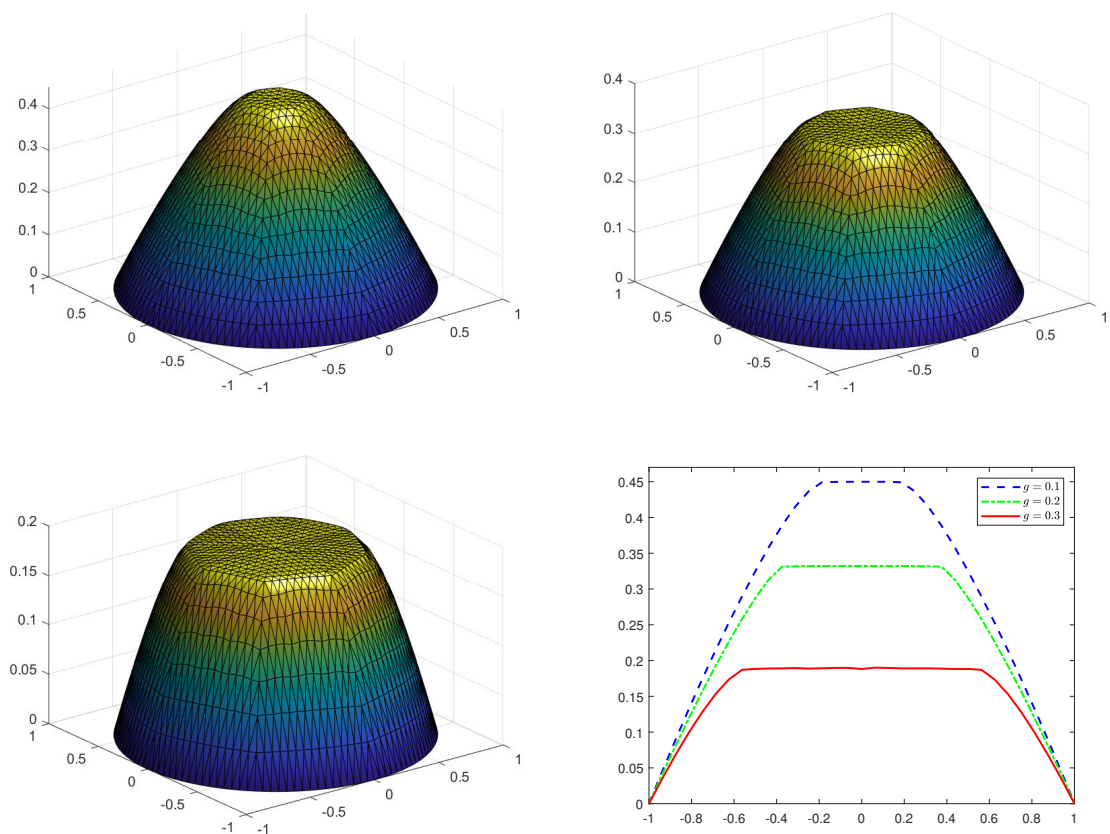


Figura 5.10: Velocidad u calculada para $p = 4$ y $g = 0.1$ (superior izquierda), $g = 0.2$ (superior derecha) y $g = 0.3$ (inferior izquierda). Perfiles de las velocidades calculadas a lo largo del diámetro de la tubería (inferior derecha).

En el Cuadro 5.6 se describe el comportamiento del Algoritmo 8 para cada caso de g . Se exhibe el valor del funcional J_h en el mínimo \bar{u} así como la norma de este punto. Como se menciona en [13, 15], para valores grandes de g el problema es menos regular y por tanto más difícil de ser aproximado, sin embargo, se puede observar que el algoritmo requirió de menos iteraciones para su convergencia para $g = 0.3$ que $g = 0.2$, lo que resulta ser una buena cualidad de nuestro método. Por otra parte, nótese que el número de iteraciones del subalgoritmo 5 incrementa cuando g es más grande, lo que en términos del tiempo de ejecución del método representa un mayor coste computacional.

g	$J_h(\bar{u})$	$\ \bar{u}\ $	Nro. its.	Its. Bundle	Its. BFGS	Tiempo
0.1	-0.3780	0.8255	7	220	0	479.62 s
0.2	-0.2174	0.7282	12	367	0	2560.78 s
0.3	-0.1872	0.5656	9	532	0	32430.54 s

Cuadro 5.6: Comportamiento general del Algoritmo 8. Parámetros: $p = 4$ y $M = A$.

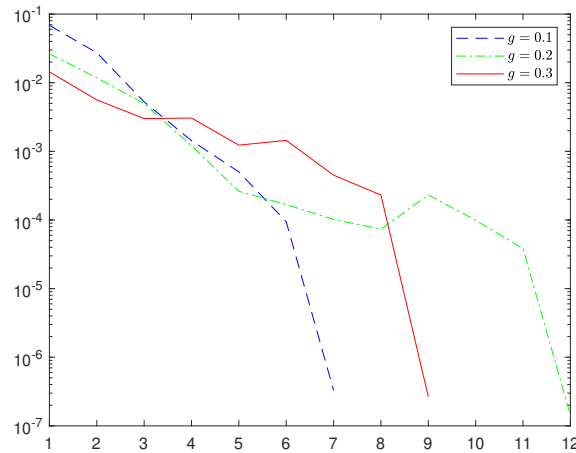


Figura 5.11: Valores de ν obtenidos en cada iteración del Algoritmo 8. Parámetros: $p = 4$ y $M = A$.

Finalmente, En la Figura 5.11 se muestra el comportamiento de convergencia del Algoritmo 8 para cada valor de g , a través de los valores de ν obtenidos en cada iteración. Es posible corroborar la convergencia superlineal del método para este caso, se puede notar como se presentan oscilaciones en las primeras iteraciones y posteriormente se presenta un descenso abrupto, clásico de este tipo de convergencia.

Experimento 2

En este experimento fijamos $\Omega \subset \mathbb{R}^2$ como el cuadrado $[-1, 1] \times [-1, 1]$. Modelamos el flujo de un material con $p = 4$. Analizamos el comportamiento del Algoritmo 8 con $g = 0.2$ y una malla dada por $h \approx 0.0047$. Para la regularización de Moreau-Yosida consideramos $M = I$, la matriz identidad en \mathbb{R}^n .

La función de velocidad y su perfil a lo largo de la diagonal de la tubería se exhiben en la Figura 5.12. De la misma manera que en el experimento anterior, los gráficos permiten apreciar las propiedades mecánicas esperadas del material. Como la tensión de corte transmitida mediante una capa del fluido decrece hacia el centro de la tubería, la velocidad toma una forma cónica con una parte plana exactamente al centro de la región.

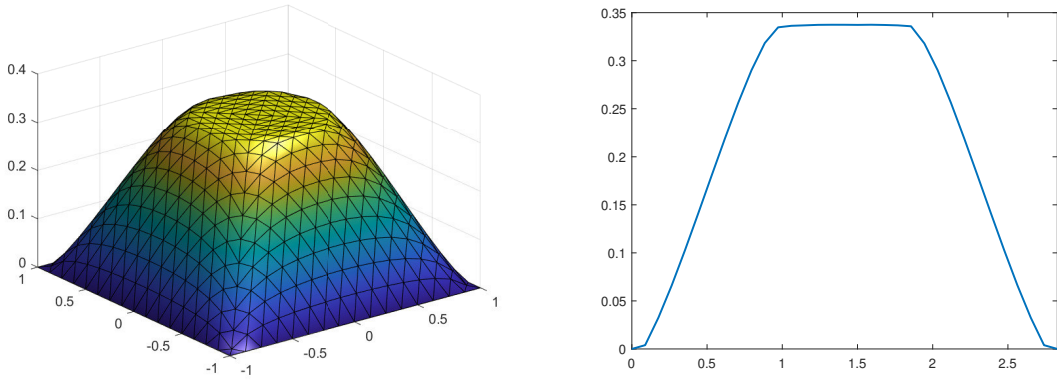


Figura 5.12: Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo de la diagonal de la tubería (derecha). Parámetros: $p = 4$, $g = 0.2$ y $M = I$.

El comportamiento del Algoritmo 8 se reporta en el Cuadro 5.3. Nuevamente se observa que existe un gran incremento en cuanto al número de iteraciones realizadas por el método hasta converger. Nótese que los valores del funcional J_h decrecen monótonamente hasta la iteración 149 y no varía hasta en las próximas 45 iteraciones, provocando un mayor coste computacional, posteriormente el valor del funcional sigue decreciendo. A lo largo de la ejecución de este experimento las matrices B_k se mantuvieron constantes. Se observa que el subalgoritmo Bundle no requiere de muchas iteraciones para converger, incluso en la primera iteración general del método. Por otra parte, nótese que $\|u^k\| \neq 0$, así, se puede inferir que el Algoritmo 8 no pasó por el punto donde J_h es no diferenciable. Por otro lado, se evidencia decrecimiento de la longitud de paso, se comprobó que el algoritmo 7 realizó como máximo 3 iteraciones internas para converger.

it.	$J_h(u^k)$	$\ u^k\ $	B_k	Its. Bundle	$ \nu $	λ_k	it. ls.
1	-0.2520	0.5868	M	9	0.0048	1	0
35	-0.2855	0.7528	M	3	$2.4094e - 04$	0.5000	2
41	-0.2862	0.7586	M	1	$2.5584e - 04$	0.2500	1
42	-0.2863	0.7587	M	1	$2.5471e - 04$	0.0493	2
149	-0.2870	0.7707	M	1	$3.3492e - 04$	0.0246	1
176	-0.2870	0.7710	M	1	$6.9767e - 04$	0.0031	1
194	-0.2870	0.7712	M	1	$3.9434e - 04$	0.0031	0
251	-0.2881	0.7802	M	1	$1.4760e - 04$	0.0031	0
290	-0.2890	0.7876	M	2	$1.6270e - 07$	0.0003	3

Cuadro 5.7: Comportamiento general del Algoritmo 8. Parámetros: $p = 4$, $g = 0.2$ y $M = I$.

En la Figura 5.13 se ilustra el comportamiento de convergencia del Algoritmo 8 a través de los valores de ν calculados en cada iteración. Se puede notar que aproximadamente hasta la iteración 240, $|\nu|$ se mantiene en el intervalo de $1e - 03$ y $1e - 04$. Posteriormente se observan oscilaciones y en la iteración final se evidencia un descenso pronunciado, lo que nuevamente permite comprobar una tasa de convergencia superlineal del método.

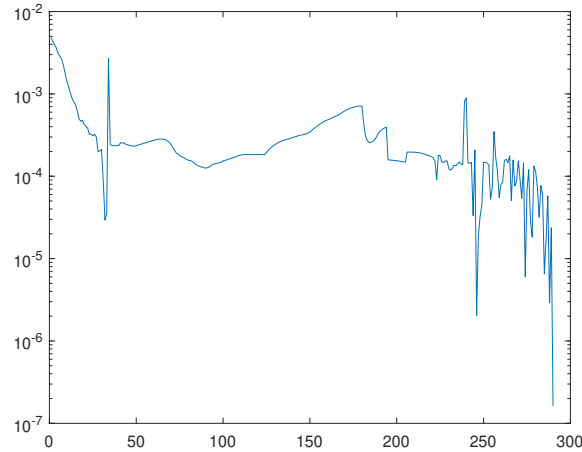


Figura 5.13: Valores de ν obtenidos en cada iteración del Algoritmo 8. Parámetros: $p = 4$, $g = 0.2$ y $M = I$.

Por último, la ejecución total del método duró 7234.87 segundos.

Experimento 3

En este experimento fijamos Ω como la bola unitaria de \mathbb{R}^2 . Como lo menciona González-Andrade en [13], cuando el valor de p es grande, entonces la resolución numérica del problema (4.12) resulta desafiante. Consecuentemente, experimentamos el Algoritmo 8 modelando el flujo de un material con $p = 8$ y $p = 10$. Analizamos el comportamiento del método con $g = 0.2$ y una malla dada por $h \approx 0.0086$. Para la regularización de Moreau-Yosida consideramos $M = A$, la matriz de rigidez obtenida por el método de elementos finitos.

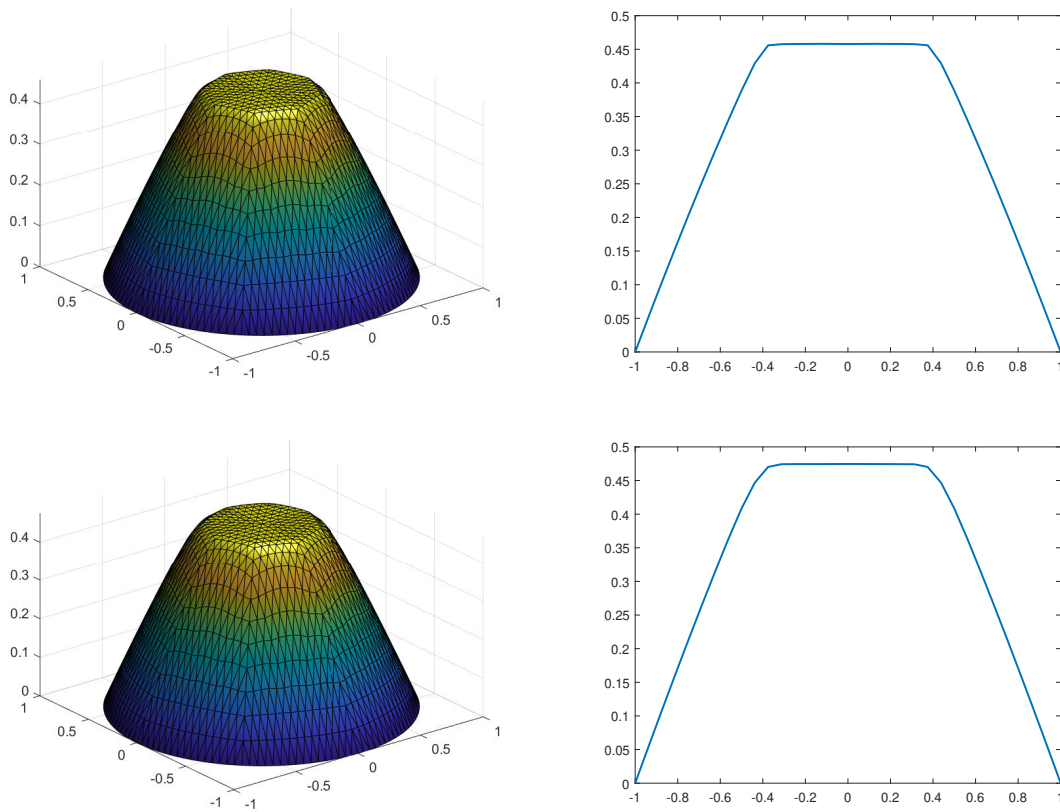


Figura 5.14: Velocidad u calculada para $p = 8$ (superior izquierda) y $p = 10$ (inferior izquierda). Perfiles de las velocidades calculadas a lo largo del diámetro de la tubería con $p = 8$ (superior derecha) y $p = 10$ (inferior derecha). Parámetros: $g = 0.2$ y $M = A$.

En la Figura 5.14 se exhiben las velocidades resultantes y su perfil a lo largo del diámetro la tubería. Nuevamente estos gráficos ilustran las propiedades mecánicas esperadas del material, es decir, la viscosidad de los fluidos espesantes incrementa con la tasa del esfuerzo de corte. Como la tensión de corte transmitida mediante una capa del fluido decrece hacia el centro de la tubería, la velocidad toma una forma cónica con una parte plana exactamente al centro de la región.

El reporte del comportamiento del Algoritmo 8 para ambos casos se encuentra en el Cuadro 5.8. Se presenta el valor del funcional J_h en sus puntos minimales así como la norma de los mismos. En cuanto al número de iteraciones es interesante observar que para $p = 10$ el método necesitó de 109 iteraciones para converger, evidenciándose la dificultad de resolución del problema. Por otra parte, se observa que las iteraciones realizadas por el subalgoritmo Bundle son numerosas, lo que influye directamente en el tiempo total de la ejecución del método. Se puede notar que la convergencia del algoritmo tomó una gran cantidad de tiempo.

p	$J_h(\bar{u})$	$\ \bar{u}\ $	Nro. its.	Its. Bundlle	Its. BFGS	Tiempo
8	-0.3376	0.8743	11	311	0	49325.69 s
10	-0.3639	0.8998	109	1045	0	92880.14 s

Cuadro 5.8: Comportamiento general del Algoritmo 8. Parámetros: $g = 0.2$ y $M = A$.

Por último, en la Figura 5.15 se muestra el comportamiento de convergencia del Algoritmo 8 para cada valor de p , a través de los valores de ν obtenidos en cada iteración. Es posible corroborar la convergencia superlineal del método para ambos. Para $p = 8$ se observa un decrecimiento monótono y posteriormente un descenso abrupto, clásico del comportamiento superlineal. Por otra parte para $p = 10$ se puede notar que los valores de ν se mantienen constantes por tramos, este fenómeno se atribuye al subalgoritmo Bundle, pues la nueva información agregada al bundle no era lo suficientemente buena para obtener un mejor valor de ν . Sin embargo, es posible notar un descenso pronunciado a partir de la iteración 97.

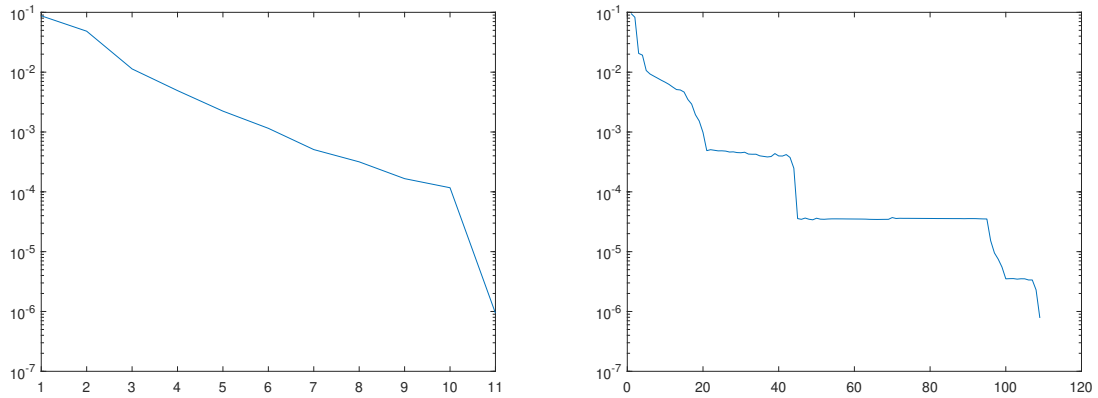


Figura 5.15: Valores de ν obtenidos por el Algoritmo 8 para $p = 8$ (izquierda) y $p = 10$ (derecha). Parámetros: $g = 0.2$ y $M = A$.

Experimento 4

En este experimento se analiza el comportamiento del Algoritmo 8 ante la variación del parámetro de inicialización δ_0 . Fijamos Ω como la bola unitaria de \mathbb{R}^2 . Modelamos el flujo de un material con $p = 4$, $g = 0.2$ y una malla dada por $h \approx 0.0386$. Para la matriz asociada a la regularización de Moreau-Yosida consideramos $M = I$. Inicializaremos el Algoritmo 8 con δ_0 igual a 1, 0.75, 0.50 y 0.25.

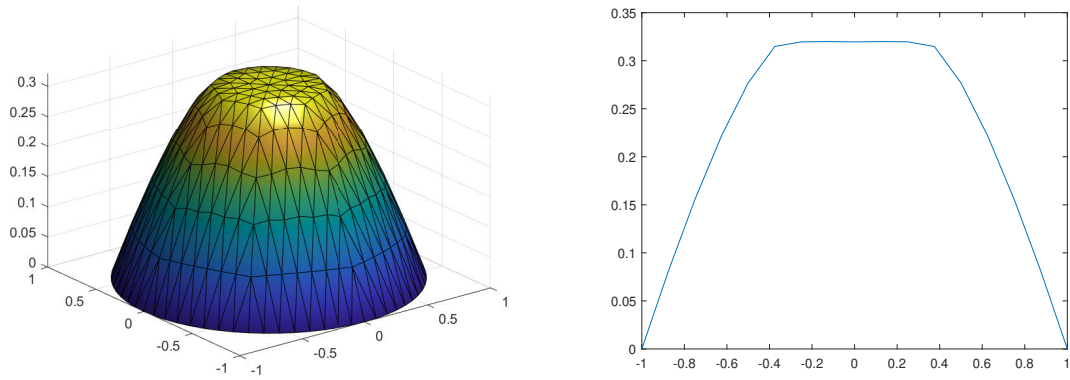


Figura 5.16: Velocidad calculada u (izquierda) y perfil de la velocidad a lo largo del diámetro de la tubería (derecha). Parámetros: $p = 4$, $g = 0.2$, $M = I$ y $\delta_0 = 0.25$.

En la Figura 5.16 se exhibe la función de velocidad y su perfil a lo largo del diámetro de la tubería. Como en el primer experimento de esta sección, estos gráficos ilustran las mismas propiedades mecánicas esperadas del material.

Para cada valor de δ_0 , se presenta el comportamiento del Algoritmo 8 en el Cuadro 5.9. En este caso, se puede observar que, para δ_0 mayor, entonces el método encuentra una solución más precisa, ya que el valor del funcional J_h en \bar{u} es menor, así como el valor de $\|\bar{u}\|$, que incrementa. Con respecto al número de iteraciones, el método con parámetro de inicialización $\delta_0 = 1$ presenta un mejor comportamiento que en los casos $\delta_0 = 0.75$ y $\delta_0 = 0.50$. Sin embargo, es evidente que el Algoritmo 8 presenta un mejor comportamiento cuando $\delta_0 = 0.25$ que en los demás casos.

Es importante mencionar que, para $\delta_0 = 0.75$ y $\delta_0 = 0.50$, el Algoritmo 8 arrojó el mismo comportamiento, se puede observar que en ambos casos se obtuvo la misma solución, el mismo número de iteraciones, mismo número de iteraciones del subalgoritmo Bundle y misma cantidad de actualizaciones BFGS. Así como la misma cantidad de tiempo de ejecución del método.

El comportamiento del subalgoritmo Bundle es bastante bueno para cada parámetro de inicialización. Es posible observar que en el último caso realiza casi el mismo número de iteraciones que las que el método general ha realizado. En los casos intermedios, se nota que el número de iteraciones realizadas son aproximadamente el doble de iteraciones que realiza el Algoritmo 8. Para el primer caso, se observa que el número de iteraciones realizadas por el subalgoritmo Bundle excede en más del triple de iteraciones generadas por el método general.

δ	$J_h(\bar{u})$	$\ \bar{u}\ $	Nro. its.	Its. Bundle	Its. BFGS	Tiempo
1	-0.2087	0.7012	30	103	9	930.87 s
0.75	-0.2086	0.6969	37	71	6	1086.27 s
0.50	-0.2086	0.6969	37	71	6	1086.27 s
0.25	-0.2080	0.6798	19	21	1	365.21 s

Cuadro 5.9: Comportamiento general del Algoritmo 8 ante distintos δ_0 . Parámetros: $p = 4$, $g = 0.2$ y $M = I$.

Como en la sección anterior, la variación del parámetro de inicialización δ_0 permite que el método realice iteraciones con la actualización BFGS. Nuevamente se observa que, cuando δ_0 es pequeño, entonces el método realiza menos actualizaciones BFGS. Esto también se apoya con los experimentos previos de esta sección, puesto que al tomar $\delta_0 = 0.0001$, el método no realiza ninguna iteración con la matriz de Quasi-Newton actualizada mediante BFGS.

La ventaja más notoria de la actualización BFGS es que el método converge más rápido hacia a la solución. Esto se puede notar en el tiempo total de ejecución del Algoritmo 8. En el primer caso, el método requiere de un tiempo menor que para el segundo y tercer caso. Incluso cuando el número de iteraciones del subalgoritmo Bundle es excesivamente mayor que las iteraciones realizadas por el método general.

it.	$J_h(u^k)$	$\ u^k\ $	B_k	Its. Bundle	$ \nu $	λ_k	it. ls.
1	-0.1695	0.4893	<i>BFGS</i>	1	0.0266	1	0
2	-0.1697	0.5703	<i>BFGS</i>	3	0.0140	1	0
3	-0.1759	0.6227	<i>M</i>	2	0.0243	0.3994	1
7	-0.2011	0.6540	<i>BFGS</i>	1	0.0043	0.3994	0
9	-0.2043	0.6597	<i>BFGS</i>	1	0.0021	0.3994	0
13	-0.2068	0.6716	<i>M</i>	1	$6.5891e - 04$	0.3994	0
15	-0.2075	0.6782	<i>BFGS</i>	1	$3.6662e - 04$	0.3994	0
18	-0.2080	0.6870	<i>BFGS</i>	1	$2.1609e - 04$	0.3994	0
23	-0.2082	0.6980	<i>BFGS</i>	1	$1.2821e - 04$	0.3994	0
25	-0.2084	0.7005	<i>BFGS</i>	1	$1.0648e - 04$	0.3994	0
26	-0.2085	0.7008	<i>BFGS</i>	7	$8.7622e - 05$	0.3994	0
29	-0.2086	0.7010	<i>M</i>	28	$1.2858e - 05$	0.3994	0
30	-0.2087	0.7012	<i>M</i>	15	$5.8191e - 06$	0.2500	1

Cuadro 5.10: Comportamiento general del Algoritmo 8. Parámetros: $p = 4$, $g = 0.2$, $M = I$ y $\delta_0 = 1$.

El Cuadro 5.10 recopila la información del desenvolvimiento del Algoritmo 8 con $\delta_0 = 1$ como parámetro de inicialización. Se observa que existe un decrecimiento monótono del valor del funcional J_h así como un incremento en la norma de u^k en cada iteración. En adición, se reporta el tipo de actualización de la matriz B_k para cada iteración. Con respecto al subproceso Bundle, se nota que no requiere de un coste computacional muy alto para converger; se observa que este realiza pocas iteraciones en su mayoría, presenta un incremento de estas en las últimas iteraciones del método general, sin embargo, no son excesivamente grandes como se ha observado en los experimentos previos. Además, se observa que el algoritmo de búsqueda lineal realiza pocas iteraciones a lo largo del método, en la mayoría de iteraciones del método la longitud de paso es constante y cambia en la última iteración.

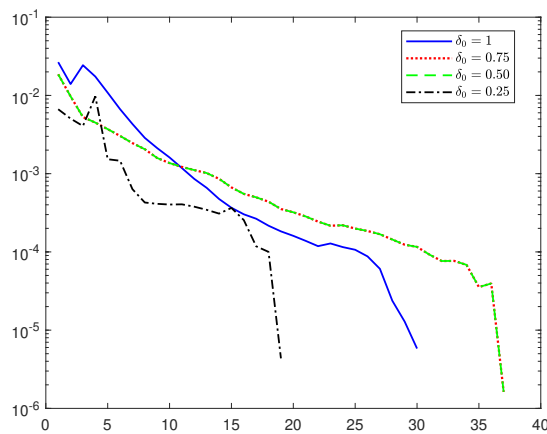


Figura 5.17: Valores de ν calculados por el Algoritmo 8 para cada δ_0 . Parámetros: $p = 4$, $g = 0.2$ y $M = I$.

En la Figura 5.17 se observa la convergencia del método a través de los valores de ν obtenidos en cada iteración. Nótese que para todos los casos, los valores de ν decaen monótonamente en su mayoría. Además, se observa que en las iteraciones finales el descenso de ν es más pronunciado, esto corrobora el orden de convergencia superlineal hacia el mínimo para este tipo de materiales.

Conclusiones y Recomendaciones

Conclusiones

- (I) El estudio de los métodos Bundle y la regularización de Moreau-Yosida de una función convexa no diferenciable permitieron comprender y construir una estrategia innovadora para obtener un método de optimización no diferenciable que converge global y superlinealmente al mínimo.
- (II) En general, el Algoritmo 8 necesita de pocas iteraciones para converger hacia la solución. Sin embargo, observamos que la ejecución total del mismo toma en promedio más de dos horas. Esto es ocasionado por el subalgoritmo Bundle, pues al ser método de primer orden necesita un número mayor de iteraciones en su inicialización.
- (III) Se comprobó tanto teórica como experimentalmente que el algoritmo general converge rápidamente a la solución. De forma particular, en el caso de fluidos espesantes ($p > 2$), el método posee un orden de convergencia superlineal hacia el mínimo. Para el caso de fluidos adelgazantes ($1 < p < 2$) no fue posible establecer un orden de convergencia.
- (IV) En comparación con el método propuesto por Huilgol y You en [15] se constató en cada experimento que el algoritmo Bundle no diferenciable realizó menos iteraciones para converger hacia la solución. Se puede decir que existen indicios de una mayor eficacia del método propuesto.

Recomendaciones

- (I) A futuro, puede ser de interés estudiar otros métodos de optimización no diferenciables con la finalidad de determinar los que se acoplan de mejor manera para la resolución numérica del problema de flujo.
- (II) Para reducir el tiempo de ejecución del algoritmo Bundle no diferenciable se propone reemplazar el subalgoritmo Bundle por otros métodos como *Variable Metric Bundle Method* o *Limited Memory Bundle Method*, pues son más eficientes y generan un menor coste computacional para su convergencia.
- (III) Dado que los experimentos para fluidos adelgazantes fueron muy alentadores, se recomienda estudiar el orden de convergencia del Algoritmo 8; y así obtener un orden de convergencia para cualquier tipo de fluidos de Herschel-Bulkley.
- (IV) Luego de observar que el enfoque “discretizar, luego optimizar” brindó los resultados esperados se propone desarrollar un método similar sobre espacios funcionales de dimensión infinita, es decir, seguir el enfoque “optimizar luego discretizar”.

Referencias

- [1] J. Alberty, C. Carstensen, and S. A. Funken. Remarks around 50 lines of matlab: short finite element implementation. *Numerical Algorithms*, 20(2/3):117–137, 1999.
- [2] M. Badiale and E. Serra. *Semilinear Elliptic Equations for Beginners*. Springer London, 2011.
- [3] A. Bagirov, N. Karmitza, and M. M. Mäkelä. *Introduction to Nonsmooth Optimization*. Springer International Publishing, 2014.
- [4] J.-F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal. *Numerical Optimization*. Springer Berlin Heidelberg, 2006.
- [5] H. Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Springer New York, 2011.
- [6] R. H. Byrd and J. Nocedal. A tool for the analysis of quasi-newton methods with application to unconstrained minimization. *Society for Industrial & Applied Mathematics*, 26(3):727–739, June 1989.
- [7] R. Chiappinelli and D. E. Edmunds. Remarks on surjectivity of gradient operators. *Mathematics*, 8(9), 2020.
- [8] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. Society for Industrial and Applied Mathematics, Jan. 2002.
- [9] J. E. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Society for Industrial and Applied Mathematics, Jan. 1996.
- [10] L. Fan, S. Liu, and S. Gao. Generalized monotonicity and convexity of non-differentiable functions. *Journal of Mathematical Analysis and Applications*, 279(1):276–289, Mar. 2003.

- [11] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.
- [12] R. Glowinski and A. Marroco. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *R.A.I.R.O.*, 9(R2):41–76, 1975.
- [13] S. González-Andrade. A preconditioned descent algorithm for variational inequalities of the second kind involving the p-laplacian operator. *Computational Optimization and Applications*, 66(1):123–162, July 2016.
- [14] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms II*. Springer Berlin Heidelberg, 1993.
- [15] R. Huilgol and Z. You. Application of the augmented lagrangian method to steady pipe flows of bingham, casson and herschel–bulkley fluids. *Journal of Non-Newtonian Fluid Mechanics*, 128(2-3):126–143, July 2005.
- [16] C. Lemaréchal and C. Sagastizábal. Practical aspects of the moreau–yosida regularization: Theoretical preliminaries. *SIAM Journal on Optimization*, 7(2):367–385, May 1997.
- [17] C. Lemaréchal and C. Sagastizábal. Variable metric bundle methods: From conceptual to implementable forms. *Mathematical Programming*, 76(3):393–410, Mar. 1997.
- [18] L. Lukšan and J. Vlček. A bundle-newton method for nonsmooth unconstrained minimization. *Mathematical Programming*, 83(1-3):373–391, 02 1998.
- [19] M. M. Mäkelä. Survey of bundle methods for nonsmooth optimization. *Optimization Methods & Software*, 17:1–29, 02 2002.
- [20] M. M. Mäkelä and P. Neittaanmäki. *Nonsmooth Optimization*. WORLD SCIENTIFIC, 1992.
- [21] R. Mifflin. A quasi-second-order proximal bundle algorithm. *Mathematical Programming*, 73(1):51–72, Apr. 1996.
- [22] R. Mifflin, D. Sun, and L. Qi. Quasi-newton bundle-type methods for nondifferentiable convex optimization. *SIAM Journal on Optimization*, 8(2):583–603, May 1998.

- [23] J. J. Moreau. Proximité et dualité dans un espace hilbertien. *Bulletin de la Société Mathématique de France*, 93:273–299, 1965.
- [24] J. Nocedal and S. Wright. *Numerical Optimization*. Springer New York, 2006.
- [25] J.-S. Pang. Newton's method for b-differentiable equations. *Mathematics of Operations Research*, 15(2):311–341, May 1990.
- [26] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer Berlin Heidelberg, 1994.
- [27] W. Sun and Y. X. Yuan. *Optimization Theory and Methods*. Kluwer Academic Publishers, 2006.