

Un algoritmo cuasi Newton inexacto global para problemas de complementariedad no lineal

Carlos Andrés Arias Torres

Universidad del Valle
Facultad de Ciencias Naturales y Exactas
Departamento de Matemáticas
Doctorado en Ciencias Matemáticas
Santiago de Cali
2018

Un algoritmo cuasi Newton inexacto global para problemas de complementariedad no lineal

Carlos Andrés Arias Torres

Trabajo de investigación presentado como requisito parcial para optar al
título de Doctor en Ciencias Matemáticas

Director
Dr. Héctor Jairo Martínez Romero
Profesor de la Universidad del Valle

Codirectora
Dra. Rosana Pérez Mera
Profesora de la Universidad del Cauca

Universidad del Valle
Facultad de Ciencias Naturales y Exactas
Departamento de Matemáticas
Doctorado en Ciencias Matemáticas
Santiago de Cali
2018

TABLA DE CONTENIDOS

Tabla de Contenidos	I
Índice de tablas	III
1. Introducción	1
2. Preliminares	5
2.1. Familia de problemas de complementariedad	7
2.1.1. Problema de Complementariedad Lineal	7
2.1.2. Problema de Complementariedad mixto	8
2.1.3. Problema de Complementariedad Horizontal	8
2.1.4. Problema de Complementariedad Vertical	9
2.1.5. Desigualdades Variacionales	10

2.1.6. Problema de Complementariedad Generalizado	10
2.2. Reformulación y solución del PCNL	11
3. Algoritmo cuasi <i>Newton</i> inexacto y teoría de convergencia	18
4. Experimentos numéricos	37
4.1. Funciones de prueba	39
4.2. Valores Propios Complementarios	47
4.3. Equilibrio económico	51
5. Sistemas de ecuaciones con restricciones	58
5.1. Teoría de convergencia	60
5.2. Experimentos numéricos	62
5.2.1. Equilibrio Químico	63
5.2.2. Ecuación H de <i>Chandrasekhar</i>	66
6. Comentarios finales	71
7. Apéndice	73
Bibliografía	77

ÍNDICE DE TABLAS

4.1. Algoritmos <i>CNIDP</i> , CNDP, PGIN y NIDP para el PCNL(F) 4.2.	40
4.2. Algoritmo <i>CNIDP</i> con $n = 10000$ y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$	41
4.3. Algoritmo <i>CNIDP</i> , Act. BM y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$	42
4.4. Algoritmo <i>CNIDP</i> , Act. BB y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$	43
4.5. Variación de $\mathbf{z}_0$, $n = 1000$	44
4.6. Algoritmos <i>CNIDP</i> , CNDP, PGIN y NIDP para el PCNL.	45
4.7. Algoritmo <i>CNIDP</i> con $n = 200$ y $\mathbf{z}_0 = 50 \times (1, 1, \dots, 1)^T$	46
4.8. Análisis de globalidad del algoritmo <i>CNIDP</i>	47
4.9. Resultados grupo A1.	49
4.10. Resultados grupo A2.	50
4.11. Notación <i>spatial price problem</i>	51

4.12. Parámetros para la función de costos.	54
4.13. Parámetros para las funciones de suministros y demanda.	54
4.14. Parámetros para las funciones de suministros y demanda.	55
4.15. Variación de $\mathbf{z}_0$. Un producto por distribuir.	55
4.16. Variación de $\mathbf{z}_0$. Dos productos por distribuir.	56
5.1. Constantes de equilibrio.	64
5.2. Algoritmos $CNIDP^+$, CNDP y DP para el problema de equilibrio químico.	65
5.3. Algoritmo $CNIDP^+$ con $\mathbf{z}_0 = 10 \times (1, 1, \dots, 1)^T$	66
5.4. Algoritmos $CNIDP^+$, CNDP y DP para la ecuación de H	68
5.5. Algoritmos $CNIDP^+$, CNDP y DP para la ecuación H . $n = 1000$	69
5.6. Algoritmo $CNIDP^+$ con $n = 1000$ y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$	69

Publicaciones basadas en esta investigación

Artículos

Artículos publicados

- C. A. Arias and J. M. Martínez, *Fast convergence of an inexact interior point method for horizontal complementarity problems*, Numerical Algorithms. <https://doi.org/10.1007/s11075-018-0480-8>

Artículos para ser sometidos

- C. A. Arias and H. J. Martínez, and R. Pérez, *Global inexact quasi-Newton method for nonlinear system of equations with constraints*. To be submitted.
- C. A. Arias and H. J. Martínez, and R. Pérez, *A new global inexact quasi-Newton method for complementarity problems*. To be submitted.

Ponencias

- Convergencia global del método cuasi Newton inexacto. Presentado en el V Latin American Workshop on Optimization and Control (Julio 5 al 8 de 2016, Tandil, Argentina).
- Inexact quasi-Newton interior point method for complementarity problems. Presentado en el III International conference on Applied Mathematics and Informatics (Noviembre 26 a Diciembre 1 de 2017, San Andrés, Colombia).

CAPÍTULO 1

INTRODUCCIÓN

Sea $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$, $F(\mathbf{x}) = (F_1(\mathbf{x}), F_2(\mathbf{x}), \dots, F_n(\mathbf{x}))^T$, donde $F_i : \mathbb{R}^n \longrightarrow \mathbb{R}$, una función no lineal y continuamente diferenciable. *El problema de complementariedad no lineal*, PCNL(F), consiste en hallar un vector $\mathbf{x}^* \in \mathbb{R}^n$ tal que

$$\mathbf{x}^* \geq 0 ; F(\mathbf{x}^*) \geq 0 ; F(\mathbf{x}^*)^T \mathbf{x}^* = 0, \quad (1.1)$$

donde la expresión $\mathbf{y} \geq 0$, para un vector $\mathbf{y} \in \mathbb{R}^n$, significa que todas sus componentes son no negativas.

Como se puede ver, (1.1) implica que, para todo $i = 1, 2, \dots, n$, $x_i F_i(\mathbf{x}) = 0$ y por tanto, que $x_i = 0$ o $F_i(\mathbf{x}) = 0$, siendo esta la razón por la cual el problema recibe el calificativo de *complementariedad*. Esta condición trae consigo implícitamente la búsqueda de un equilibrio entre la *variable* del problema y el valor que toma la función que define el problema en dicha variable. Así, el concepto de *complementariedad* es un sinónimo del concepto de *sistema en equilibrio* [29].

Por lo mencionado anteriormente, *problemas de complementariedad no lineal* surgen de forma natural en campos como la ingeniería y la física; por ejemplo, cuando se presentan situaciones de contacto mecánico [29], equilibrio del tráfico [63] y problemas de lubricación elasto-hidrodinámicos [44], así como también en el campo de la economía,

particularmente cuando se busca un equilibrio en los sistemas económicos [29].

Por lo anterior, ha sido de gran interés para la comunidad matemática el estudio de diversos métodos que permitan solucionar *problemas de complementariedad no lineal*, entre ellos, *métodos de homotopía*, que son derivados de métodos de punto fijo [23, 64]; *métodos basados en redes neuronales* [47, 48, 66], *método de ecuaciones no diferenciables* [21, 20] y *métodos suavizados* [13].

En general, los últimos dos métodos mencionados anteriormente, el método de ecuaciones no diferenciables y los métodos suavizados, se basan en la solución de un sistema de ecuaciones (diferenciable o no diferenciable) equivalente al PCNL(F), por tal razón, es natural pensar en la implementación de métodos tipo *Newton* o *Cuasi Newton* para la solución de estos.

Como es de esperarse, el método de ecuaciones no diferenciables consiste en reescribir el problema (1.1) como un sistema de ecuaciones no diferenciable, lo que conlleva al uso de herramientas como el *Jacobiano Generalizado* [14, 15, 16] para poder implementar métodos tipo Newton o cuasi Newton. Por otro lado, los métodos suavizados consisten en resolver una sucesión de problemas (sistemas de ecuaciones) diferenciables que aproximan al PCNL(F).

Hasta el momento, se han propuesto métodos tipo Newton exactos [13, 41, 46] e inexactos [27, 40, 58, 62] para resolver el problema (1.1) mediante sistemas de ecuaciones diferenciables y no diferenciables, los cuales, en general, bajo algunas condiciones de regularidad, convergen cuadráticamente a la solución del problema en el caso de los métodos exactos y convergen superlinealmente a la solución del problema en el caso de los métodos inexactos. Cabe mencionar que si bien esta estrategia se ha mostrado exitosa tanto en su versión exacta como en su versión inexacta, el hecho de tener que recurrir a la diferenciación generalizada y trabajar con los elementos del jacobiano generalizado hace que esta clase de métodos sean poco eficientes en término de tiempo de ejecución.

De la misma manera, se han propuesto métodos cuasi Newton exactos [4, 6, 11, 50] para resolver las respectivas reformulaciones del PCNL(F), los cuales, bajo condiciones de regularidad similares a las exigidas para los métodos tipo Newton, presentan una convergencia superlineal a la solución del problema. En este caso, en cada iteración de los métodos propuestos, se realiza una aproximación a algún elemento del jacobiano generalizado, lo cual los convierte, en la práctica, en métodos más eficientes en cuestiones de tiempo de ejecución. Lastimosamente, el tiempo ahorrado a través de esta estrategia no se ve reflejado cuando se intentan resolver problemas de gran tamaño ya que en cada iteración será necesario resolver, de forma exacta, un sistema de ecuaciones lineales de

gran tamaño.

Teniendo en cuenta lo anterior y con el ánimo de proponer un método eficiente (en términos de tiempo) que permita resolver Problemas de Complementariedad no Lineal de gran tamaño, sin tener que recurrir a la diferenciación generalizada, en este trabajo de investigación, proponemos y desarrollamos un método cuasi Newton inexacto global para resolver el problema (1.1) de una manera indirecta: en primer lugar, reescribiremos el PCNL(F) como un Problema de Complementariedad Horizontal (PCH) [2, 7] y posteriormente, reescribiremos el PCH como un problema de minimización. Cabe destacar que abordar el PCNL(F) de esta manera nos permitirá trabajar con reformulaciones diferenciables de la versión original del problema.

Por otro lado, debemos destacar que el método que proponemos en esta investigación, bajo una leve modificación, también es capaz de encontrar, de una manera eficiente, las raíces positivas de sistemas de ecuaciones no lineales de gran tamaño (problemas con más de mil ecuaciones e incógnitas). Ahora bien, teniendo en cuenta que este es un problema presente en muchos campos de aplicación como la química, la física nuclear y las matemáticas, hemos decidido presentar en este documento, de manera adicional, los resultados obtenidos en esta dirección.

Finalmente, debemos decir que los avances teóricos presentados en esta investigación justifican el uso de los métodos computacionales para la resolución de problemas como los aquí abordados. Es de amplio conocimiento que el método de Newton, teóricamente se muestra como el método más eficaz; sin embargo, se debe tener en cuenta que dada la precisión con la que trabajan las máquinas, en la mayoría de los casos de interés, nos tendremos que conformar con muy buenas aproximaciones tanto a las derivadas requeridas como a la solución de los sistemas de ecuaciones involucrados. Así, el desarrollo teórico de los métodos cuasi Newton y los métodos inexactos se convierten en justificaciones teóricas del uso de las máquinas en la resolución de problemas matemáticos.

El documento está organizado de la siguiente forma: en el **Capítulo 2**, mencionamos algunos hechos histórico relevantes en el desarrollo de la teoría de complementariedad. De igual manera, describimos algunos de los problemas más reconocidos de la *familia de problemas de complementariedad* y que se encuentran estrechamente relacionados con el PCNL(F). En este mismo capítulo, presentamos, de forma concisa, las estrategias de resolución mencionadas anteriormente; reformulación vía sistemas de ecuaciones no diferenciables y métodos de suavización. En el **Capítulo 3**, proponemos un *Algoritmo cuasi Newton inexacto global* para resolver el PCNL(F) a través de su reformulación como un PCH y mediante la solución de un problema de optimización. De igual manera, en este capítulo, desarrollamos la teoría de convergencia global del algoritmo propuesto.

En la primera parte del **Capítulo 4**, presentamos los resultados numéricos obtenidos al resolver con el algoritmo cuasi Newton inexacto global propuesto, algunos *problemas de complementariedad no lineal* que son reconocidos para probar algoritmos y comparamos estos resultados con los obtenidos al resolver los mismos problemas con el algoritmo tipo Newton inexacto propuesto recientemente en [2]. En la segunda parte, analizamos numéricamente detalles del desempeño de nuestro algoritmo al resolver un problema emergente en las matemáticas y un problema de aplicación en economía. En el **Capítulo 5**, extendemos el algoritmo propuesto en el **Capítulo 3** para hallar las raíces no negativas de un sistema de ecuaciones. Realizamos algunos comentarios con respecto a su teoría de convergencia y realizamos algunos ensayos numéricos con dos problemas de aplicación: uno en el campo de la química y otro en el campo de la física nuclear. Por último, en el **Capítulo 6**, hacemos algunos comentarios finales y una propuesta para futuros trabajos en el tema.

CAPÍTULO 2

PRELIMINARES

Relacionados con el PCNL(F) (1.1) se encuentran los Problemas de Complementariedad Lineal (PCL) [18, 19], Problemas de Complementariedad mixtos (PCm) [24, 29], Problemas de Complementariedad Horizontal (PCH) [2, 7], Problemas de Complementariedad Vertical (PCV) [8, 29] y Desigualdades Variacionales (DV) [28, 35]. Esta clase de problemas se enmarcan dentro de una gran familia de problemas conocidos como *problemas de complementariedad*, algunos de los cuales son descritos brevemente más adelante en este capítulo.

En general, hablar de *problemas de complementariedad* nos conlleva inmediatamente a pensar en su principal característica: las condiciones de complementariedad. Dichas condiciones, al igual que en (1.1), exigen que el producto de dos o más cantidades no negativas sea cero. En este contexto, cada una de las cantidades es una variable de decisión o una función que depende de las variables de decisión.

Las condiciones de complementariedad hicieron su aparición por primera vez cuando William Karush, en su tesis de maestría en 1939, derivó las condiciones de optimalidad para problemas de programación no lineal (PPNL) con restricciones de desigualdad. En

efecto, si consideramos el PPNL

$$\begin{array}{ll} \textbf{Minimizar} & f(x) \\ \text{s.a} & h_i(x) = 0 \quad \forall i \in \mathcal{E} \\ & h_i(x) \geq 0 \quad \forall i \in \mathcal{I} \end{array}$$

entonces, bajo algunas condiciones de regularidad de las funciones involucradas en el problema, se puede garantizar que si $\mathbf{x}^*$ es una solución local del PPNL entonces existen multiplicadores $\lambda_i^* \geq 0$ tal que

$$\begin{aligned} \nabla_x \mathcal{L}(\mathbf{x}^*, \lambda^*) &= 0, \\ h_i(\mathbf{x}^*) &= 0 \quad \forall i \in \mathcal{E}, \\ h_i(\mathbf{x}^*) &\geq 0 \quad \forall i \in \mathcal{I}, \\ \lambda_i^* &\geq 0 \quad \forall i \in \mathcal{E} \cup \mathcal{I}, \\ \lambda_i^* h_i(\mathbf{x}^*) &= 0 \quad \forall i \in \mathcal{E} \cup \mathcal{I}, \end{aligned} \tag{2.1}$$

donde $\mathcal{L}(\mathbf{x}, \lambda)$ es la respectiva función lagrangiana del PPNL [53]. Como podemos ver, la condición (2.1) es la condición de complementariedad que deben satisfacer los multiplicadores y las restricciones del PPNL. Cabe mencionar que las anteriores condiciones son conocidas como las condiciones necesarias de primer orden de optimalidad para el PPNL o simplemente condiciones KKT (Karush-Kuhn-Tucker).

Sin embargo, debemos resaltar que las condiciones de complementariedad van mucho más allá, ya que ellas aparecen en el estudio de problemas de equilibrio y surgen naturalmente en problemas provenientes de la economía, la ingeniería y las ciencias, razón por la cual el estudio de esta área ha sido de gran interés durante los últimos 60 años para la comunidad matemática.

La principal motivación para el estudio de Problemas de Complementariedad Lineal se debió a que las condiciones KKT para problemas de programación lineal y cuadrática constituyen en sí mismas un PCL o un PCL mixto [19]. Sin embargo, el estudio de PCL no fue de gran interés para la comunidad matemática hasta que, en 1963 J. T. Howson, en su tesis doctoral, y Lemke y Howson en [45] mostraron que el problema de hallar el *punto de equilibrio de Nash* de un juego *bimatrix* es equivalente a resolver un PCL y desarrollaron un ingenioso y eficiente método para resolver el problema; este método es conocido como el algoritmo del pivote complementario [18]. En 1968, la unificación de los problemas de programación lineal y cuadrática, y los juegos *bimatrix* bajo el formato de PCL realizado por Cottle y Dantzing en [18] fue visto como un gran avance en el campo y, “el estudio de los problemas de complementariedad de repente floreció” [8].

El Problema de Complementariedad No Lineal fue introducido por primera vez en 1964 por *Cottle* en su tesis doctoral y su estrecha relación con los problemas de desigualdades variacionales fue presentada por *Hartman* y *Stampacchia* en 1966 con el objetivo de calcular puntos estacionarios de PPNL. Cabe mencionar que si bien el PCNL y los problemas de desigualdades variacionales fueron presentados poco después de la aparición de los problemas de complementariedad lineal, no fue sino hasta principio de los años 70 que los algoritmos para resolver esta clase de problemas hicieron su aparición. Desde ese instante y hasta el momento, miles de investigaciones, con distintos métodos y estrategias para resolver problemas de complementariedad, han sido publicadas [20, 21, 23, 47, 48, 64, 66].

Teniendo en cuenta que el PCNL(F) es el principal objetivo de esta investigación, en la siguiente sección mencionaremos y estableceremos la relación entre algunos de los más importantes problemas de la familia de problemas de complementariedad y el Problema de Complementariedad no Lineal, los cuales serán descritos en mayor detalle, a modo de apéndice, al final de este documento. Posteriormente, presentaremos una breve descripción de cómo reescribir el PCNL(F) como un sistema de ecuaciones no lineales (diferenciable o no diferenciable) y algunas de las estrategias más destacadas para resolver problemas de complementariedad por esta vía.

2.1. Familia de problemas de complementariedad

2.1.1. Problema de Complementariedad Lineal

Dados $M \in \mathbb{R}^{n \times n}$ y $\mathbf{q} \in \mathbb{R}^n$, el Problema de Complementariedad Lineal PCL($M, \mathbf{q}$) consiste en hallar $\mathbf{x} \in \mathbb{R}^n$ tal que

$$M\mathbf{x} + \mathbf{q} \geq 0, \quad \mathbf{x} \geq 0 \quad \text{y} \quad \mathbf{x}^T(M\mathbf{x} + \mathbf{q}) = 0.$$

El caso en el que $\mathbf{q} = \mathbf{0}$ es llamado *Problema de Complementariedad Homogéneo* y tiene la particularidad de que si $\mathbf{x}^*$ es una solución del PCL($M, 0$) entonces, para cualquier escalar $\lambda > 0$, $\lambda \mathbf{x}^*$ también es solución del PCL($M, 0$). Es claro que el PCNL(F) es equivalente al PCL en el caso en que F es una función afín.

2.1.2. Problema de Complementariedad mixto

Sean $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ una función diferenciable, $\mathbf{l} = (l_1, l_2, \dots, l_n)$, y $\mathbf{u} = (u_1, u_2, \dots, u_n)$ con $l_i, u_i \in \overline{\mathbb{R}}$, $i = 1, 2, \dots, n$, donde $\overline{\mathbb{R}} = \mathbb{R} \cup \{+\infty, -\infty\}$. El Problema de Complementariedad mixto, $\text{PCm}(F)$, consiste en hallar vectores $\mathbf{x}, \mathbf{w}, \mathbf{v} \in \mathbb{R}^n$ tal que

$$F(\mathbf{x}) - \mathbf{w} + \mathbf{v} = 0 \quad (2.2)$$

$$\mathbf{l} \leq \mathbf{x} \leq \mathbf{u} \quad (2.3)$$

$$(\mathbf{x} - \mathbf{l})^T \mathbf{w} = 0 \quad (2.4)$$

$$(\mathbf{u} - \mathbf{x})^T \mathbf{v} = 0 \quad (2.5)$$

$$\mathbf{w}, \mathbf{v} \geq 0. \quad (2.6)$$

Vale la pena destacar que la desigualdad (2.3) significa que $l_i \leq x_i \leq u_i$ para cada $i = 1, 2, \dots, n$. Observe que si para todo i , $l_i = -\infty$ y $u_i = \infty$ entonces (2.4) y (2.5) implican que $\mathbf{w} = \mathbf{v} = 0$, por lo tanto, el $\text{PCm}(F)$ se transforma en un sistema de ecuaciones no lineales. Quizá esto muestra por qué muchas de las técnicas desarrolladas para resolver problemas de complementariedad se basan en reformular el problema como un sistema de ecuaciones.

Otro caso interesante sucede cuando $l_i = 0$ y $u_i = \infty$. En este caso, el $\text{PCm}(F)$ se convierte en un $\text{PCNL}(F)$. Si F es una función afín entonces el $\text{PCm}(F)$ se convierte en un $\text{PCL}(F)$.

2.1.3. Problema de Complementariedad Horizontal

Sea $H: \mathbb{R}^{n+m+n} \rightarrow \mathbb{R}^{n+m}$ una función continuamente diferenciable. El Problema de Complementariedad Horizontal, $\text{PCH}(H)$, consiste en hallar $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$, y $\mathbf{w} \in \mathbb{R}^n$ tal que

$$H(\mathbf{x}, \mathbf{y}, \mathbf{w}) = 0,$$

$$\mathbf{x}^T \mathbf{w} = 0,$$

$$\mathbf{x}, \mathbf{w} \geq 0.$$

En particular, observemos que si $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ es una función continuamente diferenciable y definimos la función $H(\mathbf{x}, \mathbf{w}) = F(\mathbf{x}) - \mathbf{w}$ entonces el PCNL(F) es equivalente al PCH(H). Es decir, todo Problema de Complementariedad no Lineal se puede ver como un problema de Complementariedad Horizontal.

Observe que el PCH(H) no exige ninguna condición especial sobre la variable $\mathbf{y}$ y por tanto, se considera una variable libre en el problema, es más, en muchos casos esta variable podría no existir, como aquel caso mencionado en el párrafo anterior.

2.1.4. Problema de Complementariedad Vertical

Si $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ es continuamente diferenciable, es fácil probar que el PCNL(F) es equivalente a resolver el problema de minimización por componentes que consiste en hallar $\mathbf{x} \in \mathbb{R}^n$ tal que

$$\min(\mathbf{x}, F(\mathbf{x})) = 0.$$

Observemos que en esta formulación intervienen dos funciones, una de ellas es la función identidad y la otra es F . Sin embargo, existen problemas en los que su formulación conlleva a resolver el problema de hallar $\mathbf{x} \in \mathbb{R}^n$ tal que

$$\min(F^1(\mathbf{x}), F^2(\mathbf{x})) = 0,$$

donde $F^1, F^2: \mathbb{R}^n \rightarrow \mathbb{R}^n$ son funciones continuamente diferenciables. Por supuesto que este problema se puede extender a una cantidad finita de funciones y este problema es el que se conoce como Problema de Complementariedad Vertical (PCV). Matemáticamente, el problema consiste en hallar $\mathbf{x} \in \mathbb{R}^n$ tal que

$$\min(F^1(\mathbf{x}), F^2(\mathbf{x}), \dots, F^k(\mathbf{x})) = 0,$$

donde $F^j: \mathbb{R}^n \rightarrow \mathbb{R}^n$, para $j = 1, 2, \dots, k$.

Claramente, si $\mathbf{x}^*$ es una solución del PCV entonces $F_i^j(\mathbf{x}^*) \geq 0$, para todo $i = 1, 2, \dots, n$ y $j = 1, 2, \dots, m$. Además, se puede asegurar que para cada índice i existe por lo menos un índice j tal que $F_i^j(\mathbf{x}^*) = 0$, de donde, para cada $i = 1, 2, \dots, n$

$$\prod_{j=1}^k F_i^j(\mathbf{x}^*) = 0.$$

2.1.5. Desigualdades Variacionales

Sean X un subconjunto no vacío, cerrado y convexo de $\mathbb{R}^n$ y $F: X \rightarrow \mathbb{R}^n$ una función diferenciable. El Problema de Desigualdades Variacionales, $DV(X, F)$, consiste en hallar un vector $\mathbf{x} \in X$ tal que

$$F(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) \geq 0, \quad \text{para todo } \mathbf{y} \in X.$$

En muchas aplicaciones, X es un poliedro de $\mathbb{R}^n$. En particular, si $X = \mathbb{R}_+^n$, donde $\mathbb{R}_+^n$ es el octante no negativo de $\mathbb{R}^n$, entonces, el $DV(X, F)$ es equivalente al PCNL(F). En el apéndice, al final de este documento, ampliamos los detalles que demuestran tal equivalencia.

2.1.6. Problema de Complementariedad Generalizado

Sean $F, G: \mathbb{R}^n \rightarrow \mathbb{R}^m$ funciones continuamente diferenciables y $\mathcal{K}$ un cono convexo y cerrado en $\mathbb{R}^m$ entonces el Problema de Complementariedad no Lineal Generalizado $PCG(F, G, \mathcal{K})$ consiste en hallar un vector $\mathbf{x}^* \in \mathbb{R}^n$ tal que

$$F(\mathbf{x}^*) \in \mathcal{K}, \quad G(\mathbf{x}^*) \in \mathcal{K}^\circ, \quad F(\mathbf{x}^*)^T G(\mathbf{x}^*) = 0,$$

donde $\mathcal{K}^\circ$ representa el cono polar de $\mathcal{K}$; es decir,

$$\mathcal{K}^\circ = \{\mathbf{y} \in \mathbb{R}^m : \mathbf{w}^T \mathbf{y} \geq 0 \quad \forall \mathbf{w} \in \mathcal{K}\}.$$

En particular, si $m = n$, $\mathcal{K} = \mathbb{R}_+^n$ y G es la función identidad entonces el $PCG(F, G, \mathcal{K})$ se transforma en el PCNL(F).

Como podemos ver, diseñar estrategias para resolver Problemas de Complementariedad Horizontal, Vertical, Desigualdades Variacionales o el Problema de Complementariedad Generalizado conlleva inmediatamente a tener una estrategia para resolver el PCNL(F) (1.1). Por esta razón, en esta investigación, optamos por diseñar una estrategia para solucionar Problemas de Complementariedad Horizontal y con esto, no solo abordamos el PCNL(F) sino un conjunto más grande de problemas.

En la siguiente sección, mostraremos cómo reformular el PCNL(F) como un sistema de ecuaciones diferenciables y no diferenciables y sus respectivas estrategias de solución.

De igual forma, presentaremos las propuestas realizadas recientemente en [2] y [7] para resolver Problemas de Complementariedad Horizontal y con ello el PCNL(F).

2.2. Reformulación y solución del PCNL

En general, el método de ecuaciones no diferenciables y los métodos suavizados para resolver el problema (1.1) se basan en la reformulación del PCNL(F). Para este fin, es necesario el uso de una función $\varphi : \mathbb{R}^2 \longrightarrow \mathbb{R}$ tal que

$$\varphi(a, b) = 0 \iff a \geq 0, \quad b \geq 0, \quad ab = 0, \quad (2.7)$$

la cual es conocida como *función de complementariedad* [41, 65].

Geométricamente, a partir de (2.7), se infiere que la traza de la función φ obtenida por la intersección con el plano xy es la curva formada por los semiejes positivos x y y , la cual no es diferenciable en $(0, 0)$, razón por la que φ no es diferenciable [4].

Dos de las *funciones de complementariedad* más utilizadas en la solución de PCNL(F) son la *función mínimo* [55, 59] y la *función de Fischer-Burmeister* [30, 31], definidas respectivamente por

$$\varphi(a, b) = \min\{a, b\} \qquad \varphi(a, b) = \sqrt{a^2 + b^2} - a - b.$$

En 1998, *Kanzow y Kleinmichel* [41] presentaron una familia de funciones de complementariedad φ_λ dependientes de un parámetro $\lambda \in (0, 4)$, definida por

$$\varphi_\lambda(a, b) = \sqrt{(a - b)^2 + \lambda ab} - a - b. \quad (2.8)$$

Observemos que esta familia de funciones incluye la *función de Fischer-Burmeister* como un caso particular, cuando $\lambda = 2$, y converge a un múltiplo de la *función mínimo* cuando λ tiende a cero.

Ahora bien, si definimos la función $\Phi : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ por

$$\Phi(\mathbf{x}) = \begin{pmatrix} \varphi(x_1, F_1(\mathbf{x})) \\ \vdots \\ \varphi(x_n, F_n(\mathbf{x})) \end{pmatrix}, \quad (2.9)$$

donde $\varphi(a, b)$ es cualquier función de complementariedad entonces, un vector $\mathbf{x}^*$ es

solución del PCNL(F), si y solo si, $\mathbf{x}^*$ es solución del sistema de ecuaciones

$$\Phi(\mathbf{x}) = 0. \quad (2.10)$$

Como consecuencia de la no diferenciabilidad de φ , el sistema de ecuaciones no lineales (2.10) es no diferenciable. De esta forma, el PCNL(F) es reformulado como un sistema de ecuaciones no lineales, no diferenciable, por lo tanto, si se piensa en la implementación de algún método tipo Newton o cuasi Newton para la solución de este, es necesario recurrir al concepto de *jacobiano generalizado*.

En 1973, *Frank H. Clarke*, en su tesis doctoral [14] y algunos de sus artículos [15, 16], expuso lo que hoy en día llamamos *jacobiano generalizado*, el cual se define, para una función no necesariamente diferenciable $G : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ y *Lipschitz continua*¹, como el conjunto

$$\partial G(\mathbf{x}) = \text{conv} \left\{ \lim_{k \rightarrow \infty} G'(\mathbf{x}_k) \in \mathbb{R}^{n \times n} : \lim_{k \rightarrow \infty} \mathbf{x}_k = \mathbf{x}, \mathbf{x}_k \in D_G \right\},$$

donde D_G es el conjunto de todos los vectores de $\mathbb{R}^n$ en los que G es diferenciable y $\text{conv}\{A\}$ representa la envolvente convexa del conjunto A .

Ahora bien, con el ánimo de proponer algoritmos globalizados [22] que permitan la solución del PCNL(F) (1.1), la comunidad científica recurre a una clase especial de funciones llamadas *funciones de mérito* [53]. La *función de mérito* más utilizada en el caso de sistemas de ecuaciones no lineales de la forma $G(\mathbf{x}) = 0$, es la función f definida por $f(\mathbf{x}) = \frac{1}{2} \|G(\mathbf{x})\|_2^2$.

Haciendo uso de esta función de mérito, y bajo algunas hipótesis razonables, se puede garantizar que todo minimizador global del problema de minimización

$$\begin{aligned} &\textbf{Minimizar} && f(\mathbf{x}) \\ &&& \mathbf{x} \in \mathbb{R}^n \end{aligned} \quad (2.11)$$

es una solución del sistema de ecuaciones $G(\mathbf{x}) = 0$.

Cabe destacar que toda solución del sistema de ecuaciones $G(\mathbf{x}) = 0$ es un *minimizador* de (2.11), sin embargo, pueden existir soluciones locales de (2.11) que no son soluciones

¹Una función $G : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ es *Lipschitz continua* si, y solo si, existe una constante positiva K tal que

$$\|G(\mathbf{x}) - G(\mathbf{y})\|_2 \leq K \|\mathbf{x} - \mathbf{y}\|_2$$

para todo $\mathbf{x}$ y $\mathbf{y} \in \mathbb{R}^n$

del sistema de ecuaciones no lineales asociado.

Teniendo en cuenta lo anterior, con el fin de *globalizar* algún método tipo Newton o cuasi Newton que busque hallar una solución a (2.10), recurriendo al *jacobiano generalizado* de Φ , se puede hacer uso de la *función de mérito* asociada a (2.10), $\Psi : \mathbb{R}^n \rightarrow \mathbb{R}$, definida por

$$\Psi(x) = \frac{1}{2}\Phi(x)^T\Phi(x) = \frac{1}{2}\|\Phi(x)\|_2^2. \quad (2.12)$$

Con esta nueva función, se puede reformular nuevamente el sistema de ecuaciones no lineales, no diferenciables (2.10) como el *problema de minimización*

$$\begin{aligned} &\textbf{Minimizar} \quad \Psi(\mathbf{x}). \\ &\mathbf{x} \in \mathbb{R}^n \end{aligned} \quad (2.13)$$

En [41], *Kanzow y Kleinmichel* prueban que, si se usa la familia de funciones de complementariedad φ_λ dada en (2.8) para reformular el PCNL(F) (1.1), entonces la función de mérito resultante Ψ es continuamente diferenciable y para cualquier matriz $H \in \partial\Phi(\mathbf{x})$, se tiene que $\nabla\Psi(\mathbf{x}) = H^T\Phi(\mathbf{x})$. Además, sugieren reescribir la *función de mérito* Ψ como

$$\Psi(\mathbf{x}) = \sum_{i=1}^n \psi_\lambda(x_i, F_i(\mathbf{x})),$$

donde $\psi_\lambda : \mathbb{R}^n \rightarrow \mathbb{R}$ se define como

$$\psi_\lambda(a, b) = \frac{1}{2}(\varphi_\lambda(a, b))^2 = \frac{1}{2} \left(\sqrt{(a-b)^2 + \lambda ab} - a - b \right)^2,$$

con el fin de demostrar que una condición suficiente para que un vector $\mathbf{x}^* \in \mathbb{R}^n$ sea solución del PCNL(F) es que $\mathbf{x}^*$ sea un punto estacionario de Ψ y que el jacobiano, $F'(\mathbf{x}^*)$, sea una *matriz* P_0 ².

En general, los algoritmos tipo Newton o cuasi Newton exactos y globalizados para resolver (1.1) mediante el problema de minimización (2.13) tienen la siguiente estructura [5, 6, 41]

²Una matriz $M \in \mathbb{R}^{n \times n}$ es una matriz P_0 , si para cualquier vector no nulo $\mathbf{y} \in \mathbb{R}^n$, existe un índice $i_0 = i_0(\mathbf{y}) \in \{1, 2, \dots, n\}$, tal que $y_{i_0} \neq 0$ y $y_{i_0}[M\mathbf{y}]_{i_0} \geq 0$.

Algoritmo 2.1. Algoritmos tipo Newton y cuasi Newton globalizados*P.0 Inicialización.**P.1 Criterio de parada.**P.2 Búsqueda direccional: Calcule $\mathbf{d}_k$ tal que*

$$\Phi(\mathbf{x}_k) + M(\mathbf{x}_k)\mathbf{d}_k = 0, \quad (2.14)$$

donde $M(\mathbf{x})$ es una matriz en el jacobiano generalizado de Φ evaluada en $\mathbf{x}$ para los métodos tipo Newton o una aproximación a esta para los métodos cuasi Newton.

*P.3 Búsqueda lineal.**P.4 Actualización de parámetros.*

Cabe mencionar que los algoritmos tipo Newton inexactos para resolver el problema (1.1) mediante la reformulación dada en (2.13) difieren de los métodos exactos en el sentido en que la dirección de descenso calculada en **P.2** se halla de forma aproximada; es decir, dado un parámetro $\theta_k \in [0, 1)$, la idea es encontrar una dirección $\mathbf{d}_k$ tal que

$$\|\Phi(\mathbf{x}_k) + M(\mathbf{x}_k)\mathbf{d}_k\| \leq \theta_k \|\Phi(\mathbf{x}_k)\|,$$

donde $M(\mathbf{x}_k)$ es una matriz en el jacobiano generalizado de Φ en $\mathbf{x}_k$. Es importante resaltar que hasta el momento no se han propuesto métodos cuasi Newton inexactos para resolver el PCNL(F) a través de esta reformulación.

Ahora bien, con el fin de reformular (aproximar) el PCNL(F) (1.1) como un sistema de ecuaciones diferenciable, se han propuesto diversos métodos de suavización de las funciones de complementariedad [13, 46, 56], los cuales consisten en introducir una variable unidimensional positiva al problema (2.10) y propender porque esta tienda a cero.

Por ejemplo, si se define $\xi : \mathbb{R}^2 \rightarrow \mathbb{R}$ como

$$\xi(\omega, \epsilon) = \frac{\sqrt{\omega^2 + 4\epsilon^2} + \omega}{2}$$

entonces una aproximación suave de la función mínimo [56] es

$$\phi(a, b, \epsilon) = a - \xi(a - b, \epsilon) \quad (2.15)$$

y una suavización de la función de Fisher-Burmeister [13] se puede definir como

$$\phi(a, b, \epsilon) = \frac{1}{2} \left(a + b - \sqrt{a^2 + b^2 + 2\epsilon^2} \right) \quad (2.16)$$

de tal manera que el sistema de ecuaciones

$$\tilde{\Phi}(\mathbf{x}, \epsilon) = \begin{pmatrix} \phi(x_1, F_1(\mathbf{x}), \epsilon) \\ \vdots \\ \phi(x_n, F(\mathbf{x}_n), \epsilon) \end{pmatrix} \quad (2.17)$$

es un sistema de ecuaciones diferenciable para todo $\epsilon > 0$. Claramente, lo que se desea a través de este enfoque es que

$$\lim_{\epsilon \rightarrow 0} \|\tilde{\Phi}(\mathbf{x}, \epsilon) - \Phi(\mathbf{x})\| = 0.$$

Similar al caso no diferenciable, si se considera la función escalar $\Psi_\epsilon : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$ dada por $\Psi_\epsilon(\mathbf{x}) = \frac{1}{2} \|\tilde{\Phi}(\mathbf{x}, \epsilon)\|^2$ entonces es posible hallar una solución del PCNL(F) (1.1) resolviendo el problema de minimización

$$\begin{aligned} &\textbf{Minimizar} \quad \Psi_\epsilon(\mathbf{x}) \\ &\mathbf{x} \in \mathbb{R}^n \end{aligned} \quad (2.18)$$

siempre que $\epsilon \rightarrow 0$.

En general, los algoritmos tipo Newton o cuasi Newton globales y suavizados propuestos para resolver el PCNL(F) tienen la siguiente estructura

Algoritmo 2.2. *Algoritmos tipo Newton y cuasi Newton globalizados y suavizados*

P.0 *Inicialización.*

P.1 *Criterio de parada.*

P.2 *Búsqueda direccional: Calcule $\mathbf{d}_k$ tal que*

$$\tilde{\Phi}(\mathbf{x}, \epsilon) + M(\mathbf{x}, \epsilon)\mathbf{d}_k = 0, \quad (2.19)$$

donde $M(\mathbf{x}, \epsilon)$ es igual a $\tilde{\Phi}'_x(\mathbf{x}, \epsilon)$ o una aproximación a ésta.

P.3 *Búsqueda lineal.*

P.4 *Actualización de parámetros.*

Nuevamente, los métodos tipo Newton inexactos basados en las aproximaciones suaves de (2.10) difieren del método exacto en la forma como es calculada la dirección de descenso $\mathbf{d}_k$, en **P.2**.

Cabe mencionar que se han propuesto diversas maneras de seleccionar el parámetro ϵ en la reformulación suave del problema (1.1), como por ejemplo, incluirlo como una variable más del problema original y hallar su valor resolviendo el sistema de ecuaciones en **P.2** o forzar un decrecimiento monótono del mismo de la siguiente manera:

$$\epsilon_{k+1} = \min \left\{ \tilde{\Phi}(\mathbf{x}_{k+1}, 0), \frac{1}{2}\epsilon_k \right\}.$$

Como podemos ver, en ambos enfoques, sistemas de ecuaciones no diferenciables y métodos suavizados, se han propuesto métodos tipo Newton y cuasi Newton exactos, los cuales se comportan bien experimentalmente para problemas relativamente pequeños. Entre tanto, cuando se piensa en resolver problemas de gran tamaño, las propuestas existentes (tipo Newton inexactas) necesitan el uso de las derivadas de la función que define el problema, lo cual los convierte en métodos costosos computacionalmente.

Por otro lado, en [2], los autores proponen un método de punto interior para resolver el problema de Complementariedad Horizontal y por ende, el PCNL(F). Este método usa direcciones tipo Newton inexactas y salvaguardan factibilidad haciendo uso de gradientes proyectados. En este artículo, los autores muestran que el método es global en el sentido que todo punto de acumulación de la sucesión generada por el algoritmo es un punto estacionario del problema de minimización emergente. De igual manera, en [1], los autores prueban que si se hace uso de direcciones de Newton exactas en el método propuesto por *Andreani et al* en [2] entonces el método goza de buenas propiedades de convergencia locales. Entre tanto, en [7], los autores prueban que el método tipo Newton inexacto propuesto en [1] goza de buenas propiedades de convergencia (convergencia lineal, superlineal y cuadrática).

Nuevamente, podemos ver que los avances realizados en esta dirección requieren del uso exacto de las derivadas de las funciones involucradas en los problemas.

En el siguiente capítulo, proponemos un algoritmo cuasi Newton inexacto para resolver Problemas de Complementariedad Horizontal y por ende, dada la relación explicada en la **Sección 2.1.3**, para resolver Problemas de Complementariedad no Lineal. Cabe mencionar que el método que proponemos desemboca en la solución de un problema de minimización diferenciable y que por tanto no será necesario recurrir a la diferenciación generalizada ni a aproximaciones suavizadas del mismo. De la misma manera, es im-

portante decir que el método se basa en la aproximación a las derivadas de la función resultante al reescribir el problema original, lo cual lo hará menos costoso computacionalmente. Además, resolverá de forma inexacta los sistemas de ecuaciones lineales resultantes en cada iteración, haciéndolo propicio para resolver problemas de gran tamaño.

CAPÍTULO 3

ALGORITMO CUASI NEWTON INEXACTO Y TEORÍA DE CONVERGENCIA

En este capítulo, abordaremos el Problema de Complementariedad Horizontal PCH(H) expuesto en la sección 2.1.3 y propondremos un algoritmo cuasi Newton inexacto para resolverlo. Cabe mencionar que la propuesta que presentaremos a continuación se realizó con el ánimo de resolver Problemas de Complementariedad no Lineal y que por ende, todo el desarrollo teórico se enriqueció de la relación existente entre los problemas, PCNL(F) y PCH(H). Sin embargo, debemos resaltar que el método propuesto es capaz de resolver, en teoría, “cualquier tipo” de Problema de Complementariedad Horizontal siempre y cuando las matrices que aproximen la matriz jacobiana en cuestión satisfagan un par de condiciones clásicas en este tipo de estudios.

Teniendo en cuenta que a partir de este momento el eje central de la investigación será el PCH(H) y la respectiva reformulación del PCNL(F) (1.1) como un PCH, nos permitimos definir nuevamente ambas situaciones.

Sea $H : \mathbb{R}^{n+m+n} \longrightarrow \mathbb{R}^{n+m}$ continuamente diferenciable; el Problema de Complementa-

riedad Horizontal (PCH(H)), consiste en hallar una terna de vectores $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$ y $\mathbf{w} \in \mathbb{R}^n$ tal que

$$\begin{aligned} H(\mathbf{x}, \mathbf{y}, \mathbf{w}) &= 0 \\ x_i w_i &= 0, \quad i = 1, 2, \dots, n \\ \mathbf{x}, \mathbf{w} &\geq 0, \end{aligned} \tag{3.1}$$

donde $\mathbf{y}$ es una variable libre del problema la cual podría, incluso, no existir. En particular, si tenemos en mente el PCNL(F) (1.1) y definimos $H(\mathbf{x}, \mathbf{w}) = F(\mathbf{x}) - \mathbf{w}$ entonces el problema (1.1) es equivalente al Problema de Complementariedad Horizontal

$$\begin{aligned} H(\mathbf{x}, \mathbf{w}) &= 0, \\ \mathbf{x}^T \mathbf{w} &= 0, \\ \mathbf{x}, \mathbf{w} &\geq 0. \end{aligned} \tag{3.2}$$

En lo que sigue nos ocuparemos del problema (3.1) y haremos algunas acotaciones, cuando sea necesario, para referirnos al problema (3.2).

Observe que si escribimos $\mathbf{z} = (\mathbf{x}, \mathbf{y}, \mathbf{w})$ para denotar el vector $\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{w} \end{pmatrix}$ y

$$\Omega = \{(\mathbf{x}, \mathbf{y}, \mathbf{w}) \in \mathbb{R}^{n+m+n} : \mathbf{x}, \mathbf{w} \geq 0\},$$

entonces podemos reescribir el PCH(H) como el sistema de ecuaciones

$$G(\mathbf{z}) = 0, \quad \mathbf{z} \in \Omega, \tag{3.3}$$

donde $G : \mathbb{R}^{n+m+n} \longrightarrow \mathbb{R}^{n+m+n}$ está definida por

$$G(\mathbf{z}) = \begin{pmatrix} H(\mathbf{z}) \\ x_1 w_1 \\ \vdots \\ x_n w_n \end{pmatrix}.$$

Con el ánimo de proponer un método global que resuelva el sistema de ecuaciones (3.3), consideramos el problema de minimización con restricción

$$\begin{array}{ll} \textbf{Minimizar} & f(\mathbf{z}) \\ \text{s.a} & \mathbf{z} \in \Omega, \end{array} \quad (3.4)$$

donde $f(\mathbf{z}) = \frac{1}{2}\|G(\mathbf{z})\|^2$ y $\|\cdot\|$ denota la norma euclidiana. En adelante, $P_\Omega(\mathbf{v})$ indicará la proyección ortogonal del vector $\mathbf{v}$ sobre Ω .

En [1, 2, 3, 7], los autores presentan un método tipo Newton inexacto de punto interior para resolver *Problemas de Complementariedad Horizontal* y desarrollan la teoría de convergencia en la que, entre otras cosas, se destaca su convergencia global en el sentido en que el algoritmo genera sucesiones que, bajo hipótesis razonables, converge a un punto estacionario del problema (3.4) con una tasa de convergencia acorde a lo esperado para métodos tipo Newton inexactos.

Ahora, teniendo en cuenta que, aunque los métodos tipo Newton gozan de buenas propiedades en cuanto a su tasa de convergencia, es bien sabido que una de sus principales debilidades es el uso de la matriz jacobiana de G , lo cual, en problemas densos o de gran tamaño, resulta ser costoso computacionalmente. Por esta razón, la comunidad matemática se ha preocupado por desarrollar métodos que gocen de buenas propiedades de convergencia pero que resulten ser más económicos computacionalmente. Así pues, se da cabida a una nutrida lista de métodos cuasi Newton para resolver una gran variedad de problemas y en los que la principal diferencia con respecto a los métodos tipo Newton es el uso de aproximaciones a la matriz jacobiana de G en lugar de usar la matriz exacta.

Teniendo en cuenta lo expuesto hasta el momento, a continuación presentamos un método cuasi Newton inexacto de punto interior para resolver el problema de minimización (3.4) en el que usamos, siempre que podamos, una dirección cuasi Newton inexacta y en el que, para garantizar factibilidad, tan solo en algunas iteraciones, se deba proyectar dicha dirección. Esto con el objetivo de retener la buenas propiedades de convergencia de los métodos cuasi Newton.

La estructura general del algoritmo cuasi Newton inexacto, con direcciones Proyectadas (CNIDP) es la siguiente:

Algoritmo 3.1. Sean $\varepsilon > 0$, $\mathbf{z}^0 \in \Omega$, $B_0 \in \mathbb{R}^{n \times n}$ no singular, $\theta \in (0, 1)$, $\beta \in (0, 1)$, $\lambda \in (0, \frac{1}{3})$, $\tau \in (0, 1)$, $c_{big} > c_{small} > 0$, $c_{small} < 1$, $\{\sigma_k\}$ y $\{\gamma_k\}$ sucesiones decrecientes tales que $k\sigma_k \rightarrow 0$ y $\gamma_k \rightarrow 0$. Para $k = 0, 1, \dots$ hacer

P.1 Criterio de parada: Si $\|G(\mathbf{z}_k)\| \leq \varepsilon$ pare. Sino, siga los pasos 2 a 6.

P.2 Dirección cuasi Newton Inexacta: Calcule

$$\mathbf{d}_k^{CNI} \in \mathbb{R}^n$$

tal que

$$\|B_k \mathbf{d}_k^{CNI} + G(\mathbf{z}_k)\| \leq \theta_k \|G(\mathbf{z}_k)\|, \quad \theta_k \in (0, \theta). \quad (3.5)$$

Si $\|\mathbf{d}_k^{CNI}\| \leq c_{big}$ y

$$0 \leq x_i^k w_i + x_i (d_w^k)_i + w_i (d_x^k)_i \leq \gamma_k \frac{(\mathbf{x}^k)^T \mathbf{w}^k}{n}, \quad i = 1, 2, \dots, n \quad (3.6)$$

entonces define $\mathbf{p}_k = \mathbf{d}_k^{CNI}$ sino, pase a **P.4**.

P.3 Tamaño de paso máximo: Dado $\tau_k \in [\tau, 1)$, calcule

$$\alpha_k^{break} = \max\{\alpha \in [0, 1] \mid \mathbf{z}_k + \alpha \mathbf{p}_k \in \Omega\} \quad (3.7)$$

y haga $\alpha_k^{max} = \tau_k \alpha_k^{break}$. Si $\alpha_k^{max} \leq c_{small}$, pase a **P.4**, de lo contrario, pase a **P.5**.

P.4 Direcciones proyectadas: Escoja $\tau_k \in [\tau, 1)$, haga $\alpha_k = \tau_k$ y define

$$\mathbf{p}_+ = P_\Omega(\mathbf{z}_k + \mathbf{d}_k^{CNI}) - \mathbf{z}_k \quad \text{y} \quad \mathbf{p}_- = -\mathbf{p}_+$$

P.4.1 Si $\|G(\mathbf{z}_k + \alpha_k \mathbf{p}_+)\| \leq [1 - \lambda(1 + \alpha_k)] \|G(\mathbf{z}_k)\| + \sigma_k$ define $\mathbf{p}_k = \mathbf{p}_+$ y pase a **P.6**

P.4.2 Si $\mathbf{z}_k + \alpha_k \mathbf{p}_- \in \Omega$ y $\|G(\mathbf{z}_k + \alpha_k \mathbf{p}_-)\| \leq [1 - \lambda(1 + \alpha_k)] \|G(\mathbf{z}_k)\| + \sigma_k$ define $\mathbf{p}_k = \mathbf{p}_-$ y pase a **P.6**

P.4.3 haga $\alpha_k = \beta \alpha_k$ y vuelva a **P.4.1**

P.5 Búsqueda lineal: calcule $\alpha_k = \alpha_k^{max} \max\{\beta^l : l = 0, 1, 2, \dots\}$ tal que

$$\|G(\mathbf{z}_k + \alpha_k \mathbf{p}_k)\| \leq [1 - \lambda(1 + \alpha_k)] \|G(\mathbf{z}_k)\| + \sigma_k \quad (3.8)$$

P.6 Actualización: Haga $\mathbf{z}_{k+1} = \mathbf{z}_k + \alpha_k \mathbf{p}_k$ y actualice B_{k+1}

Fin.

Sobre este algoritmo, es importante tener en cuenta algunas observaciones,

1. Si $\nabla f^T \mathbf{p}_+ \neq 0$ entonces $\mathbf{p}_+$ es una dirección de descenso o $\mathbf{p}_-$ lo es, por lo tanto, el ciclo en **P.4** terminará en un número finito de repeticiones.
2. Por construcción, podemos garantizar que si $\alpha_k \in (0, 1)$ entonces $\mathbf{z}_k + \alpha_k \mathbf{p}_+ \in \Omega$. Sin embargo, no es posible garantizar que $\mathbf{z}_k + \alpha_k \mathbf{p}_- \in \Omega$, por esta razón, es necesario verificar en **P.4.2** la permanencia de $\mathbf{z}_k + \alpha_k \mathbf{p}_-$ en Ω .
3. Una condición necesaria para garantizar que el ciclo en **P.5** termine en un número finito de repeticiones es que la dirección $\mathbf{d}_k^{CNI}$ sea una dirección de descenso. En [60], los autores demuestran que usando el método de gradientes conjugados para resolver el sistema (3.5) podemos construir direcciones de descenso para f .
4. Dada la escogencia inicial de α_k en cada iteración y la condición requerida en **P.4.2**, es trivial ver que $\mathbf{z}_k \in \text{int}(\Omega)$ para todo $k \geq 0$, razón por la cual podemos garantizar que $\mathbf{p}_+ \neq 0$ para todo k .

Las hipótesis bajo las cuales desarrollamos la teoría de convergencia del algoritmo propuesto son:

- H0.** Existe $\mathbf{z}^* \in \Omega$ tal que $G(\mathbf{z}^*) = 0$.
- H1.** Si $\{\mathbf{z}_k\}$ es una sucesión generada por el algoritmo CNIDP y $\{B_k\}$ es una sucesión de matrices que aproxima a $G'(\mathbf{z}_k)$ entonces existe $\tilde{k}$ tal que, para todo $k > \tilde{k}$, B_k es no singular y además existe $T > 0$ tal que $\|B_k^{-1}\| \leq T$.
- H2.** $G'(\mathbf{z})$ es una función lipschitz continua en Ω , es decir, existe una constante $L > 0$ tal que, para todo $\mathbf{z}, \mathbf{z}' \in \Omega$,

$$\|G'(\mathbf{z}) - G'(\mathbf{z}')\| \leq L \|\mathbf{z} - \mathbf{z}'\|.$$

Con respecto a las hipótesis ya establecidas, debemos observar que una consecuencia inmediata de la hipótesis **H2** es que, para todo $\mathbf{z}, \mathbf{z}' \in \Omega$,

$$\|G(\mathbf{z}') - G(\mathbf{z}) - G'(\mathbf{z})(\mathbf{z}' - \mathbf{z})\| \leq \frac{L}{2} \|\mathbf{z}' - \mathbf{z}\|^2. \quad (3.9)$$

El anterior resultado se puede deducir partiendo del hecho que

$$G(\mathbf{z}') - G(\mathbf{z}) - G'(\mathbf{z})(\mathbf{z}' - \mathbf{z}) = \int_0^1 [G'((\mathbf{z}' - \mathbf{z})t + \mathbf{z}) - G'(\mathbf{z})](\mathbf{z}' - \mathbf{z}) dt,$$

luego, si $\mathbf{z}, \mathbf{z}' \in \Omega$ entonces, tomando normas en ambos lados de la anterior igualdad y, teniendo en cuenta la hipótesis **H2**, es posible probar (3.9).

En el primer teorema que presentamos, demostramos que el algoritmo CNIDP genera una sucesión tal que $G(\mathbf{z}_k)$ converge a cero.

Teorema 3.1. *Si $\{\mathbf{z}_k\}$ es una sucesión generada por el algoritmo CNIDP entonces*

$$\lim_{k \rightarrow \infty} G(\mathbf{z}_k) = 0 \quad (3.10)$$

Demostración: Sean $\mathbf{p}_k$ una dirección calculada en **P.2** o **P.4** y α_k el tamaño del paso calculado en las búsquedas lineales llevadas a cabo en **P.4.1**, **P.4.2** o **P.5**, entonces,

$$\begin{aligned} \|G(\mathbf{z}_{k+1})\| &\leq (1 - \lambda(1 + \alpha_k))\|G(\mathbf{z}_k)\| + \sigma_k \\ &\leq (1 - \lambda)\|G(\mathbf{z}_k)\| + \sigma_k \\ &\leq (1 - \lambda)^{(k+1)}\|G(\mathbf{z}_0)\| + (k + 1)\sigma_{k+1}, \end{aligned}$$

pero como $\lambda \in (0, \frac{1}{3})$ y $k\sigma_k \rightarrow 0$ entonces podemos concluir que $\|G(\mathbf{z}_k)\| \rightarrow 0$ siempre que $k \rightarrow \infty$ de lo cual inferimos el resultado deseado. ■

En el siguiente teorema, demostramos que bajo una condición de regularidad, el algoritmo CNIDP genera sucesiones que convergen a la solución del problema (3.3).

Teorema 3.2. *Asuma la hipótesis **H1**. Si $\{\mathbf{z}_k\}$ es una sucesión generada por el algoritmo CNIDP y $\mathbf{z}^*$ es un punto de acumulación de la sucesión tal que $G'(\mathbf{z}^*)$ es no singular entonces $\|G(\mathbf{z}^*)\| = 0$ y además,*

$$\lim_{k \rightarrow \infty} \mathbf{z}_k = \mathbf{z}^* \quad (3.11)$$

Demostración: En primer lugar, como $\mathbf{z}^*$ es un punto de acumulación de $\{\mathbf{z}_k\}$ entonces existe una subsucesión $\{\mathbf{z}_{k_j}\}$ tal que $\mathbf{z}_{k_j} \rightarrow \mathbf{z}^*$ cuando $j \rightarrow \infty$. Luego, como G es continua entonces

$$G(\mathbf{z}_{k_j}) \rightarrow G(\mathbf{z}^*).$$

Ahora, por el **Teorema 3.1**, sabemos que $G(\mathbf{z}_k) \rightarrow 0$, por lo tanto, podemos inferir que $G(\mathbf{z}^*) = 0$.

En segundo lugar, sea $K = \|G'(\mathbf{z}^*)^{-1}\|$ y sea δ lo suficientemente pequeño tal que, para todo $\mathbf{y} \in B(\mathbf{z}^*, \delta)$ se cumple que

- i. $G'(\mathbf{y})^{-1}$ existe.
- ii. $\|G'(\mathbf{y})^{-1}\| < 2K$.
- iii. $\|G(\mathbf{y}) - G(\mathbf{z}^*) - G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\| \leq \frac{1}{2K}\|\mathbf{y} - \mathbf{z}^*\|$.

Cabe mencionar que la existencia de tal δ la podemos garantizar gracias a los Lemas 1.1 y 1.2 en [26].

Observemos que si $\mathbf{y} \in B(\mathbf{z}^*, \delta)$ entonces

$$\begin{aligned} \|G(\mathbf{y})\| &= \|G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*) + G(\mathbf{y}) - G(\mathbf{z}^*) - G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\| \\ &\geq \|G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\| - \|G(\mathbf{y}) - G(\mathbf{z}^*) - G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\| \\ &\geq \|G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\| - \frac{1}{2K}\|\mathbf{y} - \mathbf{z}^*\|. \end{aligned}$$

Ahora,

$$\|\mathbf{y} - \mathbf{z}^*\| = \|G'(\mathbf{z}^*)^{-1}G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\| \leq \|G'(\mathbf{z}^*)^{-1}\| \cdot \|G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\| = K \|G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\|$$

así que,

$$\frac{1}{K}\|\mathbf{y} - \mathbf{z}^*\| \leq \|G'(\mathbf{z}^*)(\mathbf{y} - \mathbf{z}^*)\|,$$

por lo tanto,

$$\|G(\mathbf{y})\| \geq \frac{1}{2K}\|\mathbf{y} - \mathbf{z}^*\|;$$

es decir,

$$\|\mathbf{y} - \mathbf{z}^*\| \leq 2K\|G(\mathbf{y})\| \quad \text{para todo } \mathbf{y} \in B(\mathbf{z}^*, \delta). \quad (3.12)$$

Por otro lado, sea $\epsilon \in (0, \delta/8)$ dado. Como $\mathbf{z}^*$ es un punto de acumulación de $\{\mathbf{z}_k\}$ y $G(\mathbf{z}^*) = 0$ entonces existe un k suficientemente grande tal que

$$\mathbf{z}_k \in S_\epsilon = \left\{ \mathbf{y} : \mathbf{y} \in B(\mathbf{z}^*, \delta/2), \quad [M(1 + \theta)]\|G(\mathbf{y})\| < \epsilon \text{ y } \sigma_k < \frac{\epsilon}{K} \right\},$$

donde $M = \max\{K, T\}$ y T es la constante de la hipótesis **H1**, por lo tanto,

$$\begin{aligned}
 \|\mathbf{d}_k^{CNI}\| &= \|B_k^{-1}[-G(\mathbf{z}_k) + G(\mathbf{z}_k) + B_k \mathbf{d}_k^{CNI}]\| \\
 &\leq \|B_k^{-1}\| [\|G(\mathbf{z}_k)\| + \|G(\mathbf{z}_k) + B_k \mathbf{d}_k^{CNI}\|] \\
 &\leq M[\|G(\mathbf{z}_k)\| + \theta\|G(\mathbf{z}_k)\|] \\
 &= M(1 + \theta)\|G(\mathbf{z}_k)\| \\
 &< \epsilon.
 \end{aligned} \tag{3.13}$$

Ahora, como $\epsilon \in (0, \delta/8)$ entonces podemos inferir que $\|\mathbf{d}_k^{CNI}\| \leq \frac{\delta}{2}$. Por otro lado, por propiedades de la proyección,

$$\begin{aligned}
 \|P_\Omega(\mathbf{z}_k + \mathbf{d}_k^{CNI}) - \mathbf{z}_k\| &= \|P_\Omega(\mathbf{z}_k + \mathbf{d}_k^{CNI}) - P_\Omega(\mathbf{z}_k)\| \\
 &\leq \|\mathbf{d}_k^{CNI}\|
 \end{aligned} \tag{3.14}$$

En consecuencia, en cualquiera de los casos en que sea calculada la dirección $\mathbf{p}_k$, podemos garantizar que $\|\mathbf{p}_k\| \leq \frac{\delta}{2}$, de donde,

$$\begin{aligned}
 \|\mathbf{z}_{k+1} - \mathbf{z}^*\| &= \|\mathbf{z}_k + \alpha_k \mathbf{p}_k - \mathbf{z}^*\| \\
 &\leq \|\mathbf{z}_k - \mathbf{z}^*\| + |\alpha_k| \|\mathbf{p}_k\| \\
 &< \frac{\delta}{2} + \frac{\delta}{2} = \delta.
 \end{aligned}$$

Por lo tanto, $\mathbf{z}_{k+1} \in B(\mathbf{z}^*, \delta)$. Ahora, como

$$\|G(\mathbf{z}_{k+1})\| < \|G(\mathbf{z}_k)\| + \sigma_k < \frac{\epsilon}{M(1 + \theta)} + \sigma_k \leq \frac{\epsilon}{K(1 + \theta)} + \frac{\epsilon}{K}$$

y por (3.12)

$$\|\mathbf{z}_{k+1} - \mathbf{z}^*\| < 2K\|G(\mathbf{z}_{k+1})\|,$$

concluimos que

$$\|z_{k+1} - z^*\| < \frac{2\epsilon}{1+\theta} + 2\epsilon < \frac{2}{1+\theta} \cdot \frac{\delta}{8} + \frac{\delta}{4} < \frac{\delta}{2},$$

por lo tanto, $z_{k+1} \in S_\epsilon$.

En resumen, para todo k mayor que un $\bar{k}$ suficientemente grande, se cumple que $z_k \in S_\epsilon$ y como $\|G(z_k)\| \rightarrow 0$, de (3.12), concluimos que $z_k \rightarrow z^*$ cuando $k \rightarrow \infty$. ■

En el siguiente lema, garantizamos que las direcciones p_k generadas por el algoritmo CNIPD se encuentran acotadas superiormente.

Lema 3.1. *Bajo la hipótesis **H1**. Si p_k es la dirección calculada en **P.2** o **P.4** entonces existe una constante κ tal que*

$$\|p_k\| \leq \kappa \|G(z_k)\|.$$

Demostración: De (3.13) y (3.14), inferimos que

$$\|p_-\| = \|p_+\| \leq \|d_k^{CNI}\| \leq T(1+\theta)\|G(z_k)\| \leq 2T\|G(z_k)\|,$$

luego, para todo $k \geq 0$, existe $\kappa = 2T$ tal que

$$\|p_k\| \leq \kappa \|G(z_k)\|.$$

■

A continuación, establecemos una condición necesaria para garantizar la existencia de una dirección cuasi Newton inexacta que satisfaga (3.5) y (3.6).

Teorema 3.3. *Sea $z \in \Omega$ tal que $G(z) \neq 0$. Si existe $\bar{d} \neq 0$ tal que*

$$\|G(z) + B\bar{d}\| < \|G(z)\|$$

y

$$0 \leq x_i w_i + x_i \overline{(d_w)_i} + w_i \overline{(d_x)_i} = \mu_i, \quad i = 1, 2, \dots, n \quad (3.15)$$

entonces existe $\theta_{\min} \in [0, 1)$ tal que, para cualquier $\theta \in [\theta_{\min}, 1)$, existen $d \in \mathbb{R}^{n+m+n}$ y $\lambda \in [0, 1)$ tal que

$$\|G(z) + Bd\| \leq \theta \|G(z)\| \quad (3.16)$$

y

$$x_i w_i + x_i (d_w)_i + w_i (d_x)_i = \mu_i + \lambda x_i w_i.$$

En particular, si $\|G(\mathbf{z}) + B\bar{\mathbf{d}}\| = 0$ y $x_i w_i + x_i \overline{(d_w)_i} + w_i \overline{(d_x)_i} = \mu_i = 0$ entonces, para cualquier $\gamma \in (0, 1)$ y $\theta \in [0, \frac{\gamma}{n})$, existe $\mathbf{d} \in \mathbb{R}^{n+m+n}$ tal que (3.16) se cumple y además,

$$x_i w_i + x_i (d_w)_i + w_i (d_x)_i \leq \gamma \frac{x^T w}{n}.$$

Demostracion. Claramente, $\bar{\mathbf{d}} \neq 0$. Sean

$$\bar{\theta} = \frac{\|G(\mathbf{z}) + B\bar{\mathbf{d}}\|}{\|G(\mathbf{z})\|} \quad (3.17)$$

y

$$\mathbf{d} = \frac{1 - \theta}{1 - \bar{\theta}} \bar{\mathbf{d}}, \quad (3.18)$$

con $\theta \in [\bar{\theta}, 1)$. Observemos que

$$\begin{aligned} \|G(\mathbf{z}) + B\mathbf{d}\| &= \left\| G(\mathbf{z}) + B \cdot \frac{1 - \theta}{1 - \bar{\theta}} \bar{\mathbf{d}} \right\| \\ &= \left\| \frac{(1 - \bar{\theta})G(\mathbf{z}) + (1 - \theta)B\bar{\mathbf{d}}}{1 - \bar{\theta}} \right\| \\ &= \left\| \frac{G(\mathbf{z}) - \bar{\theta}G(\mathbf{z}) + B\bar{\mathbf{d}} - \theta B\bar{\mathbf{d}} + \theta G(\mathbf{z}) - \theta G(\mathbf{z})}{1 - \bar{\theta}} \right\| \\ &= \left\| \frac{(\theta - \bar{\theta})G(\mathbf{z}) + (1 - \theta)(G(\mathbf{z}) + B\bar{\mathbf{d}})}{1 - \frac{\|G(\mathbf{z}) + B\bar{\mathbf{d}}\|}{\|G(\mathbf{z})\|}} \right\| \\ &\leq \frac{(\theta - \bar{\theta})\|G(\mathbf{z})\| + (1 - \theta)\|G(\mathbf{z}) + B\bar{\mathbf{d}}\|}{\frac{\|G(\mathbf{z})\| - \|G(\mathbf{z}) + B\bar{\mathbf{d}}\|}{\|G(\mathbf{z})\|}} \\ &= \frac{[\theta\|G(\mathbf{z})\| - \bar{\theta}\|G(\mathbf{z})\| + \|G(\mathbf{z}) + B\bar{\mathbf{d}}\| - \theta\|G(\mathbf{z}) + B\bar{\mathbf{d}}\|] \|G(\mathbf{z})\|}{\|G(\mathbf{z})\| - \|G(\mathbf{z}) + B\bar{\mathbf{d}}\|}. \end{aligned}$$

Por (3.17), sabemos que

$$\bar{\theta} \|G(\mathbf{z})\| = \|G(\mathbf{z}) + B\bar{\mathbf{d}}\|,$$

por lo tanto,

$$\|G(\mathbf{z}) + B\mathbf{d}\| \leq \frac{\theta (\|G(\mathbf{z})\| - \|G(\mathbf{z}) + B\bar{\mathbf{d}}\|) \|G(\mathbf{z})\|}{\|G(\mathbf{z})\| - \|G(\mathbf{z}) + B\bar{\mathbf{d}}\|} = \theta \|G(\mathbf{z})\|.$$

Así, si definimos $\theta_{min} = \bar{\theta}$, tenemos (3.16).

Ahora, de (3.15) y (3.18),

$$\begin{aligned} x_i w_i + x_i (d_w)_i + w_i (d_x)_i &= x_i w_i + \left(\frac{1 - \theta}{1 - \bar{\theta}} \right) (x_i \overline{(d_w)_i} + w_i \overline{(d_x)_i}) \\ &= x_i w_i + \left(\frac{1 - \theta}{1 - \bar{\theta}} \right) (\mu_i - x_i w_i) \\ &\leq \mu_i + \left(1 - \frac{1 - \theta}{1 - \bar{\theta}} \right) x_i w_i \end{aligned}$$

por tanto, si $\lambda = 1 - \left(\frac{1 - \theta}{1 - \bar{\theta}} \right)$ entonces existe $\mathbf{d} \in \mathbb{R}^{n+m+n}$ tal que (3.16) se cumple y

$$x_i w_i + x_i (d_w)_i + w_i (d_x)_i \leq \mu_i + \lambda x_i w_i.$$

Finalmente, supongamos que $\|G(\mathbf{z}) + B\bar{\mathbf{d}}\| = 0$ y que $x_i w_i + x_i \overline{(d_w)_i} + w_i \overline{(d_x)_i} = \mu_i = 0$ entonces $\bar{\theta} = 0$ y $\lambda = \theta$. Así, para cualquier $\theta \in [0, 1)$, existe $\mathbf{d} \in \mathbb{R}^{n+m+n}$ tal que (3.16) se satisface y, además,

$$x_i w_i + x_i (d_w)_i + w_i (d_x)_i \leq \theta x_i w_i;$$

en consecuencia, si $\theta \in [0, \frac{\gamma}{n})$ con $\gamma \in (0, 1)$ entonces

$$x_i w_i + x_i (d_w)_i + w_i (d_x)_i \leq \gamma \frac{x^T w}{n}.$$

■

Observemos que

$$G'(\mathbf{z}) = \begin{pmatrix} H'_x(\mathbf{z}) & H'_y(\mathbf{z}) & H'_w(\mathbf{z}) \\ W & 0 & X \end{pmatrix},$$

donde $W = \text{diag}(w_1, w_2, \dots, w_n)$ y $X = \text{diag}(x_1, x_2, \dots, x_n)$. En particular, para la función G obtenida a partir del Problema de Complementariedad Horizontal (3.2), tenemos que

$$G'(z) = \begin{pmatrix} H'_x(z) & -I \\ W & X \end{pmatrix} = \begin{pmatrix} F'(x) & -I \\ W & X \end{pmatrix}.$$

Luego, si A_k es una aproximación de $F'(x_k)$ y

$$B_k = \begin{pmatrix} A_k & -I \\ W & X \end{pmatrix} \quad (3.19)$$

es la respectiva aproximación a $G'(z_k)$ tal que satisface la hipótesis **H1** entonces podemos garantizar que, para cada k , existe $\bar{d}_k = -B_k^{-1}G(z_k)$ tal que $\|G(z_k) + B_k\bar{d}_k\| < \|G(z_k)\|$ y $x_i w_i + x_i \overline{(d_w)_i} + w_i \overline{(d_x)_i} = 0$. Así, por el teorema que hemos acabado de probar, concluimos que en cada iteración es posible encontrar una dirección $\mathbf{d}_k^{CNI}$ que satisfaga (3.5) y (3.6) para cualquier $\theta \in (0, \frac{\gamma_k}{n})$.

La principal conclusión del teorema que presentamos a continuación es que, eventualmente, la dirección usada en la rutina del algoritmo CNIDP será la dirección cuasi Newton inexacta calculada en **P.2**. Es decir, solo será necesario proyectar la dirección $\mathbf{d}^{CNI}$ en algunas primeras iteraciones.

Teorema 3.4. *Asuma la hipótesis **H1**. Sea $\{z^k\}$ una sucesión generada por el algoritmo CNIDP. Si z^* un punto de acumulación de la sucesión tal que $G'(z^*)$ es no singular y $c_{big} \geq 2T$ entonces, para $\bar{k}$ suficientemente grande, la dirección $\mathbf{p}^k$, para $k \geq \bar{k}$, es definida por el paso **P.2** del algoritmo.*

Demostración: Por el **Teorema 3.2**, sabemos que $z_k \rightarrow z^*$ y que $G(z_k) \rightarrow 0$, luego, existe $\bar{k} > 0$ tal que **H1** se satisface y además, $\|G(z_k)\| < 1$ para todo $k \geq \bar{k}$, en consecuencia, si $k \geq \bar{k}$ entonces

$$\begin{aligned} \|\mathbf{d}_k^{CNI}\| &= \|B_k^{-1} [-G(z_k) + G(z_k) + B_k \mathbf{d}_k^{CNI}]\| \\ &\leq \|B_k^{-1}\| [\|G(z_k)\| + \|G(z_k) + B_k \mathbf{d}_k^{CNI}\|] \\ &\leq T(1 + \theta_k) \|G(z_k)\| \\ &\leq 2T \\ &\leq c_{big}; \end{aligned}$$

por lo tanto, la dirección calculada en **P.2** será aceptada. ■

El siguiente teorema es fundamental para demostrar las buenas propiedades de convergencia (en cuanto a velocidad) del algoritmo CNIDP pues en él, básicamente, demostramos que el tamaño de paso “completo”, eventualmente, será aceptado.

Teorema 3.5. *Asuma la hipótesis **H1** y sea $\{z_k\}$ una sucesión generada por el algoritmo CNIDP. Si $z^* \in \Omega$ es un punto de acumulación de dicha sucesión, $G'(z^*)$ es no singular, $c_{big} \geq 2T$ y γ_k es tal que $\gamma_k \frac{(x^k)^T w^k}{n} \leq \gamma_k x_i^k w_i^k \leq x_i^k w_i^k$ entonces,*

$$\lim_{k \rightarrow \infty} \alpha_k^{break} = 1$$

Demostración: Por el **Teorema 3.2**, sabemos que $z_k \rightarrow z^*$, $k \rightarrow \infty$ y por el **Teorema 3.4**, inferimos que la dirección d_k^{CNI} calculada en (3.5) será aceptada en el paso **P.2** del algoritmo. Ahora, como $G'(z^*)$ es no singular entonces existe $\bar{k}$ tal que $G'(z_k)$ es no singular para todo $k \geq \bar{k}$, pero como

$$G'(z_k) = \begin{pmatrix} H'_x(z_k) & H'_y(z_k) & H'_w(z_k) \\ W_k & 0 & X_k \end{pmatrix}$$

entonces podemos inferir que x_i^k y w_i^k no se anulan simultáneamente. Sin pérdida de generalidad, supongamos $x_i^k > 0$ para $i = 1, 2, \dots, n$.

De (3.6), sabemos que

$$0 \leq x_i^k w_i^k + x_i^k (d_w^k)_i + w_i^k (d_x^k)_i \leq \gamma_k \frac{(x^k)^T w^k}{n} \quad (3.20)$$

y por hipótesis, tenemos que

$$\gamma_k \frac{(x^k)^T w^k}{n} \leq x_i^k w_i^k, \quad i = 1, 2, \dots, n \quad (3.21)$$

en consecuencia,

$$x_i^k (d_w^k)_i + w_i^k (d_x^k)_i \leq 0, \quad (3.22)$$

pero como $x_i^k > 0$ y $w_i^k \geq 0$ entonces $(d_x^k)_i \leq 0$ o $(d_w^k)_i \leq 0$ pero no puede suceder que ambos sean no negativos simultáneamente, es decir, no es posible que $(d_x^k)_i = 0$ y $(d_w^k)_i > 0$ y viceversa. Por tal razón, hemos considerado los siguientes casos posibles:

Caso 1: $(d_x^k)_i = (d_w^k)_i = 0$.

Caso 2: $(d_x^k)_i < 0$ y $(d_w^k)_i \geq 0$.

Caso 3: $(d_x^k)_i < 0$ y $(d_w^k)_i < 0$.

Caso 4: $(d_x^k)_i \geq 0$ y $(d_w^k)_i < 0$.

Si el Caso 1 ocurre para todos los índices i entonces la iteración actual es solución del problema, por lo tanto, supondremos que los casos 2, 3 o 4 ocurren por lo menos para algún índice i . Ahora bien, de (3.7), inferimos que α_k^{break} debe ser tal que

$$x_i^k + \alpha_k^{break}(d_x^k)_i \geq 0 \quad \text{y} \quad w_i^k + \alpha_k^{break}(d_w^k)_i \geq 0. \quad (3.23)$$

Luego,

- Si i es un índice para el cual el Caso 1 ocurre entonces (3.23) será satisfecho por cualquier $\alpha_k^{break} \geq 0$.
- Si i es un índice para el cual el Caso 2 ocurre entonces (3.23) será satisfecho siempre que

$$\alpha_k^{break} \leq -\frac{x_i^k}{(d_x^k)_i} \quad (3.24)$$

- Si i es un índice para el cual el Caso 3 ocurre entonces (3.23) será satisfecho siempre que

$$\alpha_k^{break} \leq \min \left\{ -\frac{x_i^k}{(d_x^k)_i}, -\frac{w_i^k}{(d_w^k)_i} \right\}. \quad (3.25)$$

- Finalmente, si i es un índice para el cual el Caso 4 ocurre entonces (3.23) será satisfecho siempre que

$$\alpha_k^{break} \leq -\frac{w_i^k}{(d_w^k)_i}. \quad (3.26)$$

Como α_k^{break} debe satisfacer (3.24), (3.25), y (3.26) simultáneamente entonces

$$\alpha_k^{break} = \min_{i \in I, j \in J, l \in L} \left\{ -\frac{x_i^k}{(d_x^k)_i}, \min \left\{ -\frac{x_j}{(d_x^k)_j}, -\frac{w_j}{(d_w^k)_j} \right\}, -\frac{w_l}{(d_w^k)_l} \right\},$$

donde I es el conjunto de índices para los cuales el Caso 2 ocurre, J es el conjunto de índices para los cuales el Caso 3 ocurre y L es el conjunto de índices para los cuales el Caso 4 ocurre.

Observe que en los Casos 3 y 4 $w_i^k \neq 0$, ya que de lo contrario, (3.20) implica que $(d_w^k)_i = 0$, lo cual no es posible. Por otra parte, como $G'(\mathbf{z}^*)$ es no singular entonces, por el **Teorema 3.2**, $G(\mathbf{z}^*) = 0$, luego, por las condiciones de complementariedad y no negatividad, $x_i^* = 0$ o $w_i^* = 0$. Ahora, como

$$G'(\mathbf{z}^*) = \begin{pmatrix} H'_x(\mathbf{z}^*) & H'_y(\mathbf{z}^*) & H'_w(\mathbf{z}^*) \\ W^* & 0 & X^* \end{pmatrix},$$

de la no singularidad de $G'(\mathbf{z}^*)$ deducimos que x_i^* y w_i^* no se pueden anular simultáneamente. Nuevamente, sin pérdida de generalidad, supongamos $x_i^* = 0$.

Como $w_i^k \rightarrow w_i^* \neq 0$ y por el **Lema 3.1** existe una constante $\kappa > 0$ tal que $\|\mathbf{d}_k\| \leq \kappa \|G(\mathbf{z}^k)\|$ entonces

$$\lim_{k \rightarrow \infty} -\frac{w_i^k}{(d_w^k)_i} = \infty. \quad (3.27)$$

Así, para k suficientemente grande, podemos asumir que

$$\alpha_k^{break} = \min_{i \in I \cup J} \left\{ -\frac{x_i^k}{(d_x^k)_i} \right\}.$$

Ahora, por (3.6) e hipótesis del teorema,

$$0 \leq x_i^k w_i^k + x_i^k (d_w^k)_i + w_i^k (d_x^k)_i \leq \gamma_k x_i^k w_i^k,$$

por lo tanto,

$$|x_i^k + (d_x^k)_i| w_i^k - |x_i^k (d_w^k)_i| \leq |(x_i^k + (d_x^k)_i) w_i^k + x_i^k (d_w^k)_i| \leq \gamma_k x_i^k w_i^k.$$

Así que,

$$|x_i^k + (d_x^k)_i| \leq x_i^k \left(\frac{\gamma_k w_i^k + |(d_w^k)_i|}{w_i^k} \right). \quad (3.28)$$

Nuevamente, por el **Lema 3.1**, tenemos que para k suficientemente grande existe una constante κ tal que $\|\mathbf{d}^k\| \leq \kappa \|G(\mathbf{z}^k)\|$ luego, $|(d_w^k)_i| \leq \kappa \|G(\mathbf{z}^k)\|$ y como $G(\mathbf{z}^k) \rightarrow 0$, podemos concluir que $|(d_w^k)_i| \rightarrow 0$. Por otro lado, para los Casos 3 y 4, tenemos que $w_i^k \rightarrow w^* \neq 0$ y como $\gamma_k \rightarrow 0$ entonces

$$\lim_{k \rightarrow \infty} \left(\frac{\gamma_k w_i^k + |(d_w^k)_i|}{w_i^k} \right) = 0.$$

Luego, de (3.28), deducimos que $x_i^k + (d_x^k)_i = o(x_i^k)$. En consecuencia,

$$\lim_{k \rightarrow \infty} -\frac{x_i^k}{(d_x^k)_i} = 1,$$

lo cual completa la prueba del teorema. ■

Corolario 3.1. *Sea $\{z_k\}$ una sucesión generada por el algoritmo CNIDP. Si $z^* \in \Omega$ es un punto de acumulación de dicha sucesión, $G'(z^*)$ es no singular, $c_{big} \geq 2T$ y $\tau_k \rightarrow 1$, $k \rightarrow \infty$ entonces*

$$\lim_{k \rightarrow \infty} \alpha_k^{max} = 1$$

Demostración: Como $\alpha_k^{max} = \tau_k \alpha_k^{break}$ entonces, del teorema anterior, podemos deducir que $\alpha_k^{max} \rightarrow 1$. ■

En el siguiente teorema, demostramos las buenas propiedades de convergencia del algoritmo CNIDP. En él, establecemos condiciones bajo las cuales podríamos garantizar una tasa de convergencia superlineal.

Teorema 3.6. *Asuma las hipótesis **H1** y **H2** y sea $\{z_k\}$ una sucesión generada por el algoritmo CNIDP. Si $z^* \in \Omega$ es un punto de acumulación de dicha sucesión, $G'(z^*)$ es no singular, $c_{big} \geq 2T$, $\tau_k \rightarrow 1$, $\theta_k \rightarrow 0$ y además,*

$$\lim_{k \rightarrow \infty} \frac{\|(G'(z_k) - B_k)p_k\|}{\|p_k\|} = 0 \quad (3.29)$$

entonces,

$$\alpha_k = \alpha_k^{max}$$

para k suficientemente grande y además, $z_k \rightarrow z^*$ superlinealmente.

Demostración: Por el **Teorema 3.4**, sabemos que $p_k = d_k^{CNI}$ para todo k suficientemente grande. Ahora,

$$\begin{aligned} G(z_k + \alpha_k p_k) &= G(z_k) + \int_0^1 G'(z_k + t\alpha_k p_k) \alpha_k p_k dt \\ &= (1 - \alpha_k)G(z_k) + \alpha_k(G'(z_k)p_k - B_k p_k + B_k p_k + G(z_k)) + \\ &\quad \int_0^1 [G'(z_k + t\alpha_k p_k) - G'(z_k)] \alpha_k p_k dt. \end{aligned}$$

Por tanto, y teniendo en cuenta que $\alpha_k \in (0, 1)$, **H2** y que $\|\mathbf{p}_k\| \leq \kappa \|G(\mathbf{z}_k)\|$,

$$\begin{aligned}
 \|G(\mathbf{z}_k + \alpha_k \mathbf{p}_k)\| &\leq (1 - \alpha_k) \|G(\mathbf{z}_k)\| + \alpha_k \frac{\|(G'(z_k) - B_k) \mathbf{p}_k\|}{\|\mathbf{p}_k\|} \cdot \|\mathbf{p}_k\| + \alpha_k \|B_k \mathbf{p}_k + G(\mathbf{z}_k)\| + \\
 &\quad \int_0^1 \|G'(\mathbf{z}_k + t\alpha_k \mathbf{p}_k) - G'(\mathbf{z}_k)\| \|\alpha_k \mathbf{p}_k\| dt \\
 &\leq (1 - \alpha_k) \|G(\mathbf{z}_k)\| + \alpha_k \frac{\|(G'(z_k) - B_k) \mathbf{p}_k\|}{\|\mathbf{p}_k\|} \cdot \kappa \|G(\mathbf{z}_k)\| + \alpha_k \theta_k \|G(\mathbf{z}_k)\| + \\
 &\quad \frac{L}{2} \alpha_k^2 \|\mathbf{p}_k\|^2 \\
 &\leq \left[(1 - \alpha_k) + \kappa \frac{\|(G'(z_k) - B_k) \mathbf{p}_k\|}{\|\mathbf{p}_k\|} + \theta_k + \frac{L}{2} \kappa^2 \|G(\mathbf{z}_k)\| \right] \|G(\mathbf{z}_k)\|,
 \end{aligned}$$

pero como $G(\mathbf{z}_k) \rightarrow 0$ entonces $\frac{L}{2} \kappa^2 \|G(\mathbf{z}_k)\| \leq \frac{1}{6}$ para k suficientemente grande. Ahora, por simplicidad, denotaremos

$$\mu_k = \frac{\|(G'(z_k) - B_k) \mathbf{p}_k\|}{\|\mathbf{p}_k\|}.$$

Observemos que

$$1 - \alpha_k + \mu_k \kappa + \theta_k + \frac{1}{6} \leq 1 - 2\lambda \quad (3.30)$$

siempre que

$$\alpha_k \geq \frac{1}{6} + 2\lambda + (\mu_k \kappa + \theta_k),$$

pero como $\lambda \in (0, 1/3)$ entonces será suficiente que

$$\alpha_k \geq \frac{5}{6} + (\mu_k \kappa + \theta_k) \quad (3.31)$$

para garantizar que (3.30) se cumpla.

Finalmente, por el **Corolario 3.1**, sabemos que $\alpha_k^{max} \rightarrow 1$ y como por hipótesis $\mu_k, \theta_k \rightarrow 0$ entonces, para k suficientemente grande, α_k^{max} satisface (3.31) y por tanto (3.30), de tal manera que

$$\begin{aligned}
 \|G(\mathbf{z}_k + \alpha_k^{max} \mathbf{p}_k)\| &\leq (1 - 2\lambda) \|G(\mathbf{z}_k)\| \\
 &\leq [1 - \lambda(1 + \alpha_k^{max})] \|G(\mathbf{z}_k)\|,
 \end{aligned}$$

es decir, α_k^{max} satisface la condición (3.8) en el paso **P.5** del algoritmo y por tanto, $\alpha_k = \alpha_k^{max}$, para k suficientemente grande.

Por otro lado, como $G'(\mathbf{z}^*)$ es no singular entonces, para k suficientemente grande, se tiene que $G'(\mathbf{z}_k)$ es no singular y además, $\|G'(\mathbf{z}_k)^{-1}\| \leq 2\|G'(\mathbf{z}^*)^{-1}\| = 2K$ de donde,

$$\begin{aligned}
 \|\mathbf{z}_{k+1} - \mathbf{z}^*\| &= \|\mathbf{z}_k + \alpha_k^{max} \mathbf{p}_k - \mathbf{z}^* + G'(\mathbf{z}_k)^{-1}G(\mathbf{z}_k) - G'(\mathbf{z}_k)^{-1}G(\mathbf{z}_k)\| \\
 &\leq \|\mathbf{z}_k - \mathbf{z}^* - G'(\mathbf{z}_k)^{-1}G(\mathbf{z}_k)\| + \|G'(\mathbf{z}_k)^{-1}[G(\mathbf{z}_k) + G'(\mathbf{z}_k)\mathbf{p}_k]\| + |\alpha_k^{max} - 1|\|\mathbf{p}_k\| \\
 &\leq \|G'(\mathbf{z}_k)^{-1}\| \|G'(\mathbf{z}_k)(\mathbf{z}_k - \mathbf{z}^*) + G(\mathbf{z}^*) - G(\mathbf{z}_k)\| + \\
 &\quad \|G'(\mathbf{z}_k)^{-1}\| \left[\|G(\mathbf{z}_k) + G'(\mathbf{z}_k)\mathbf{p}_k + B_k\mathbf{p}_k - B_k\mathbf{p}_k\| \right] + (1 - \alpha_k^{max})\kappa\|G(\mathbf{z}_k)\| \\
 &\leq LK\|\mathbf{z}_k - \mathbf{z}^*\|^2 + 2K \left[\|G(\mathbf{z}_k) + B_k\mathbf{p}_k\| + \|G'(\mathbf{z}_k)\mathbf{p}_k - B_k\mathbf{p}_k\| \right] + \\
 &\quad (1 - \alpha_k^{max})\kappa\|G(\mathbf{z}_k)\| \\
 &\leq LK\|\mathbf{z}_k - \mathbf{z}^*\|^2 + 2K \left[\theta_k\|G(\mathbf{z}_k)\| + \mu_k\|\mathbf{p}_k\| \right] + (1 - \alpha_k^{max})\kappa\|G(\mathbf{z}_k)\| \\
 &\leq LK\|\mathbf{z}_k - \mathbf{z}^*\|^2 + 2K \left[\theta_k + \mu_k\kappa \right] \|G(\mathbf{z}_k)\| + (1 - \alpha_k^{max})\kappa\|G(\mathbf{z}_k)\|.
 \end{aligned}$$

Ahora, por el teorema del valor medio, sabemos que existe ξ_k en el segmento que une a $\mathbf{z}_k$ con $\mathbf{z}^*$ tal que

$$\|G(\mathbf{z}_k)\| = \|G(\mathbf{z}_k) - G(\mathbf{z}^*)\| \leq \|G'(\xi_k)\|\|\mathbf{z}_k - \mathbf{z}^*\|,$$

pero como $\mathbf{z}_k \rightarrow \mathbf{z}^*$ y G' es continua, podemos asegurar que existe una constante $R > 0$ tal que, para k suficientemente grande, $\|G'(\xi_k)\| \leq R$; así pues,

$$\|G(\mathbf{z}_k)\| \leq R\|\mathbf{z}_k - \mathbf{z}^*\|$$

y por lo tanto,

$$\begin{aligned}
 \|\mathbf{z}_{k+1} - \mathbf{z}^*\| &\leq LK\|\mathbf{z}_k - \mathbf{z}^*\|^2 + 2K \left[\theta_k + \mu_k\kappa \right] R\|\mathbf{z}_k - \mathbf{z}^*\| + (1 - \alpha_k^{max})R\kappa\|\mathbf{z}_k - \mathbf{z}^*\| \\
 &\leq \left[LK\|\mathbf{z}_k - \mathbf{z}^*\| + 2KR(\theta_k + \kappa\mu_k) + (1 - \alpha_k^{max})R\kappa \right] \|\mathbf{z}_k - \mathbf{z}^*\|
 \end{aligned}$$

con lo cual podemos garantizar la convergencia superlineal de la sucesión pues $\mathbf{z}_k \rightarrow \mathbf{z}^*$, $\mu_k, \theta_k \rightarrow 0$ y $\alpha_k^{max} \rightarrow 1$. ■

Como se puede ver, hemos demostrado que el algoritmo CNIDP genera sucesiones que mantienen factibilidad y que, bajo hipótesis razonables, convergen, con una tasa de convergencia superlineal, a la solución del problema. Vale la pena destacar que el hecho de resolver de forma inexacta el sistema de ecuaciones $B_k \mathbf{d}_k^{CNI} = G(\mathbf{z}_k)$ y el hecho de no tener que proyectar, en cada la iteración, la dirección cuasi Newton inexacta calculada en **P.2** permiten inferir que el algoritmo CNIDP es un algoritmo competitivo con respecto a los métodos tipo Newton conocidos hasta el momento para resolver Problemas de Complementariedad Horizontal y Problemas de Complementariedad no Lineal.

Con el objetivo de ratificar los resultados teóricos demostrados en este capítulo y evaluar las expectativas generadas alrededor del algoritmo CNIDP, en el siguiente capítulo, presentamos los resultados numéricos obtenidos al resolver algunos problemas de Complementariedad no Lineal reformulándolos como Problemas de Complementariedad Horizontal y comparamos estos resultados con los obtenidos al resolver los mismos problemas con el algoritmo tipo Newton propuesto en [2].

CAPÍTULO 4

EXPERIMENTOS NUMÉRICOS

En este capítulo, reportamos los resultados obtenidos tras realizar experimentos numéricos con varios problemas que permiten resaltar las fortalezas del algoritmo *CNIDP*. Con este propósito, los códigos de los algoritmos fueron escritos en MATLAB® y realizamos las pruebas numéricas en un computador con procesador Intel(R) Core(TM) i7 con 3 GHz en cada núcleo y una RAM de 16 GB.

Como criterio de parada, usamos $\|G(\mathbf{z})\| \leq \varepsilon = 10^{-6}$. Los valores para los parámetros requeridos en la implementación del algoritmo fueron los siguientes: la norma máxima permitida para $\mathbf{d}^{CN}$ en **P.2** fue de 10^4 ($c_{big} = 10^4$), la cota inferior para el tamaño de paso máximo en **P.3** fue $c_{small} = 10^{-4}$. La contracción en cada iteración interna de la búsqueda lineal en **P.4** y **P.5** fue $\beta = 0.5$. De igual manera, en **P.4** y **P.5**, $\lambda = 10^{-4}$ y $\sigma_k = 1/k^2$. La contracción que permite mantener las iteraciones factibles (en el interior de Ω) fue $\tau_k = \tau = 0.9995$; el número máximo de iteraciones que permitimos realizar fue $N = 1000$ y el procedimiento con el que hallamos la dirección cuasi Newton inexacta en **P.2** fue el de gradientes conjugados (cgs), aunque también presentamos algunos resultados positivos al realizar las pruebas con otros procedimientos incluidos en MATLAB® para resolver (3.5).

Uno de los aspectos más importantes en la implementación del algoritmo *CNIDP* es la

forma cómo se deben actualizar las aproximaciones a la matriz jacobiana G' en cada iteración. Teniendo en cuenta la hipótesis **H1**, hemos realizado los experimentos numéricos haciendo uso de las actualizaciones secantes de cambio mínimo. Estas actualizaciones satisfacen la ecuación secante y su cambio, medido en alguna norma, es mínimo con respecto a la aproximación actual B_k .

Con respecto a esta familia de actualizaciones, se debe resaltar que el hecho de que satisfagan la ecuación secante y el cambio entre una aproximación y otra sea mínimo, permiten que la sucesión de matrices $\{B_k\}$ satisfagan la propiedad de deterioración acotada [10][22], la cual garantiza que las matrices de la sucesión permanezcan en una vecindad de $G'(\mathbf{z}^*)$. Esta propiedad, en general, es muy importante ya que esta nos permite, por ejemplo, garantizar que la sucesión $\{\|B_k\|\}$ o incluso, la sucesión $\{\|B_k^{-1}\|\}$, es acotada.

En particular, hemos llevado a cabo nuestros experimentos numéricos haciendo uso de la actualización “mala” de Broyden [9], la cual tiene la siguiente forma,

$$B_{k+1} = B_k + \frac{(\mathbf{y}_k - B_k \mathbf{s}_k) \mathbf{y}_k^T B_k}{\mathbf{y}_k^T \mathbf{y}_k}, \quad (4.1)$$

donde $\mathbf{y}_k = G(\mathbf{z}_{k+1}) - G(\mathbf{z}_k)$ y $\mathbf{s}_k = \mathbf{z}_{k+1} - \mathbf{z}_k$.

A continuación, reportaremos los resultados obtenidos al resolver algunos problemas de prueba generados de forma aleatoria, el Problema de los Valores Propios Complementarios y un problema emergente en el campo de la economía. De igual forma, comparamos los resultados obtenidos con el algoritmo *CNIDP* con los resultados obtenidos al resolver los problemas con tres algoritmos más. El primero de ellos, es una versión similar al *CNIDP*, sin embargo, en este, la dirección en (3.5) es obtenida de forma exacta, dicho algoritmo lo denominaremos *CNDP*. El segundo de los algoritmos con los cuales compararemos el desempeño de la propuesta hecha en esta tesis es el algoritmo propuesto en [2], este es un método tipo Newton inexacto en el cual se hace uso del gradiente proyectado cuando la dirección calculada en **P2** no satisface los requerimientos necesarios, este método lo denotaremos *PGIN*. El tercer algoritmo con el cual compararemos el método *CNIDP* es una versión de este en el que se hace uso de la matriz jacobiana exacta, es decir, éste también es un método tipo Newton. Este método será denotado *NIDP*.

En las tablas en las que reportamos los resultados, usamos las siguientes convenciones:

Método: denota el algoritmo usado para resolver los problemas; estos serán *CNIDP*, *CNDP*, *PGIN* o *NIDP*.

k : es el número de iteraciones que realizaron los algoritmos para resolver los problemas.

It. In.: identifica el número de iteraciones internas llevadas a cabo por el método cgs o sus similares en los algoritmo *CNIDP*, *NIDP* y *PGIN*.

Proy.: indica el número de iteraciones en las que los algoritmos hicieron uso de sus respectivas direcciones proyectadas.

z_0 : es el punto inicial con el cual se probó cada algoritmo.

-- : indica que el método no convergió.

n : denota el tamaño de cada problema.

Act.: indica la actualización secante de cambio mínimo usada: BB (Broyden Buena) y BM (Broyden Mala).

$$RelRes = \frac{\|z^{k+1} - z^*\|}{\|z^k - z^*\|}.$$

t : denota el tiempo cpu de cada uno de los algoritmos cuando éstos resolvieron los problemas propuestos.

En lo que sigue, presentamos una breve descripción de cada uno de los problemas y los resultados obtenidos para cada uno de ellos.

4.1. Funciones de prueba

En esta sección, describiremos dos problemas con los cuales se llevaron a cabo los experimentos numéricos aquí presentados. Cabe mencionar que esta clase de problemas son clásicos en la realización de experimentos numéricos concernientes al campo de la complementariedad y han sido usados por varios autores para probar sus algoritmos [33, 34, 49, 55]

El primer Problema de Complementariedad no Lineal está dado por la función $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$, definida por

$$F_i(\mathbf{x}) = -x_{i+1} + 2x_i - x_{i-1} + \frac{1}{3}x_i^3 + 1, \quad i = 1, \dots, n \quad (4.2)$$

en la que, $x_0 = x_{n+1} = 0$ [34].

De acuerdo a lo mencionado en el capítulo anterior, este PCNL(F) es equivalente al PCH(H)

$$\begin{aligned} F(\mathbf{x}) - \mathbf{w} &= 0 \\ \mathbf{x}^T \mathbf{w} &= 0 \\ \mathbf{x}, \mathbf{w} &\geq 0, \end{aligned}$$

donde $H(\mathbf{x}, \mathbf{w}) = F(\mathbf{x}) - \mathbf{w}$.

En la **Tabla 4.1**, mostramos los resultados obtenidos al resolver este problema, variando su tamaño, haciendo uso de la actualización *Broyden* mala y tomando como punto inicial el vector de unos del tamaño apropiado.

<i>Método</i>	<i>n</i>	<i>k</i>	<i>It. In.</i>	<i>Proy.</i>	<i>t (s)</i>
<i>CNIDP</i>	100	15	105	6	0.1094
CNDP	100	15	*	6	0.0781
PGIN	100	121	1381	33	0.7031
NIDP	100	14	92	5	0.1250
<i>CNIDP</i>	1000	18	178	8	1.3594
CNDP	1000	16	*	10	1.6250
PGIN	1000	115	1129	51	7.8281
NIDP	1000	13	141	5	0.9844
<i>CNIDP</i>	5000	13	114	6	20.4848
CNDP	5000	13	*	7	45.4688
PGIN	5000	—	—	—	—
NIDP	5000	48	56	44	76.4531
<i>CNIDP</i>	10000	18	291	4	112.3750
CNDP	10000	13	*	7	305.4219
PGIN	10000	—	—	—	—
NIDP	10000	48	56	44	282.5156
<i>CNIDP</i>	25000	23	349	4	801.5313
CNDP	25000	—	*	—	—
PGIN	25000	—	—	—	—
NIDP	25000	—	—	—	—

Tabla 4.1: Algoritmos *CNIDP*, CNDP, PGIN y NIDP para el PCNL(F) 4.2.

Como se puede ver, el método *CNIDP* fue exitoso en todos los casos, notándose una superioridad, en términos de tiempo, cuando el problema a resolver fue de gran tamaño ($n = 5000, 10000$ y 25000). De igual manera, queremos destacar las pocas iteraciones internas de cgs que efectuaron los métodos *CNIDP* y *NIDP* en relación al tamaño de cada problema y el número de iteraciones externas de cada uno de ellos. En este mismo sentido, es satisfactorio notar la poca cantidad de proyecciones que realizó el algoritmo *CNIDP* con respecto al número total de iteraciones necesarias para resolver cada problema.

Con el ánimo de analizar el comportamiento en detalle del algoritmo *CNIDP*, en la **Tabla 4.2**, mostramos los valores obtenidos para α_k^{max} , α_k^{break} , α_k y $RelRes$ en cada iteración al resolver el anterior problema con tamaño $n = 10000$.

k	α_k^{break}	$\alpha_k^{m\acute{a}x}$	α_k	θ	$RelRes$
1	0.2904	0.2903	0.1451	0.7	0.9965
2	1	0.9995	0.2498	7×10^{-11}	0.9832
3	0.4914	0.4911	0.2455	7×10^{-11}	0.9731
4	*	0.9995	0.9995	7×10^{-11}	0.9650
5	*	0.9995	0.9995	7×10^{-11}	0.8731
6	0.0037	0.0036	0.0036	7×10^{-11}	0.6540
7	0.6704	0.6700	0.1675	7×10^{-11}	0.3211
8	0.0103	0.0102	0.0102	7×10^{-11}	0.2731
9	0.0041	0.0040	0.0040	7×10^{-11}	0.2321
10	*	0.9995	0.9995	7×10^{-11}	0.1516
11	*	0.9995	0.9995	7×10^{-11}	0.0321
12	0.0359	0.0358	0.0358	7×10^{-11}	0.0121
13	0.6006	0.6002	0.6002	7×10^{-11}	0.0102
14	0.7128	0.7124	0.7124	7×10^{-11}	0.0087
15	0.8024	0.8019	0.8019	7×10^{-11}	0.0056
16	0.8846	0.8842	0.8842	7×10^{-11}	0.0021
17	0.9366	0.9361	0.9361	7×10^{-11}	0.0005
18	0.9412	0.9407	0.9407	7×10^{-11}	0.0001

Tabla 4.2: Algoritmo *CNIDP* con $n = 10000$ y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$.

En esta tabla, el símbolo * indica que en la respectiva iteración, el método *CNIDP* hizo uso de la dirección cuasi Newton proyectada. Como se puede ver, este experimento ratifica los resultados demostrados en los **Teoremas 3.1, 3.4 y 3.5**, ya que solo fue

necesario proyectar algunas de las primeras iteraciones y los tamaños de paso máximo α_k^{break} y α_k^{max} convergen a 1. De igual manera, el hecho que $RelRes \rightarrow 0$ indica una tasa de convergencia superlineal, acorde con lo demostrado en el **Teorema 3.6**.

Ahora bien, teniendo en cuenta que la forma en la que se aproximan las matrices jacobianas y que el método inexacto para encontrar $\mathbf{d}_k$ en **P.2** son dos factores que pueden alterar notablemente el desempeño del algoritmo, a continuación presentamos los resultados numéricos obtenidos al variar estos dos factores.

Cabe mencionar que, teóricamente, se ha demostrado que las direcciones obtenidas a través del método de gradientes conjugados (cgs) son direcciones de descenso para la función f del problema de minimización (3.4); sin embargo, hasta el momento, no se han obtenido resultados teóricos acerca del comportamiento de las direcciones obtenidas por algún otro método. Por esta razón, hemos decidido presentar en este documento los avances numéricos obtenidos en esta dirección.

<i>Método</i>	<i>n</i>	<i>k</i>	<i>It. In.</i>	<i>Proy.</i>	<i>t (s)</i>
cgs	100	15	105	6	0.1094
gmres	100	11	30	5	0.0156
bicg	100	12	43	4	0.0156
bicgstab	100	10	23	4	0.0156
cgs	1000	18	178	8	1.3594
gmres	1000	14	82	5	0.5781
bicg	1000	17	179	7	1.3750
bicgstab	1000	19	174	9	1.7031
cgs	5000	13	114	6	20.4848
gmres	5000	—	—	—	—
bicg	5000	16	141	8	26.8438
bicgstab	5000	—	—	—	—
cgs	10000	18	291	4	112.3750
gmres	10000	—	—	—	—
bicg	10000	17	181	7	108.1250
bicgstab	10000	—	—	—	—

Tabla 4.3: Algoritmo *CNIDP*, Act. BM y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$.

<i>Método</i>	<i>n</i>	<i>k</i>	<i>It.</i>	<i>In.</i>	<i>Proy.</i>	<i>t (s)</i>
cgs	100	13	62	8	8	0.1719
gmres	100	13	41	8	8	0.1250
bicg	100	9	63	5	5	0.0625
bicgstab	100	11	88	4	4	0.0156
cgs	1000	14	56	10	10	1.0313
gmres	1000	16	52	10	10	0.9688
bicg	1000	10	62	6	6	0.6563
bicgstab	1000	9	38	6	6	0.5469
cgs	5000	18	39	15	15	30.0313
gmres	5000	14	32	10	10	12.2969
bicg	5000	10	62	6	6	15.2344
bicgstab	5000	9	61	5	5	13.8281
cgs	10000	18	39	15	15	112.8750
gmres	10000	15	32	11	11	45.1875
bicg	10000	10	62	6	6	58.3438
bicgstab	10000	13	141	5	5	78.2813

Tabla 4.4: Algoritmo *CNIDP*, Act. BB y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$.

En las **Tablas 4.3** y **4.4**, reportamos los resultados obtenidos al resolver, con el algoritmo *CNIDP*, el PCNL(F) determinado por la función F definida en (4.2) al variar el método con el cual fue calculada la dirección en **P.2**. En la primera de ellas, se muestran los resultados cuando la actualización usada para aproximar las matrices jacobianas fue la actualización **Broyden** mala y, en la segunda, se muestran los resultados al usar la actualización **Broyden** buena.

Estos experimentos numéricos sugieren que podría ser posible hacer uso del algoritmo *CNIDP* cuando éste se implemente con algún método inexacto diferente al cgs. En muchos de los casos incluso, tanto el número de iteraciones externas como el número de iteraciones internas y el tiempo de ejecución se vio considerablemente disminuido. Cabe resaltar que estos son apenas unos resultados preliminares y que por tanto abren la puerta a futuras investigaciones en esta dirección.

Por otro lado, con el ánimo de evaluar la cualidad de convergencia global del algoritmo *CNIDP*, realizamos pruebas con éste y el PCNL(F) determinado por la función F definida en (4.2) variando el punto inicial con el cual se ejecutó. Los resultados obtenidos son presentados en la **Tabla 4.5**.

<i>Método</i>	$\mathbf{z}_0$	k	<i>It. In.</i>	<i>Proy.</i>	t (s)
<i>CNIDP</i>	$0.1 * (1, 1, \dots, 1)^T$	11	11	8	1.2012
CNDP	$0.1 * (1, 1, \dots, 1)^T$	35	*	12	2.9000
PGIN	$0.1 * (1, 1, \dots, 1)^T$	111	1362	23	3.7304
NIDP	$0.1 * (1, 1, \dots, 1)^T$	12	44	4	0.4844
<i>CNIDP</i>	$5 * (1, 1, \dots, 1)^T$	27	58	7	1.4063
CNDP	$5 * (1, 1, \dots, 1)^T$	36	*	23	3.7304
PGIN	$5 * (1, 1, \dots, 1)^T$	—	—	—	—
NIDP	$5 * (1, 1, \dots, 1)^T$	11	13	5	1.2813
<i>CNIDP</i>	$10 * (1, 1, \dots, 1)^T$	19	62	7	0.7813
CNDP	$10 * (1, 1, \dots, 1)^T$	—	*	—	—
PGIN	$10 * (1, 1, \dots, 1)^T$	—	—	—	—
NIDP	$10 * (1, 1, \dots, 1)^T$	12	44	4	1.4375
<i>CNIDP</i>	$50 * (1, 1, \dots, 1)^T$	59	242	16	1.5629
CNDP	$50 * (1, 1, \dots, 1)^T$	—	*	—	—
PGIN	$50 * (1, 1, \dots, 1)^T$	—	—	—	—
NIDP	$50 * (1, 1, \dots, 1)^T$	11	41	4	2.3594
<i>CNIDP</i>	$100 * (1, 1, \dots, 1)^T$	102	315	21	4.3594
CNDP	$100 * (1, 1, \dots, 1)^T$	—	*	—	—
PGIN	$100 * (1, 1, \dots, 1)^T$	—	—	—	—
NIDP	$100 * (1, 1, \dots, 1)^T$	51	60	3	7.4063

Tabla 4.5: Variación de $\mathbf{z}_0$, $n = 1000$.

Como se puede ver, los algoritmos *CNIDP* y NIDP fueron exitosos en todos los casos, sin embargo, el método *CNIDP* fue más rápido, en términos de tiempo, que su contraparte, el método Newton inexacto.

En conclusión, los experimentos llevados a cabo con el PCNL(F) determinado por la función F definida en (4.2), nos permitieron ratificar algunos de los resultados teóricos más importantes demostrados en el capítulo anterior. De igual forma, nos permitieron evaluar su característica de ser un método global e incluso, nos permitieron generar algunas hipótesis con respecto al uso de diversos métodos inexactos para la implementación de algoritmos tipo Newton inexactos.

El segundo Problema de Complementariedad no Lineal con el cual realizamos diversas

pruebas numéricas corresponde al problema definido por la función $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ donde

$$F_i(\mathbf{x}) = x_i \left(\sum_{j=1}^n x_j \right) - n. \quad (4.3)$$

Vale la pena destacar que la matriz jacobiana de F es una matriz densa y que por lo tanto no es posible, en este caso, tomar ventaja de la estructura de la misma. De igual manera, es importante mencionar que en este problema se hizo uso del paquete de diferenciación automática de MATLAB[®] para calcular la respectiva matriz Jacobiana en los casos en los que esta fue requerida. En la **Tabla 4.6**, mostramos los resultados obtenidos al resolver el PCNL(F) definido por (4.3), variando su tamaño y tomando como vector inicial al vector $\mathbf{z}_0 = 10 \times (1, 1, \dots, 1)^T$.

<i>Método</i>	<i>n</i>	<i>k</i>	<i>It. In.</i>	<i>Proy.</i>	<i>t (s)</i>
<i>CNIDP</i>	10	3	4	0	0.0003
CNDP	10	3	*	0	0.0003
PGIN	10	4	6	1	0.1875
NIDP	10	3	4	0	0.1875
<i>CNIDP</i>	30	5	6	0	0.0008
CNDP	30	5	*	0	0.0313
PGIN	30	4	7	1	1.2188
NIDP	30	4	6	0	1.1563
<i>CNIDP</i>	50	5	6	0	0.0013
CNDP	50	5	*	0	0.0518
PGIN	50	4	7	1	4.2813
NIDP	50	4	6	0	4.1875
<i>CNIDP</i>	100	5	6	0	0.0064
CNDP	100	5	*	0	0.0619
PGIN	100	4	7	1	27.5781
NIDP	100	4	6	0	27.7969
<i>CNIDP</i>	200	5	6	4	0.0215
CNDP	200	5	*	0	0.1250
PGIN	200	4	7	1	202.9375
NIDP	200	4	6	0	202.1094
<i>CNIDP</i>	500	5	6	0	0.0781
CNDP	500	5	*	0	0.3221
PGIN	500	4	7	1	2.964×10^3
NIDP	500	4	6	0	2.964×10^3

Tabla 4.6: Algoritmos *CNIDP*, CNDP, PGIN y NIDP para el PCNL.

Con estos resultados, podemos apreciar el alto costo computacional de los métodos tipo Newton en contraste con la rapidez, en términos de tiempo, de los métodos cuasi Newton. Este es un claro ejemplo que sustenta la necesidad de proponer estrategias basadas en métodos tipo Newton que no tengan la necesidad de hacer uso de las derivadas de las funciones en juego. Es evidente que en muchos de los problemas de la vida real que son modelados matemáticamente, es muy difícil acceder a una expresión que represente sus derivadas e incluso, cuando es posible acceder a ellas, el costo computacional es bastante elevado, como se pudo apreciar en este ejemplo, de aquí la razón para proponer y hacer uso de métodos cuasi Newton.

En la **Tabla 4.7**, analizaremos el comportamiento detallado del algoritmo *CNIDP* cuando este resolvió el anterior PCNL, de tamaño $n = 200$ y tomando como punto inicial el vector $\mathbf{z}_0 = 50 \times (1, 1, \dots, 1)^T$.

k	α_k^{break}	$\alpha_k^{m\acute{a}x}$	α_k	θ	$RelRes$
1	1	0.9995	0.9995	0.5000	0.9765
2	0.5541	0.5539	0.2769	0.2500	0.9324
3	1	0.9995	0.9995	0.1111	0.7115
4	0.2164	0.2163	0.2163	0.0625	0.6514
5	1	0.9995	0.9995	0.0400	0.5321
6	1	0.9995	0.9995	0.0278	0.4184
7	1	0.9995	0.9995	0.0204	0.2956
8	1	0.9995	0.9995	0.0156	0.1503
9	1	0.9995	0.9995	0.0123	0.0956
10	1	0.9995	0.9995	0.0100	0.0542
11	1	0.9995	0.9995	0.0083	0.0125
12	1	0.9995	0.9995	0.0069	0.00068

Tabla 4.7: Algoritmo *CNIDP* con $n = 200$ y $\mathbf{z}_0 = 50 \times (1, 1, \dots, 1)^T$.

Estos resultados permiten ratificar nuevamente los hechos teóricos demostrados en los **Teoremas 3.1, 3.4 y 3.5** ya que, en este caso, el algoritmo *CNIDP* aceptó la dirección cuasi Newton en todas su iteraciones, es decir, no fue necesario la proyección de las mismas sobre el ortante positivo de $\mathbb{R}^n$. De igual manera, podemos observar que $\alpha_k^{m\acute{a}x} = \alpha_k$ converge a 1 y además que $RelRes$ converge a cero a medida que θ converge a cero, lo cual refleja la convergencia superlineal del algoritmo.

Finalizaremos los experimentos numéricos con el PCNL(F) definido por (4.3) mostrando,

en la **Tabla 4.8**, el comportamiento del algoritmo *CNIDP* cuando este fue ejecutado desde distintos puntos iniciales. Para ello, realizamos pruebas con 500 puntos iniciales seleccionados de forma aleatoria y cuyas componentes se encuentran distribuidas uniformemente en el intervalo $[0, 10]$. En la **Tabla 4.8**, mostramos el porcentaje de éxito del algoritmo *CNIDP* para cada uno de los problemas de distinto tamaño y comparamos los resultados con el algoritmo cuasi Newton local para resolver Problemas de Complementariedad no Lineal presentado en [4].

n	<i>CNIDP</i>	<i>Alg. Local</i>
10	100 %	62.2 %
100	100 %	51.2 %
500	99.8 %	61 %
1000	99.6 %	44 %

Tabla 4.8: Análisis de globalidad del algoritmo *CNIDP*.

Como se puede ver, el método *CNIDP* convergió, en promedio, en un 99.8 % de los experimentos realizados mientras su contraparte, el algoritmo local, convergió en promedio, en un 54.6 % de los experimentos realizados. Estos experimentos muestran la importancia de incluir estrategias de globalización a los algoritmos propuestos para resolver, no solo Problemas de Complementariedad, sino problemas más generales.

En la siguiente sección, mostramos y analizamos los resultados obtenidos al resolver un problema reconocido por la comunidad matemática como un problema complejo.

4.2. Valores Propios Complementarios

Dada una matriz $A \in \mathbb{R}^{n \times n}$ y una matriz definida positiva $B \in \mathbb{R}^{n \times n}$, el problema de los valores propios complementarios (EiCP por sus siglas en inglés), consiste en hallar un escalar $\lambda > 0$ y un vector $\mathbf{x} \in \mathbb{R}^n \setminus \{0\}$ tal que

$$\begin{aligned} \mathbf{w} &= (\lambda B - A)\mathbf{x}, \\ \mathbf{x}^T \mathbf{w} &= 0, \\ \mathbf{x} &\geq 0, \quad \mathbf{w} \geq 0. \end{aligned} \tag{4.4}$$

Este problema surge en diversas aplicaciones de la ingeniería y la física, por ejemplo

en el estudio de la frecuencia de resonancia de estructuras y la estabilidad de sistemas dinámicos [17].

Para cualquier solución $(\lambda, \mathbf{x})$ del EiCP, λ es llamado *valor propio complementario* del par (A, B) y $\mathbf{x}$ es su correspondiente *vector propio complementario*. Ahora bien, como el conjunto de vectores propios complementarios asociados a un valor propio complementario es un cono entonces no hay pérdida de generalidad al considerar solo las soluciones que satisfagan $\|\mathbf{x}\|_1 = \mathbf{e}^T \mathbf{x} = p$ con $p > 0$ y $\mathbf{e}$ un vector de unos.

En este problema, la estructura de la matriz A juega un rol importante. Por ejemplo, el caso en el que las matrices A y B son simétricas ha sido ampliamente estudiado en [36, 37, 57] y en estos y otros artículos los autores muestran que el EiCP es equivalente a hallar un punto estacionario de un programa de optimización diferenciable que envuelve el cociente *Rayleigh* en un simplex. Por otro lado, el caso en el que B es simétrica y A es una matriz no simétrica es abordado en [36, 38] y en sus artículos los autores proponen diversas formas de abordar el problema. Proponen reescribir el problema como un Problema de Programación Matemática con restricciones de complementariedad o reescribirlo como un problema de optimización global o como un Problema de Complementariedad no Lineal. Para nuestros experimentos, abordaremos el caso en el que A no es simétrica y transformaremos el EiCP (4.4) en un problema de complementariedad no lineal; para tal fin, observemos que (4.4) puede ser reescrito como

$$\begin{aligned} \mathbf{w} &= B\mathbf{x} - x_{n+1}A\mathbf{x}, \\ \mathbf{e}^T \mathbf{x} &= p \\ \mathbf{x}^T \mathbf{w} &= 0, \\ \mathbf{x} &\geq 0, \quad \mathbf{w} \geq 0, \quad x_{n+1} \geq 0, \end{aligned}$$

donde $x_{n+1} = \frac{1}{\lambda}$. Por lo tanto, si una nueva variable no negativa w_{n+1} se introduce y es tal que

$$w_{n+1} = \mathbf{e}^T \mathbf{x} - p, \quad \text{y} \quad x_{n+1}w_{n+1} = 0$$

entonces el EiCP (4.4) es equivalente al PCNL(F), donde $F : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+1}$ está definida por

$$F(\mathbf{x}, x_{n+1}) = \begin{pmatrix} B\mathbf{x} - x_{n+1}A\mathbf{x} \\ \mathbf{e}^T \mathbf{x} - p \end{pmatrix}. \quad (4.5)$$

Para llevar a cabo nuestros experimentos, hemos seguido las indicaciones dadas en [38] con el ánimo de comparar nuestros resultados con los obtenidos por los autores. Para

tal fin, $B = I_n$ y A fue generada de forma aleatoria de tal manera que sus elementos estuvieran uniformemente distribuidos en el intervalo $[0, 1]$ o $[-50, 50]$. Estos dos tipos de experimentos los agrupamos en los conjuntos A1 y A2, respectivamente.

Cabe mencionar que en esta sección compararemos el algoritmo CNIDP con los algoritmos PATH [25] y CASET [39] e incluiremos en este comparativo los resultados obtenidos al resolver los mismos problemas con el algoritmo PGIN.

En la **Tabla 4.9**, encontramos el desempeño de los algoritmos antes mencionados en términos de número de iteraciones y tiempo CPU cuando estos intentaron resolver los problemas del grupo A1.

n	Método	k	$t(s)$	n	Método	k	$t(s)$
3	CNIDP	33	0.01	30	CNIDP	291	0.46
	PGIN	11	0.01		PGIN	15	0.03
	CASET	7	0.00		CASET	63	0.00
	PATH	–	–		PATH	136	0.92
6	CNIDP	40	0.01	40	CNIDP	294	0.70
	PGIN	9	0.01		PGIN	23	0.02
	CASET	14	0.00		CASET	85	0.00
	PATH	105	0.91		PATH	133	0.80
10	CNIDP	66	0.04	50	CNIDP	282	0.64
	PGIN	9	0.00		PGIN	20	0.06
	CASET	25	0.00		CASET	102	0.01
	PATH	101	0.85		PATH	–	–
20	CNIDP	120	0.39	100	CNIDP	290	0.85
	PGIN	13	0.01		PGIN	34	0.02
	CASET	53	0.00		CASET	203	0.05
	PATH	134	0.97		PATH	212	1.21

Tabla 4.9: Resultados grupo A1.

Con este grupo de experimentos, podemos destacar el hecho de que el algoritmo *CNIDP* convergió en todos los experimentos y lo hizo en un rango de tiempo aceptable. Si bien no fue el más rápido de los algoritmos, tuvo un desempeño mejor que el algoritmo *PATH* el cual es ampliamente usado para resolver problemas que envuelven condiciones de complementariedad.

En la **Tabla 4.10**, podemos ver los resultados obtenidos por los algoritmos al resolver el conjunto de problemas pertenecientes al grupo A2.

n	Método	k	$t(s)$	n	Método	k	$t(s)$
3	<i>CNIDP</i>	31	0.01	30	<i>CNIDP</i>	287	0.91
	PGIN	15	0.04		PGIN	216	0.14
	CASET	3	0.05		CASET	183	0.02
	PATH	–	–		PATH	–	–
6	<i>CNIDP</i>	61	0.03	40	<i>CNIDP</i>	315	2.81
	PGIN	14	0.01		PGIN	251	2.83
	CASET	7	0.03		CASET	241	0.00
	PATH	–	–		PATH	–	–
10	<i>CNIDP</i>	136	0.07	50	<i>CNIDP</i>	387	3.34
	PGIN	68	0.03		PGIN	–	–
	CASET	25	0.02		CASET	323	0.03
	PATH	–	–		PATH	–	–
20	<i>CNIDP</i>	292	0.39	100	<i>CNIDP</i>	1303	6.17
	PGIN	165	0.78		PGIN	–	–
	CASET	71	0.02		CASET	1050	0.19
	PATH	–	–		PATH	–	–

Tabla 4.10: Resultados grupo A2.

En este caso, vemos nuevamente al algoritmo *CNIDP* como un algoritmo competitivo con respecto a los algoritmos PGIN y PATH. De igual forma, destacamos el hecho de que el algoritmo *CNIDP* convergió en el 100% de los experimentos. Sin embargo, debemos decir que para el experimento con $n = 100$ fue necesario extender el número máximo de iteraciones permitidas a 1500 para lograr resultados satisfactorios tanto con el algoritmo *CNIDP* como con el algoritmo CASET.

En la siguiente sección, mostraremos y analizaremos los resultados obtenidos al resolver un problema de complementariedad no lineal emergente en el campo de la economía.

4.3. Equilibrio económico

En esta sección, analizaremos los resultados obtenidos al resolver el problema de equilibrio económico conocido como “*Spatial Price Equilibrium Problem*” [61].

Este problema consiste en hallar el punto de equilibrio entre el flujo de cierta mercancía por una red vial y el precio de esta en las distintas tiendas en las que debe ser distribuida. A continuación, incluimos una tabla con la información de la notación usada para este problema.

<i>Símbolo</i>	<i>Significado</i>
N	Número de nodos (tiendas) en la red.
$T(n)$	Número de arcos que salen del nodo n .
$H(n)$	Número de arcos que llegan al nodo n .
$A = [A_{na}]$	Matriz de incidencia nodo/arco, $A_{na} = \begin{cases} 1 & \text{si } a \in T(n) \\ -1 & \text{si } a \in H(n) \\ 0 & \text{en otro caso} \end{cases}$
P_{ij}	Conjunto de caminos entre los nodos i y j .
$p \in P_{ij}$	Camino entre los nodos i y j .
$h_p, \mathbf{h}$	Flujo en el camino p y vector de flujo en los caminos respectivamente.
$f_a, \mathbf{f}$	Flujo en el arco a y vector de flujo de arcos respectivamente.
$C_a(\mathbf{f})$	Precio del servicio de transporte sobre el arco a .
$\pi_n, \boldsymbol{\pi}$	Precio del producto en el nodo n y vector de precios respectivamente.
$S_n(\boldsymbol{\pi})$	Suministro del producto en el nodo n .
$D_n(\boldsymbol{\pi})$	Demanda del producto en el nodo n .

Tabla 4.11: Notación *spatial price problem*.

De acuerdo a lo presentado en [61], el problema que nos ocupa en esta sección satisface las siguientes condiciones:

1. No negatividad de precios y flujo en los arcos,

$$f_a \geq 0, \quad D_n \geq 0, \quad S_n \geq 0, \quad \pi_n \geq 0 \quad \forall n, a.$$

2. Conservación de los suministros, demandas y flujo en los arcos,

$$D_n - S_n + \sum_j \sum_{p \in P_{nj}} h_p - \sum_i \sum_{p \in P_{in}} h_p = 0 \quad \forall n.$$

3. El precio del producto entregado es igual al precio del producto local si el flujo en el arco es positivo,

$$f_a \geq 0, \quad a = (i, j) \implies \pi_i + c_a = \pi_j \quad \forall a.$$

4. El flujo en el arco a es cero si el precio de entrega excede el precio local,

$$\pi_i + c_a > \pi_j, \quad a = (i, j) \implies f_a = 0 \quad \forall a.$$

En [32], los autores mostraron que si la función de costo sobre los arcos $C_a(\mathbf{f})$ es positiva y las funciones de suministro y demanda son tales que

$$\pi_n = 0 \implies D_n(\boldsymbol{\pi}) - S_n(\boldsymbol{\pi}) \geq 0$$

entonces $(\mathbf{f}, S, D, \boldsymbol{\pi})$ es una solución del problema *spatial price equilibrium* si y solo si resuelve el PCNL(F), donde F es la función definida por

$$F(\mathbf{x}) = F \begin{pmatrix} \mathbf{f} \\ \boldsymbol{\pi} \end{pmatrix} = \begin{pmatrix} C(\mathbf{f}) + A^T \boldsymbol{\pi} \\ S(\boldsymbol{\pi}) - D(\boldsymbol{\pi}) + A\mathbf{f} \end{pmatrix} \quad (4.6)$$

La notación usada hasta el momento parece ser para el transporte de un solo producto sin embargo, dado que en general las funciones usadas están acopladas, las formulaciones para múltiples productos se pueden realizar mediante el uso de múltiples copias de la red.

En nuestros experimentos, hemos supuesto dos casos: uno en el que se debe distribuir un único producto y otro en el que se deben distribuir dos productos entre 5 tiendas (nodos).

La red diseñada para estos experimentos se muestra en la figura (4.1)

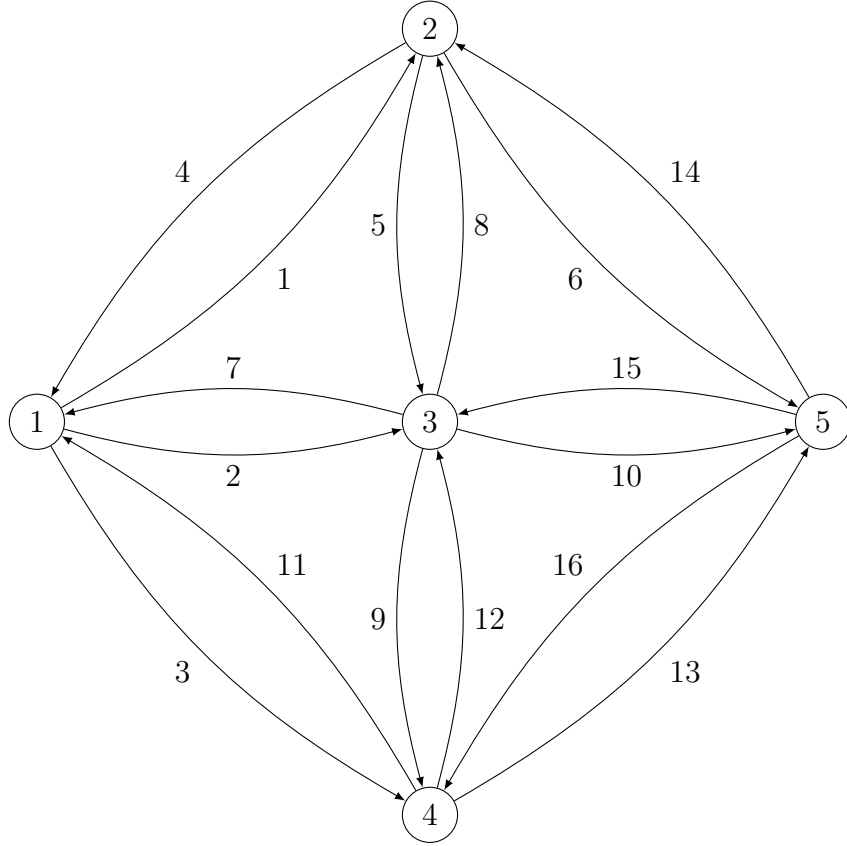


Figura 4.1: Red de distribución.

Las funciones de costo, suministro y demanda son como las sugeridas en [61]. La función de costo tiene la forma

$$C_a^k(\mathbf{f}) = \Gamma_a^k + \Omega_a^k (f_a^k)^4 + \sum_{j \neq k} \Delta_a^{kj} f_a^j,$$

la función de suministro tiene la forma

$$S_n^k(\boldsymbol{\pi}) = B_n^k + J_n^k (\pi_n^k)^2 + \sum_{j \neq n} u_{nj}^k \pi_j^k$$

y la función de demanda es de la forma

$$D_n^k(\boldsymbol{\pi}) = E_n^k + G_n^k (\pi_n^k)^2 + \sum_{j \neq n} w_{nj}^k \pi_j^k.$$

Donde el superíndice k hace referencia al producto que será distribuido y los valores de los parámetros necesarios en cada una de las anteriores funciones son dados en las **Tablas 4.12, 4.13 y 4.14**.

a	k	Γ_a^k	Ω_a^k	Δ_a^{12}	Δ_a^{21}	a	k	Γ_a^k	Ω_a^k	Δ_a^{12}	Δ_a^{21}
1	1	1.0	2.00×10^{-4}	0.1	—	9	1	3.0	2.00×10^{-4}	0.2	—
	2	2.0	1.00×10^{-4}	—	0.1		2	1.0	2.00×10^{-4}	—	0.1
2	1	2.0	1.98×10^{-5}	0.2	—	10	1	1.0	1.98×10^{-5}	0.4	—
	2	4.0	3.96×10^{-5}	—	0.2		2	2.0	7.92×10^{-5}	—	0.2
3	1	3.0	3.00×10^{-4}	0.3	—	11	1	2.0	1.00×10^{-4}	0.1	—
	2	2.0	1.00×10^{-4}	—	0.4		2	4.0	4.00×10^{-4}	—	0.7
4	1	1.0	1.60×10^{-3}	0.2	—	12	1	3.0	3.00×10^{-5}	0.2	—
	2	1.0	4.80×10^{-3}	—	0.5		2	4.0	4.00×10^{-5}	—	0.2
5	1	2.0	2.00×10^{-4}	0.1	—	13	1	1.0	1.00×10^{-3}	0.3	—
	2	2.0	3.00×10^{-4}	—	0.6		2	4.0	4.00×10^{-3}	—	0.1
6	1	3.0	5.94×10^{-5}	0.4	—	14	1	2.0	2.00×10^{-3}	0.4	—
	2	4.0	3.96×10^{-5}	—	0.3		2	1.0	1.00×10^{-3}	—	1.0
7	1	1.0	4.00×10^{-4}	0.5	—	15	1	3.0	3.00×10^{-4}	0.2	—
	2	2.0	2.00×10^{-4}	—	0.2		2	6.0	6.00×10^{-4}	—	0.3
8	1	2.0	1.60×10^{-3}	1.0	—	16	1	1.0	1.00×10^{-3}	0.1	—
	2	2.0	1.60×10^{-3}	—	0.1		2	1.0	1.00×10^{-3}	—	0.2

Tabla 4.12: Parámetros para la función de costos.

$i \backslash j$	u_{ij}^1					u_{ij}^2				
	1	2	3	4	5	1	2	3	4	5
1	0.00	0.02	0.01	0.01	0.00	0.00	0.04	0.02	0.02	0.00
2	0.05	0.00	0.02	0.00	0.02	0.10	0.00	0.04	0.00	0.04
3	0.04	0.02	0.00	0.02	0.04	0.08	0.04	0.00	0.04	0.08
4	0.03	0.00	0.02	0.00	0.02	0.06	0.00	0.04	0.00	0.04
5	0.00	0.01	0.01	0.04	0.00	0.00	0.02	0.02	0.08	0.00

Tabla 4.13: Parámetros para las funciones de suministros y demanda.

Es de anotar que para todos los experimentos se tomó $w_{ij}^k = -u_{ij}^k$.

n	k	B_n^k	J_n^k	E_n^k	G_n^k
1	1	15.0	0.02	40.0	0.02
	2	20.0	0.03	30.0	0.03
2	1	20.0	0.04	60.0	0.04
	2	10.0	0.06	80.0	0.06
3	1	15.0	0.04	20.0	0.04
	2	10.0	0.06	60.0	0.06
4	1	40.0	0.02	60.0	0.02
	2	40.0	0.03	50.0	0.03
5	1	30.0	0.01	40.0	0.01
	2	10.0	0.02	80.0	0.02

Tabla 4.14: Parámetros para las funciones de suministros y demanda.

En la **Tabla 4.15**, presentamos los resultados al resolver el problema resultante de la red diseñada en la Figura (4.1) y suponiendo que se debe distribuir un único producto.

<i>Método</i>	$\mathbf{z}_0$	k	<i>It.</i>	<i>In.</i>	<i>Proy.</i>	t (s)
<i>CNIDP</i>	$0.1 * (1, 1, \dots, 1)^T$	35	159	13	0.1563	
CNDP	$0.1 * (1, 1, \dots, 1)^T$	—	*	—	—	
NIDP	$0.1 * (1, 1, \dots, 1)^T$	—	—	—	—	
<i>CNIDP</i>	$(1, 1, \dots, 1)^T$	42	202	15	0.1875	
CNDP	$(1, 1, \dots, 1)^T$	—	*	—	—	
NIDP	$(1, 1, \dots, 1)^T$	32	157	10	1.3721	
<i>CNIDP</i>	$5 * (1, 1, \dots, 1)^T$	39	161	13	0.1721	
CNDP	$5 * (1, 1, \dots, 1)^T$	10	*	1	0.0313	
NIDP	$5 * (1, 1, \dots, 1)^T$	23	112	6	1.1542	
<i>CNIDP</i>	$10 * (1, 1, \dots, 1)^T$	14	21	7	0.0469	
CNDP	$10 * (1, 1, \dots, 1)^T$	—	*	—	—	
NIDP	$10 * (1, 1, \dots, 1)^T$	9	92	3	0.3221	
<i>CNIDP</i>	$50 * (1, 1, \dots, 1)^T$	24	157	12	0.0933	
CNDP	$50 * (1, 1, \dots, 1)^T$	—	*	—	—	
NIDP	$50 * (1, 1, \dots, 1)^T$	—	—	—	—	

Tabla 4.15: Variación de $\mathbf{z}_0$. Un producto por distribuir.

Estos resultados ratifican al algoritmo *CNIDP* como un método global en el sentido que convergió a la solución del problema desde todos los puntos iniciales con los cuales fue probado. Por otro lado, como era de esperarse, en los casos en que los métodos cuasi Newton con dirección exacta, CNDP, y Newton con dirección inexacta, NIDP, convergieron, estos los hicieron en una cantidad inferior de iteraciones que el método *CNIDP*; sin embargo, este último fue más rápido en términos de tiempo que sus dos contrapartes.

De igual manera, destacamos el bajo número de iteraciones internas que debió realizar el algoritmo *CNIDP*, en cada uno de los ensayos, con respecto al tamaño del problema a resolver. En este sentido, el algoritmo *CNIDP* realizó en promedio 7 iteraciones internas por cada iteración externa, lo que equivale a 1/6 del tamaño original del problema.

En la **Tabla 4.16**, vemos el resultado de los experimentos llevados a cabo suponiendo que esta vez se deben distribuir dos productos. Esta suposición duplica el tamaño del problema y por ende la complejidad del mismo.

<i>Método</i>	$\mathbf{z}_0$	k	<i>It. In.</i>	<i>Proy.</i>	t (s)
<i>CNIDP</i>	$0.1 * (1, 1, \dots, 1)^T$	108	1101	18	0.2132
CNDP	$0.1 * (1, 1, \dots, 1)^T$	—	*	—	—
NIDP	$0.1 * (1, 1, \dots, 1)^T$	154	121	147	3.2136
<i>CNIDP</i>	$(1, 1, \dots, 1)^T$	109	981	21	0.2234
CNDP	$(1, 1, \dots, 1)^T$	—	*	—	—
NIDP	$(1, 1, \dots, 1)^T$	84	632	12	2.4215
<i>CNIDP</i>	$5 * (1, 1, \dots, 1)^T$	102	661	25	0.2233
CNDP	$5 * (1, 1, \dots, 1)^T$	65	*	31	0.5631
NIDP	$5 * (1, 1, \dots, 1)^T$	81	625	10	2.8722
<i>CNIDP</i>	$10 * (1, 1, \dots, 1)^T$	124	1321	32	0.2297
CNDP	$10 * (1, 1, \dots, 1)^T$	109	*	6	0.9393
NIDP	$10 * (1, 1, \dots, 1)^T$	101	934	21	3.4215
<i>CNIDP</i>	$50 * (1, 1, \dots, 1)^T$	193	1351	28	0.3733
CNDP	$50 * (1, 1, \dots, 1)^T$	78	*	23	0.5624
NIDP	$50 * (1, 1, \dots, 1)^T$	106	1034	9	3.5612

Tabla 4.16: Variación de $\mathbf{z}_0$. Dos productos por distribuir.

Estos resultados evidencian, nuevamente, una superioridad, en términos de tiempo, del algoritmo *CNIDP* frente a sus dos homólogos. Como era de esperarse, la complejidad

de este problema se vio reflejada en el incremento del número de iteraciones externas en cada uno de los experimentos. Con estos resultados, también podemos verificar el bajo número de iteraciones internas que el algoritmo *CNIDP* debió realizar por cada iteración externa; en este caso, fueron en promedio 11 iteraciones internas por cada iteración externa, lo que equivale aproximadamente a $1/8$ del tamaño del problema a resolver.

Con los experimentos llevados a cabo, podemos enmarcar el algoritmo *CNIDP* como un algoritmo competitivo y de bajo costo computacional, hecho que se ve reflejado en el tiempo de ejecución al resolver los problemas a los que fue enfrentado. Además, el hecho de resolver los problemas de prueba y desde distintos puntos de inicialización lo muestran como un método robusto. Si bien somos conscientes de que antes de emitir un juicio sobre el algoritmo *CNIDP*, es necesario realizar muchas mas pruebas, los resultados mostrados en este capítulo dejan entrever un futuro prometedor para el nuevo método propuesto.

Por otra parte, como resultados de los experimentos llevados a cabo, queremos destacar el hecho de dejar abiertos nuevos problemas de investigación tales como el uso de diversos métodos inexactos que garanticen un descenso en la función de mérito que se desea minimizar. Cabe resaltar que avances en esta dirección serán de gran aporte no solo a las temáticas concernientes a la complementariedad sino a la optimización en general.

En el siguiente capítulo, mostraremos cómo el algoritmo *CNIDP*, tras una pequeña modificación, es capaz de resolver sistemas de ecuaciones no lineales con restricciones de no negatividad, manteniendo la buenas propiedades de convergencia que se demostraron teóricamente en el **Capítulo 3**.

CAPÍTULO 5

SISTEMAS DE ECUACIONES CON RESTRICCIONES

Como lo habíamos mencionado en la introducción de este documento y como se ha podido ver a lo largo del desarrollo teórico del mismo, el algoritmo CNIDP podría eventualmente resolver sistemas de ecuaciones con restricciones de no negatividad. Claro está, es necesario incluir algunas modificaciones para tal fin, pero vale la pena decir que tales modificaciones no alterarán significativamente la demostración de los teoremas presentados en el **Capítulo 3**.

Con respecto a la clase de problemas que abordaremos en este capítulo, es importante decir que estos llaman la atención de la comunidad matemática debido a su aparición en diversos campos de aplicación. En particular, al finalizar este capítulo, presentaremos los resultados numéricos al resolver dos problemas emergentes en los campos de la química y la física nuclear.

Para ser más precisos, el problema que abordaremos en este capítulo será el de resolver el sistema de ecuaciones

$$G(\mathbf{z}) = 0, \quad \mathbf{z} \in \Omega, \quad (5.1)$$

donde $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ es una función continuamente diferenciable y

$$\Omega = \{z \in \mathbb{R}^n : z \geq 0\}.$$

Hemos decidido usar las mismas convenciones que las usadas en el **Capítulo 3** para identificar la función que define el sistema de ecuaciones y el conjunto en el cual deseamos hallar la solución; sin embargo, debemos tener en cuenta que, en este caso, G es una función cualquiera y Ω es el octante no negativo de $\mathbb{R}^n$.

En este sentido, el sistema de ecuaciones (5.1) puede ser reformulado como el problema de minimización con restricciones

$$\begin{array}{ll} \textit{Minimizar} & f(z) \\ \textit{s.a} & z \in \Omega, \end{array} \quad (5.2)$$

donde $f(z) = \frac{1}{2}\|G(z)\|^2$, y $\|\cdot\|$ denota la norma euclidiana.

Como se puede ver, matemáticamente, el problema (5.2) es idéntico al problema (3.4), de ahí nuestro interés en mostrar el algoritmo CNIDP como un algoritmo capaz de resolver sistemas de ecuaciones con restricciones de no negatividad.

Con el ánimo de no repetir las mismas líneas describiendo el algoritmo CNIDP expuesto en el **Capítulo 3**, queremos resaltar que la única modificación que hemos realizado en este método, para abordar sistemas de ecuaciones más generales, es la eliminación del requerimiento (3.6) ya que, como se puede ver, (3.6) es una condición relacionada directamente con el Problema de Complementariedad Horizontal. A este método modificado lo bautizaremos método $CNIDP^+$. Claramente, la eliminación de (3.6) trae consigo por lo menos una consecuencia y es que, a no ser que se establezca una condición similar a (3.6), teóricamente, solo será posible demostrar que el algoritmo $CNIDP^+$ genera sucesiones que convergen a soluciones que se encuentran en el interior de Ω .

En el estudio que hemos realizado, no hemos establecido condiciones adicionales o similares a (3.6), por lo tanto, nos enfocaremos en problemas cuya solución sea estrictamente positiva. En este sentido, dejamos la puerta abierta a futuras investigaciones en el tema que puedan conllevar a establecer condiciones que garanticen un buen desempeño del algoritmo incluso cuando las soluciones de los problemas se encuentren en la frontera de Ω .

A continuación, presentamos los resultados de convergencia del algoritmo $CNIDP^+$.

5.1. Teoría de convergencia

Con respecto a los resultados presentados en esta sección, vale la pena decir que, salvo un teorema, los demás (incluidas sus demostraciones) son similares a los presentadas en el **Capítulo 3**. De igual manera, debemos decir que las hipótesis bajo las cuales se desarrolla esta teoría de convergencia son las mismas hipótesis expuestas en el **Capítulo 3**, por tal razón, solo escribiremos los enunciados de los teoremas que constituyen la teoría de convergencia del algoritmo y hacemos la demostración solo del teorema que difiere de aquellos presentados en el **Capítulo 3**.

Teorema 5.1. *Si $\{z_k\}$ es una sucesión generada por el algoritmo $CNIDP^+$ entonces*

$$\lim_{k \rightarrow \infty} G(z_k) = 0. \quad (5.3)$$

Teorema 5.2. *Asuma la hipótesis **H1**. Si $\{z_k\}$ es una sucesión generada por el algoritmo $CNIDP^+$ y z^* es un punto de acumulación de la sucesión tal que $G'(z^*)$ es no singular entonces $\|G(z^*)\| = 0$ y además,*

$$\lim_{k \rightarrow \infty} z_k = z^*. \quad (5.4)$$

Lema 5.1. *Bajo la hipótesis **H1**. Si p_k es la dirección calculada en **P.2** o **P.4** entonces existe una constante κ tal que*

$$\|p_k\| \leq \kappa \|G(z_k)\|.$$

El siguiente teorema tiene una leve modificación con respecto a su homólogo, el **Teorema 3.3**. Sin embargo, su demostración sigue las mismas ideas de la prueba realizada en el **Capítulo 3**.

Teorema 5.3. *Sea $z \in \Omega$ tal que $G(z) \neq 0$. Si existe $\bar{d} \neq 0$ tal que*

$$\|G(z) + B\bar{d}\| < \|G(z)\|, \quad (5.5)$$

entonces, para cada $\theta \in [0, 1)$, existe $d \in \mathbb{R}^n$ tal que

$$\|G(z) + Bd\| \leq \theta \|G(z)\|. \quad (5.6)$$

Nuevamente, gracias a la hipótesis **H1** y en virtud del **Teorema 5.2**, podemos garantizar que para k suficientemente grande, existe $\bar{d} = -B_k^{-1}G(z)$ tal que (5.5) se satisface. En consecuencia, podemos garantizar la existencia de una dirección cuasi Newton inexacta

que satisfaga (5.6) para cualquier $\theta \in (0, 1)$.

Teorema 5.4. *Asuma la hipótesis **H1**. Sea $\{z^k\}$ una sucesión generada por el algoritmo $CNIDP^+$. Si z^* un punto de acumulación de la sucesión tal que $G'(z^*)$ es no singular y $c_{big} \geq 2T$ entonces, para k suficientemente grande, la dirección p^k es definida por el paso **P.2** del algoritmo.*

El siguiente teorema es la principal novedad en este desarrollo teórico y con él podremos demostrar que el tamaño de paso “completo” será eventualmente aceptado siempre y cuando la solución del problema sea estrictamente positiva.

Teorema 5.5. *Asuma la hipótesis **H1** y sea $\{z_k\}$ una sucesión generada por el algoritmo $CNIDP^+$. Si $z^* \in \text{int}(\Omega)$ es un punto de acumulación de dicha sucesión, $G'(z^*)$ es no singular y $c_{big} \geq 2T$ entonces*

$$\lim_{k \rightarrow \infty} \alpha_k^{break} = 1.$$

Demostración: Como $z^* \in \text{int}(\Omega)$ entonces existe $\delta > 0$ tal que $B(z^*, \delta) \subset \text{int}(\Omega)$. Ahora, por el **Teorema 5.2**, sabemos que $z_k \rightarrow z^*$, luego, existe $\bar{k} > 0$ tal que $z_k \in B(z^*, \frac{\delta}{2})$ para todo $k > \bar{k}$. Por otro lado, por el **Teorema 5.1** y **Lema 5.1**, sabemos que $\|d_k^{CNI}\| \rightarrow 0$ luego, existe $\tilde{k}$ tal que $\|d_k^{CNI}\| < \frac{\delta}{2}$ siempre que $k > \tilde{k}$. Así, si $k > \max\{\bar{k}, \tilde{k}\}$ entonces

$$\|z_k + d_k^{CNI} - z^*\| \leq \|z_k - z^*\| + \|d_k^{CNI}\| < \frac{\delta}{2} + \frac{\delta}{2} = \delta;$$

es decir, $z_k + d_k^{CNI} \in \text{int}(\Omega)$ y por tanto, $\alpha_k^{break} = 1$ para todo k suficientemente grande. ■

Corolario 5.1. *Sea $\{z_k\}$ una sucesión generada por el algoritmo $CNIDP^+$. Si $z^* \in \text{int}(\Omega)$ es un punto de acumulación de dicha sucesión, $G'(z^*)$ es no singular, $c_{big} \geq 2T$ y $\tau_k \rightarrow 1$, $k \rightarrow \infty$ entonces*

$$\lim_{k \rightarrow \infty} \alpha_k^{max} = 1.$$

Teorema 5.6. *Asuma las hipótesis **H1** y **H2** y sea $\{z_k\}$ una sucesión generada por el algoritmo $CNIDP^+$. Si $z^* \in \text{int}(\Omega)$ es un punto de acumulación de dicha sucesión, $G'(z^*)$ es no singular, $c_{big} \geq 2T$, $\tau_k \rightarrow 1$, $\theta_k \rightarrow 0$ y además,*

$$\lim_{k \rightarrow \infty} \frac{\|(G'(z_k) - B_k)p_k\|}{\|p_k\|} = 0 \quad (5.7)$$

entonces

$$\alpha_k = \alpha_k^{max}$$

para k suficientemente grande y además, $z_k \rightarrow z^*$ superlinealmente.

Como pudimos ver, el hilo teórico seguido anteriormente es similar al desarrollo presentado en el **Capítulo 3**, por lo que las eventuales conclusiones que podamos hacer acerca de este algoritmo serán similares a las hechas para el algoritmo CNIDP. Para verificar el potencial del algoritmo $CNIDP^+$ para resolver sistemas de ecuaciones no lineales con restricciones de no negatividad, en la siguiente sección, presentamos los resultados numéricos al resolver dos problemas de aplicación. Uno en el campo de la química y otro en el campo de la física nuclear.

5.2. Experimentos numéricos

En esta sección, reportamos los resultados obtenidos tras realizar experimentos numéricos con dos problemas que permiten resaltar las propiedades del algoritmo $CNIDP^+$. Al igual que los experimentos presentados en el **Capítulo 4**, los códigos de los algoritmos fueron escritos en MATLAB[®] y realizamos las pruebas numéricas en un computador con procesador Intel(R) Core(TM)2 Duo de 3 GHz en cada núcleo y una RAM de 4 GB.

Los valores de los parámetros requeridos para la ejecución del programa fueron exactamente los mismos usados para llevar a cabo los experimentos del **Capítulo 4**. De igual manera, el método de actualización de las matrices B_k fue el de *Broyden* malo.

A continuación, reportaremos los resultados obtenidos al resolver dos problemas prácticos que surgen en la química y la física. De igual forma, comparamos los resultados obtenidos con el algoritmo $CNIDP^+$ con los resultados obtenidos al resolver los problemas con dos algoritmos más. El primero de ellos, es una versión similar al $CNIDP^+$, sin embargo, en este la dirección es obtenida de forma exacta en **P.2**, dicho algoritmo lo denominaremos CNDP. El segundo de los algoritmos con los cuales compararemos el desempeño de la propuesta hecha en esta tesis es el algoritmo propuesto en [51], en el cual, en cada iteración, los autores proponen proyectar la dirección cuasi Newton obtenida al resolver, de forma exacta, el sistema de ecuaciones

$$B_k d^{CNI} = -G(z_k),$$

este algoritmo lo denotaremos DP.

En las tablas en las que reportamos los resultados, usamos las siguientes convenciones:

Método: denota el algoritmo usado para resolver los problemas; estos serán $CNIDP^+$, CNDP o DP.

k : es el número de iteraciones que realizaron los algoritmos para resolver los problemas.

It. In.: identifica el número de iteraciones internas llevadas a cabo por el método cgs en el algoritmo $CNIDP^+$.

Proy.: indica el número de iteraciones en las que los algoritmos CNPD y $CNIDP^+$ hicieron uso de sus respectivas direcciones proyectadas como direcciones de descenso.

z_0 : es el punto inicial con el cual se probó cada algoritmo.

-- : indica que el método no convergió.

n : denota el tamaño de cada problema.

$$RelRes = \frac{\|z^{k+1} - z^*\|}{\|z^k - z^*\|}.$$

t : denota el tiempo *cpu* de cada uno de los algoritmos cuando estos resolvieron los problemas propuestos.

En lo que sigue, presentamos una breve descripción de cada uno de los problemas y los resultados obtenidos para cada uno de ellos.

5.2.1. Equilibrio Químico

El primero de los problemas con los que realizamos las pruebas nos traslada al campo de la química. Este problema, conocido como problema de equilibrio químico [42][52], gira en torno a la combustión de propano (C_3H_8) en el aire para formar 10 productos. A continuación, presentamos el sistema de ecuaciones resultante, tras realizar las manipulaciones estequiométricas necesarias para cada producto. Las incógnitas, z_i , $i = 1, \dots, 10$, representan el número de moles del producto i formado por mol de propano consumido.

$$\begin{aligned}
z_1 + z_4 - 3 &= 0 \\
2z_1 + z_2 + z_4 + z_7 + z_8 + z_9 + 2z_{10} - R &= 0 \\
2z_2 + 2z_5 + z_6 + z_7 - 8 &= 0 \\
2z_3 + z_9 - 4R &= 0 \\
K_1 z_2 z_4 - z_1 z_5 &= 0 \\
K_2 \sqrt{z_2 z_4} - z_6 \sqrt{\frac{z_1 p}{z_T}} &= 0 \\
K_3 \sqrt{z_1 z_2} - z_7 \sqrt{\frac{z_4 p}{z_T}} &= 0 \\
K_4 z_1 - \frac{z_4 z_8 p}{z_T} &= 0 \\
K_5 z_1 \sqrt{z_3} - z_4 z_9 \sqrt{\frac{p}{z_T}} &= 0 \\
K_6 z_1^2 - \frac{z_4^2 z_{10} p}{z_T} &= 0 \\
z_i &\geq 0, \quad i = 1, 2, \dots, n.
\end{aligned} \tag{5.8}$$

En este caso, R es un parámetro que expresa las cantidades relativas de combustible y aire, p es la presión en atmósferas y K_i , $i = 1, \dots, 5$ son las constantes de equilibrio. Cabe resaltar que las constantes de equilibrio son físicas y son obtenidas por medición [52]. En general, los valores dados en la Tabla 5.1 son recomendados para propósitos de prueba de algoritmos.

Constante	Valor
K_1	1.930×10^{-1}
K_2	2.597×10^{-3}
K_3	3.448×10^{-3}
K_4	1.799×10^{-5}
K_5	2.155×10^{-4}
K_6	3.846×10^{-5}

Tabla 5.1: Constantes de equilibrio.

Se debe tener en cuenta que las soluciones físicas de este problemas son tales que $z_i \geq 0$ para todo i , aunque existen soluciones del sistema (5.8) que no satisfacen esta restricción.

En la **Tabla 5.2**, mostramos los resultados obtenidos al resolver el problema de equilibrio químico con los algoritmos $CNIDP^+$, CNDP y DP. Cabe destacar que con el fin de evaluar la globalidad del algoritmo, hemos realizado varios experimentos variando el punto inicial con el que fueron ejecutados.

<i>Método</i>	z_0	k	<i>It. In.</i>	<i>Proy.</i>	t (s)
$CNIDP^+$	$(1, 1, \dots, 1)^T$	57	147	11	0.0820
CNDP	$(1, 1, \dots, 1)^T$	49	*	10	0.0780
DP	$(1, 1, \dots, 1)^T$	—	*	—	—
$CNIDP^+$	$5 \times (1, 1, \dots, 1)^T$	67	132	9	0.0564
CNDP	$5 \times (1, 1, \dots, 1)^T$	43	*	9	0.0624
DP	$5 \times (1, 1, \dots, 1)^T$	—	*	—	—
$CNIDP^+$	$10 \times (1, 1, \dots, 1)^T$	43	99	10	0.0732
CNDP	$10 \times (1, 1, \dots, 1)^T$	—	*	—	—
DP	$10 \times (1, 1, \dots, 1)^T$	220	*	220	0.3520
$CNIDP^+$	$50 \times (1, 1, \dots, 1)^T$	220	857	21	0.4124
CNDP	$50 \times (1, 1, \dots, 1)^T$	263	*	62	0.4212
DP	$50 \times (1, 1, \dots, 1)^T$	461	*	461	0.6552
$CNIDP^+$	$0.1 \times (1, 1, \dots, 1)^T$	53	151	11	0.0870
CNDP	$0.1 \times (1, 1, \dots, 1)^T$	50	*	11	0.0936
DP	$0.1 \times (1, 1, \dots, 1)^T$	—	*	—	—

Tabla 5.2: Algoritmos $CNIDP^+$, CNDP y DP para el problema de equilibrio químico.

Los resultados presentados en la **Tabla 5.2** nos permiten confirmar la globalidad del método $CNIDP^+$ ya que convergió en todos los ensayos a la solución del problema, a diferencia de los métodos CNDP y DP, los cuales divergieron para algunos puntos iniciales. De igual forma, estos experimentos muestran un algoritmo $CNIDP^+$ competitivo frente a sus dos homólogos ya que, a pesar de converger en un mayor número de iteraciones (en casi todos los experimentos), este requirió menos tiempo para hallar la solución del problema, aspecto que es característico de los métodos inexactos.

Una situación particular se presentó cuando el punto inicial fue el vector $50 \times (1, 1, \dots, 1)^T$ ya que en este caso, los tres métodos convergieron pero el algoritmo $CNIDP^+$ lo hizo en menos iteraciones que los otros dos algoritmos. Este comportamiento atípico lo atribuimos, posiblemente, al hecho que el método $CNIDP^+$ proyectó, considerablemente, menos direcciones que los métodos CNDP y DP, razón por la cual suponemos que estos métodos necesitaron de más iteraciones para converger a la solución del problema.

Finalmente, queremos decir que presentamos los resultados obtenidos al resolver el problema de equilibrio químico porque a pesar de ser un problema relativamente pequeño (de orden 10), el método inexacto se mostró a la altura de los métodos exactos los cuales, teórica y experimentalmente, han mostrado un mejor desempeño en estos casos.

En la **Tabla 5.3**, mostramos el comportamiento detallado del método $CNIDP^+$ cuando el punto inicial fue el vector $10 \times (1, 1, \dots, 1)^T$

k	α_k^{break}	$\alpha_k^{m\acute{a}x}$	α_k	θ	$RelRes$
31	0.5471	0.5468	0.5468	0.0323	0.7343
32	*	*	0.9995	0.0313	0.7841
33	*	*	0.9995	0.0303	0.7211
34	0.7643	0.7639	0.7639	0.0294	0.4325
35	0.9327	0.9322	0.9322	0.0286	0.3126
36	1	0.9995	0.9995	0.0278	0.2507
37	1	0.9995	0.9995	0.0270	0.1831
38	1	0.9995	0.9995	0.0263	0.1245
39	1	0.9995	0.9995	0.0256	0.0842
40	1	0.9995	0.9995	0.0250	0.0231
41	1	0.9995	0.9995	0.0244	0.0076
42	1	0.9995	0.9995	0.0238	0.0021
43	1	0.9995	0.9995	0.0232	0.0002

Tabla 5.3: Algoritmo $CNIDP^+$ con $\mathbf{z}_0 = 10 \times (1, 1, \dots, 1)^T$.

Como podemos ver, para k suficientemente grande, α_k^{break} y $\alpha_k = \alpha_k^{max}$ convergen a 1, como lo mostramos en los **Teoremas 5.5** y **5.6**. Más aun, $RelRes \rightarrow 0$, lo cual refleja una convergencia superlineal del método. El símbolo (*) indica que en la k -ésima iteración el método usó una dirección proyectada. Observe que para k suficientemente grande, la dirección cuasi Newton inexacta siempre fue aceptada.

5.2.2. Ecuación H de *Chandrasekhar*

En el campo de la radiación atmosférica, la función H de *Chandrasekhar* aparece como solución (de la ecuación denominada con su mismo nombre) a problemas relacionados con la dispersión y la transferencia radioactiva [12]. Si $c \in (0, 1)$ entonces la ecuación H

adopta la siguiente forma

$$H(\mu) - \left(1 - \frac{c}{2} \int_0^1 \frac{\mu H(\nu)}{\mu + \nu} d\nu\right)^{-1} = 0. \quad (5.9)$$

Si discretizamos la anterior ecuación, aproximando la integral a través de la regla del punto medio; es decir, haciendo

$$\int_0^1 g(\mu) d\mu \approx \frac{1}{n} \sum_{j=1}^n g(\mu_j)$$

donde $\mu_i = (i - 1/2)/n$, para $i = 1, 2, \dots, n$, entonces el problema discreto que tendríamos que resolver sería el sistema $G(\mathbf{z}) = 0$ de n ecuaciones, con el mismo número de variables, $z_i = H(\mu_i)$, donde

$$G_i(\mathbf{z}) = z_i - \left(1 - \frac{c}{2N} \sum_{j=1}^N \frac{\mu_i z_j}{\mu_i + \mu_j}\right)^{-1}.$$

Se sabe que tanto la ecuación (5.9) como el problema discreto resultante, tienen dos soluciones para cada $c \in (0, 1)$ [43]. Sin embargo, solo una de ellas, la positiva, tiene un significado físico de trascendencia.

En la **Tabla 5.4**, podemos ver los resultados obtenidos al intentar hallar la solución positiva de la ecuación H de Chandrasekhar con los algoritmos $CNIDP^+$, CNDP y DP. Cabe mencionar que los experimentos fueron llevados a cabo variando el número de nodos en la discretización de la ecuación (5.9), lo cual evidentemente varió el tamaño del respectivo sistema de ecuaciones a resolver. De igual manera, resaltamos que todos los experimentos fueron llevados a cabo con $c = 0.9$. Los resultados presentados en la **Tabla 5.4** fueron llevados a cabo tomando como punto inicial el vector $\mathbf{z}_0 = (1, 1, \dots, 1)^T$.

<i>Método</i>	<i>n</i>	<i>k</i>	<i>It. In.</i>	<i>Proy.</i>	<i>t (s)</i>
<i>CNIDP</i> ⁺	100	7	13	0	0.1404
CNDP	100	6	*	0	0.1502
DP	100	6	*	6	0.1560
<i>CNIDP</i> ⁺	1000	7	10	0	5.3600
CNDP	1000	6	*	0	11.6485
DP	1000	6	*	6	11.7937
<i>CNIDP</i> ⁺	3000	8	15	0	53.0559
CNDP	3000	7	*	0	285.2322
DP	3000	7	*	7	279.1482
<i>CNIDP</i> ⁺	5000	8	15	0	147.1089
CNDP	5000	7	*	0	1206.5432
DP	5000	7	*	7	1184.6754

Tabla 5.4: Algoritmos *CNIDP*⁺, CNDP y DP para la ecuación de *H*.

Este experimento permite resaltar la ventaja de usar procedimientos inexactos para hallar la dirección $\mathbf{d}_k^{CNI}$. Como podemos observar, cgs requirió aproximadamente de dos iteraciones internas por cada iteración externa para resolver el sistema (3.5), efecto que se ve reflejado en el tiempo de ejecución de cada algoritmo. Es claro que el algoritmo *CNIDP*⁺ muestra un mejor desempeño, en términos de tiempo, a partir de 1000 nodos y esta ventaja se incrementa a medida que aumenta el tamaño del sistema a resolver; es decir, a medida que aumenta el número de nodos en la discretización.

En la **Tabla 5.5**, podemos ver el desempeño de los métodos *CNIDP*⁺, CNDP y DP cuando éstos fueron ejecutados variando el punto inicial y dejando el número de nodos fijo en $n = 1000$.

<i>Método</i>	$\mathbf{z}_0$	k	<i>It.</i>	<i>In.</i>	<i>Proy.</i>	t (s)
$CNIDP^+$	$0.1 \times (1, 1, \dots, 1)^T$	8	12	0	0	5.1984
CNDP	$0.1 \times (1, 1, \dots, 1)^T$	7	*	0	0	15.8655
DP	$0.1 \times (1, 1, \dots, 1)^T$	7	*	7	7	14.6017
$CNIDP^+$	$10 \times (1, 1, \dots, 1)^T$	25	47	0	0	20.5175
CNDP	$10 \times (1, 1, \dots, 1)^T$	—	*	—	—	—
DP	$10 \times (1, 1, \dots, 1)^T$	16	*	16	16	41.7927
$CNIDP^+$	$(1, \dots, 1, 0, \dots, 0)^T$	16	28	0	0	9.8125
CNDP	$(1, \dots, 1, 0, \dots, 0)^T$	9	*	0	0	19.1569
DP	$(1, \dots, 1, 0, \dots, 0)^T$	9	*	9	9	15.5421
$CNIDP^+$	$(1, 0, \dots, 0)^T$	10	15	0	0	5.9124
CNDP	$(1, 0, \dots, 0)^T$	9	*	0	0	19.2037
DP	$(1, 0, \dots, 0)^T$	9	*	9	9	10.3157

Tabla 5.5: Algoritmos $CNIDP^+$, CNDP y DP para la ecuación H . $n = 1000$

Como se puede ver, este experimento nuevamente exalta la robustez del algoritmo $CNIDP^+$, pues convergió en todos los ensayos, para distintos puntos iniciales y con un buen desempeño en términos de tiempo con respecto a los otros dos algoritmos.

Para finalizar los experimentos referentes al problema de *Chandrasekhar*, en la **Tabla 5.6**, reportamos el comportamiento detallado del algoritmo $CNIDP^+$ cuando este resolvió la respectiva ecuación con $n = 1000$ y tomando como punto inicial el vector $\mathbf{z}_0 = (1, \dots, 1)$.

k	α_k^{break}	$\alpha_k^{m\acute{a}x}$	α_k	θ	$RelRes$
1	1	0.9995	0.9995	0.5000	0.5256
2	1	0.9995	0.9995	0.3333	0.4334
3	1	0.9995	0.9995	0.2500	0.3123
4	1	0.9995	0.9995	0.2000	0.1044
5	1	0.9995	0.9995	0.1667	0.0855
6	1	0.9995	0.9995	0.1429	0.0093
7	1	0.9995	0.9995	0.1250	0.0002

Tabla 5.6: Algoritmo $CNIDP^+$ con $n = 1000$ y $\mathbf{z}_0 = (1, 1, \dots, 1)^T$.

Los resultados presentados en la **Tabla 5.6** nos permiten ver un buen despeño del algoritmo $CNIDP^+$, pues los tamaños de paso máximo $\alpha_k^{break} = 1$ y $\alpha_k = 0.9995$ fueron aceptados en todas las iteraciones. Cabe resaltar que este comportamiento fue mucho mejor que el garantizado por los **Teoremas 5.5** y **5.6**. De igual manera, podemos observar una convergencia superlineal de la sucesión que genera el algoritmo a la solución del problema, lo cual se ve reflejado en el hecho que $RelRes \rightarrow 0$ a medida que $\theta \rightarrow 0$.

Finalizamos la presentación de este documento realizando algunos comentarios sobre el trabajo realizado, los logros obtenidos y los posibles trabajos de investigación que pueden surgir a partir de este.

CAPÍTULO 6

COMENTARIOS FINALES

Resolver el Problema de Complementariedad Horizontal a través del problema de minimización (3.4) es importante en el sentido que da robustés a los algoritmos cuyo objetivo es resolver el sistema de ecuaciones no lineales (3.3), los cuales por sí solos ya son bastante significativos en la solución de problemas de este tipo.

En este trabajo, propusimos un algoritmo *cuasi Newton inexacto global* (CNIDP) para resolver Problemas de Complementariedad Horizontal y en particular, para resolver Problemas de Complementariedad no Lineal. Este método puede ser útil, en primer lugar, cuando la matriz jacobiana de la función que define el problema es costosa de calcular y en segundo lugar, cuando el usuario no tiene información de la posible solución del problema. El algoritmo que propusimos es un algoritmo de punto interior y se basa en el uso, siempre que sea posible, de direcciones cuasi Newton inexactas para generar una nueva iteración. En el caso en que una dirección cuasi Newton inexacta no sea aceptada o genere una nueva iteración por fuera de la región de factibilidad, dicha dirección es proyectada sobre el octante positivo de $\mathbb{R}^{n+m+n}$.

Cabe resaltar que el método CNIDP es un método libre de derivadas en el sentido en que no es necesario conocer la matriz jacobiana de la función que define el PCH para realizar la búsqueda lineal. De igual manera, se demostró que el método propuesto goza de buenas

propiedades de convergencia y que, bajo algunas hipótesis razonables, este puede tener una convergencia hasta superlineal. Los experimentos llevados a cabo y aquí reportados dejaron ver un buen desempeño del algoritmo CNIDP para resolver problemas de gran tamaño.

También, propusimos una leve modificación al método CNIDP para resolver sistemas de ecuaciones no lineales con restricciones de no negatividad más generales. Ante tal modificación, realizamos el respectivo desarrollo teórico y probamos, bajo el supuesto de buscar soluciones estrictamente positivas, que el método goza de buenas propiedades de convergencia. Los experimentos numéricos llevados a cabo en problemas emergentes en la química y la física permitieron mostrar la potencialidad de este algoritmo cuando no se conoce la solución del problema y cuando el tamaño del problema a resolver es considerablemente grande.

Consideramos que este trabajo abre la puerta a otras investigaciones relacionadas con el uso de métodos tipo Newton. Por ejemplo, es de gran interés ahondar en la investigación sobre posibles actualizaciones de las matrices jacobianas que eventualmente podrían mejorar el rendimiento del algoritmo. De igual manera, creemos que pensar en posibles condiciones sobre la dirección $\mathbf{d}_k^{CNI}$ similares a la condición (3.6) darían un gran espaldarazo al método $CNIDP^+$ para resolver sistemas de ecuaciones con restricciones de no negatividad. En este mismo sentido, consideramos viable generalizar el estudio del método $CNIDP^+$ a problemas con restricciones sobre conjuntos por lo menos convexos, lo cual, evidentemente, conllevaría a la propuesta de un método para resolver el Problema de Complementariedad Generalizado presentado en la **Sección 2.1.6**.

Finalmente, destacamos el hecho de que como producto de este trabajo de investigación se escribieron tres artículos titulados: “*Global inexact quasi-Newton method for nonlinear system of equations with constraints*”, “*A new global inexact quasi-Newton method for nonlinear complementarity problems*” y “*Fast convergence of an inexact interior point method for horizontal complementarity problems*” [7]. Con respecto a los dos primeros, vale la pena decir que se encuentran en proceso de evaluación y con respecto al tercer artículo, es importante mencionar que ya se encuentra publicado en la revista *Numerical Algorithms*. De igual manera, queremos mencionar que algunos de los resultados de esta investigación fueron expuestos en dos eventos internacionales: *Latin American Workshop on Optimization and Control* (LAWOC-2016) llevado a cabo en la ciudad de Tandil (Argentina) e *International Conference on Applied Mathematics and Informatics* (ICAMI-2017) llevado a cabo en la ciudad de San Andrés (Colombia).

CAPÍTULO 7

APÉNDICE

En este capítulo analizaremos, con un poco más de detalle, los problemas de complementariedad de la familia mencionada en la **Sección 2.1**.

Iniciaremos este análisis con el problema de complementariedad que quizá dio pie a la comunidad matemática para investigar en temáticas concernientes al campo de la complementariedad, dicho problema es el problema de complementariedad lineal.

Si bien este problema era ya conocido por la comunidad a principio de los años 60, no fue sino hasta 1963 que llamó la atención de los matemáticos, en particular, de *Howson* y *Lemke* [45] quienes demostraron que el problema de hallar el punto de equilibrio de *Nash* era equivalente a resolver un PCL. Sin embargo, solo hasta 1968 este problema se logró consolidar como un problema de gran interés en las matemáticas pues fue en ese año que *Cottle* y *Dantzing* en [18] mostraron que los problemas de programación lineal y cuadrática y el problema de los juegos *bimatrix* se podían expresar como un PCL. Es más, existen muchos autores que atribuyen la propuesta del PCL a Cottle y Dantzing y ubican, en la historia, la aparición de este problema en el año 1968.

Ahora bien, recordemos que si $M \in \mathbb{R}^{n \times n}$ y $\mathbf{q} \in \mathbb{R}^n$ son, respectivamente, una matriz

y un vector cualquiera, el PCL($M, \mathbf{q}$) consiste en hallar $\mathbf{x} \in \mathbb{R}^n$ tal que

$$M\mathbf{x} + \mathbf{q} \geq 0, \quad \mathbf{x} \geq 0 \quad \text{y} \quad \mathbf{x}^T(M\mathbf{x} + \mathbf{q}) = 0. \quad (7.1)$$

Con respecto a este problema se han escrito y publicado centenares de artículos y varios libros en los que, en algunos, se desarrollan métodos para hallar soluciones al problema y, en otros, se establecen condiciones bajo las cuales se puede caracterizar el conjunto solución del problema [4, 6, 18, 19, 21, 28, 40, 54], entre otros.

Por otro lado observemos que, si suponemos que $F(\mathbf{x}) = M\mathbf{x} + \mathbf{q}$, entonces, (7.1) se podría escribir como

$$F(\mathbf{x}) \geq 0, \quad \mathbf{x} \geq 0 \quad \text{y} \quad \mathbf{x}^T F(\mathbf{x}) = 0.$$

Claramente, la anterior expresión es más general e hizo pensar a la comunidad matemática en el caso en el que F no fuera una función lineal. Así pues, en 1964, Cottle, en su tesis doctoral, introdujo por primera vez el problema de complementariedad no lineal.

En los años posteriores se fue desarrollando la teoría relacionada con las condiciones de complementariedad y se fueron estableciendo los problemas de la “familia de problemas de complementariedad” presentada en la **Sección 2.1**. Como es de esperarse, existen casos en los que, bajo algunas condiciones, uno de los problemas de la familia es equivalente a otro de los problemas de la misma. En las líneas escritas en el **Capítulo 2**, tratamos de establecer las relaciones entre cada uno de los problemas y el PCNL, en particular, queremos destacar el hecho de que, bajo algunas consideraciones, cada problema de la familia se puede reducir a un PCNL. En efecto,

1. Claramente el PCL es un caso particular del PCNL, por lo tanto, cualquier estrategia propuesta para resolver problemas de complementariedad no lineal es aplicable a problemas de complementariedad lineal.
2. Con respecto al problema de complementariedad mixto presentado en la **Sección 2.1.2**, es fácil ver que si $l_i = 0$ y $u_i = \infty$ para $i = 1, 2, \dots, n$ entonces, la desigualdad (2.3) se reduce a $\mathbf{x} \geq 0$, la igualdad (2.4) se transforma en $\mathbf{x}^T \mathbf{w} = 0$ y la igualdad (2.5) implica que $\mathbf{v} = 0$, por lo tanto, el PCm(F) se transforma en el problema de hallar $\mathbf{x}, \mathbf{w} \in \mathbb{R}^n$ tal que

$$\begin{aligned} F(\mathbf{x}) - \mathbf{w} &= 0 \\ \mathbf{x}^T \mathbf{w} &= 0 \\ \mathbf{x}, \mathbf{w} &\geq 0 \end{aligned}$$

el cual, evidentemente, es un PCNL(F).

3. Con respecto al problema de complementariedad horizontal, en el **Capítulo 3** y en particular en (3.2), vimos bajo qué condiciones un PCH equivalente a un PCNL.
4. Claramente, si en un problema de complementariedad vertical intervienen dos funciones en su formulación y una de ellas es la función identidad entonces el problema se reducirá a un PCNL, tal como se vio en la **Sección 2.1.4**
5. Finalmente, con respecto al problema de desigualdades variacionales, aprovecharemos estas líneas para demostrar la equivalencia, en un caso particular, entre éste y un PCNL. Recordemos que dados un Conjunto no vacío y convexo X y una función continuamente diferenciable F , el problema de desigualdades variacionales consiste en hallar $\mathbf{x} \in X$ tal que

$$F(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) \geq 0, \quad \text{para todo } \mathbf{y} \in X. \quad (7.2)$$

El caso particular donde se obtiene la equivalencia entre un problema de desigualdades variacionales y un PCNL sucede cuando $X = \mathbb{R}_+^n$, donde $\mathbb{R}_+^n$ es el octante no negativo de $\mathbb{R}^n$.

En efecto, en primer lugar, observe que si $\mathbf{x}^*$ es una solución del PCNL(F) y $\mathbf{y} \in \mathbb{R}_+^n$ entonces,

$$F(\mathbf{x}^*)^T(\mathbf{y} - \mathbf{x}^*) = F(\mathbf{x}^*)^T \mathbf{y} - F(\mathbf{x}^*)^T \mathbf{x}^* = F(\mathbf{x}^*)^T \mathbf{y}$$

pero como $F(\mathbf{x}^*) \geq 0$ (por ser $\mathbf{x}^*$ solución del PCNL(F)) y $\mathbf{y} \geq 0$ podemos inferir que $F(\mathbf{x}^*)^T \mathbf{y} \geq 0$ en consecuencia,

$$F(\mathbf{x}^*)^T(\mathbf{y} - \mathbf{x}^*) \geq 0$$

y como $\mathbf{y} \in \mathbb{R}_+^n$ es cualquier vector, concluimos que $\mathbf{x}^*$ es una solución del $DV(\mathbb{R}_+^n, F)$.

Recíprocamente, si $\mathbf{x}^*$ es una solución del $DV(\mathbb{R}_+^n, F)$ entonces, $\mathbf{x}^* \geq 0$ y (7.2) se satisface para cualquier vector $\mathbf{y} \in \mathbb{R}_+^n$. En particular, si $\mathbf{y} = 0$ y si $\mathbf{y} = 2\mathbf{x}$ obtendríamos respectivamente que: $F(\mathbf{x}^*)^T \mathbf{x}^* \leq 0$ y que $F(\mathbf{x}^*)^T \mathbf{x}^* \geq 0$ por lo tanto, concluimos que

$$F(\mathbf{x}^*)^T \mathbf{x}^* = 0.$$

Análogamente, si suponemos que $\mathbf{y} = \mathbf{x}^* + \mathbf{e}_i$, donde $\mathbf{e}_i$ es el i -ésimo vector canónico de $\mathbb{R}^n$ entonces, de (7.2), podemos concluir que $F_i(\mathbf{x}^*) \geq 0$ para cada $i = 1, 2, \dots, n$, en consecuencia, $F(\mathbf{x}^*) \geq 0$ y por lo tanto, $\mathbf{x}^*$ es una solución del PCNL(F).

Lo anterior demuestra que el conjunto solución del $DV(\mathbb{R}_+^n, F)$ y el conjunto solución del PCNL(F) son iguales.

Si bien en este apéndice nos hemos centrado en resaltar los casos en los que los diversos problemas de la familia de complementariedad son equivalentes al PCNL, es bien sabido que existen muchos casos en los que tales equivalencias no son posibles y para ello, se deben diseñar métodos que aprovechen la estructura de cada uno de los problemas. En este sentido, en esta investigación nos preocupamos por proponer un método para resolver el problema de complementariedad horizontal el cual, por lo mencionado en el ítem 3, también es capaz de resolver problemas de complementariedad no lineal y, en este mismo sentido, invitamos a aquellas personas interesadas en investigar en estos temas, a proponer nuevas estrategias que permitan resolver los diversos problemas de complementariedad existentes.

BIBLIOGRAFÍA

- [1] R. Andreani, J. J. Júdice, J. M. Martínez, and T. Martini, *Feasibility problems with complementarity constraints*, Eur. J. Oper. Res. **249** (2016), no. 1, 41 – 54.
- [2] R. Andreani, J. J. Júdice, J. M. Martínez, and J. Patricio, *A projected-gradient interior-point algorithm for complementarity problems*, Numer. Algorithms **57** (2011), no. 4, 457 – 485.
- [3] R. Andreani, J. Júdice, M. Martínez, and J. Patricio, *On the natural merit function for solving complementarity problems*, Math. Program. **130** (2011), no. 1, 211 – 223.
- [4] F. E. Arenas, H. J. Martínez, and R. Pérez, *Least change secant update methods for nonlinear complementarity problem*, Ing. Cienc. **11** (2015), no. 21, 11–36.
- [5] C. A. Arias, *Un algoritmo cuasi Newton global para problemas de complementariedad no lineal*, Tesis de Maestría, Universidad del Cauca, 2014.
- [6] C. A. Arias, H. J. Martínez, and R. Pérez, *A global cuasi Newton Algorithm for nonlinear complementarity problems*, Pac. J. Optim. **13** (2017), no. 1, 1 – 15.
- [7] C. A. Arias and J. M. Martínez, *Fast convergence of an inexact interior-point method for horizontal complementarity problems*, Numer. Algorithms <https://doi.org/10.1007/s11075-018-0480-8> (2018).
- [8] S. Billups and K. Murty, *Complementarity problems*, J. Comput. Appl. Math. **124** (2000), no. 1-2, 303 – 318.

- [9] C. G. Broyden, *A Class of Methods for Solving Nonlinear Simultaneous Equations*, Math. Comput. **19** (1965), no. 92, 577 – 593.
- [10] C. G. Broyden, J. E. Dennis, and J. J. Moré, *On the local and superlinear convergence of quasi-newton methods*, IMA J. Appl. Math. **12** (1973), no. 3, 223 – 245.
- [11] S. Buhmiller and N. Krejić, *A new smoothing quasi-newton method for nonlinear complementarity problems*, J. Comput. Appl. Math. **211** (2008), no. 2, 141 – 155.
- [12] S. Chandrasekhar, *Radiative transfer*, 1960.
- [13] X. Chen, *Smoothing methods for complementarity problems and their applications: A survey*, J. Oper. Res. Soc. Jpn. **43** (2000), no. 1, 32 – 47.
- [14] F. H. Clarke, *Necessary Conditions for Nonsmooth Problems in Optimal Control and the Calculus of Variations*, Ph.D. thesis, University of Washington, 1973.
- [15] F. H. Clarke, *Generalized gradients and applications*, Trans. Am. Math. Soc **205** (1975), 247 – 262.
- [16] Frank H. Clarke, *Optimization and nonsmooth analysis*, Society for Industrial and Applied Mathematics, 1990.
- [17] A. Costa, J. Martins, I. Figueiredo, and J. Júdice, *The directional instability problem in systems with frictional contacts.*, Comput. Methods Appl. Mech. Eng. **193** (2004), no. 1, 357 – 384.
- [18] R. Cottle and G. Dantzing, *Complementary pivot theory of mathematical programming*, Linear Alg. Appl. **1** (1968), no. 1, 103 – 125.
- [19] R. Cottle, J. Pang, and R. Stone, *The linear complementarity problem*, Society for Industrial and Applied Mathematics, 2009.
- [20] T. De luca, F. Facchinei, and Kanzow C., *A theoretical and numerical comparison of some semismooth algorithms for complementarity problems*, Comput. Optim. Appl. **16** (2000), no. 2, 173 – 205.
- [21] T. De luca, F. Facchinei, and C. Kanzow, *A semismooth equation approach to the solution of nonlinear complementarity problems*, Math. Program **75** (1996), no. 3, 407 – 439.
- [22] J. Dennis and R. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Society for Industrial and Applied Mathematics, (1996).

- [23] J. Ding and H. Yin, *A new homotopy method for solving nonlinear complementarity problems*, Numer. Math. Theory. Me. **16** (2007), 155 – 163.
- [24] S. Dirkse and M. Ferris, *Mcplib: a collection of nonlinear mixed complementarity problems.*, Optim. Method. Softw. **5** (1995), no. 4, 319 – 345.
- [25] ———, *The path solver: a nonmonotone stabilization scheme for mixed complementarity problems.*, Optim. Method Softw. **5** (1995), no. 2, 123 – 156.
- [26] S. Eisenstat and H. Walker, *Globally convergent inexact newton methods*, SIAM J. Optim **4** (1994), no. 2, 393 – 422.
- [27] F. Facchinei and C. Kanzow, *A nonsmooth inexact newton method for the solution of large-scale nonlinear complementarity problems.*, Math. Program. **76** (1997), no. 3, 493 – 512.
- [28] M. Ferris and C. Kanzow, *Complementarity and related problems: a survey*, 1995.
- [29] M. C. Ferris and J. S. Pang, *Engineering and economic applications of complementarity problems*, SIAM Rev. **39** (1997), no. 4, 669 – 713.
- [30] A. Fischer, *A special newton-type optimization method*, Optimization. **24** (1992), no. 3-4, 269 – 284.
- [31] A. Fischer and C. Kanzow, *On finite termination of an iterative method for linear complementarity problems*, Math. Program. **74** (1996), no. 3, 279 – 292.
- [32] T. Friesz, R. Tobin, P. Smith, and P. Harker, *A nonlinear complementarity formulation and solution procedure for the general derived demand network equilibrium problem*, J. Regional Sci. **23** (1983), no. 3, 337 – 359.
- [33] M. A. Gomes-Ruggiero, J. M. Martinez, and S. A. Santos, *Solving nonsmooth equations by means of quasi-newton methods with globalization*, Recent Advances in Nonsmooth Optimization, World Scientific Publisher, Singapore (1995), 121 – 140.
- [34] M. Haddou and M. Patrick, *Smoothing methods for nonlinear complementarity problems*, J. Optim. Theory Appl. **160** (2014), no. 3, 711 – 729.
- [35] P. T. Harker and J. S. Pang, *Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications*, Math. Program. **48** (1990), no. 1-3, 161 – 220.
- [36] A. Iusem, J. Júdice, V. Sessa, and P. Sarabando, *On the solution of the symmetric eigenvalue complementarity problem by the spectral projected gradient algorithm.*, Numer. Algorithms **47** (2008), no. 4, 391 – 407.

- [37] J. Júdice, M. Raydan, S. Rosa, and S. Santos, *On the solution of the symmetric eigenvalue complementarity problem by the spectral projected gradient algorithm.*, Numer. Algorithms **47** (2008), no. 4, 391 – 407.
- [38] J. Júdice, H. Serali, and I. Ribeiro, *The eigenvalue complementarity problem.*, Comput. Optim. Appl. **37** (2007), no. 1, 139 – 156.
- [39] J. Júdice, H. Serali, I. Ribeiro, and Faustino A., *Complementarity active-set algorithm for mathematical programming problems with equilibrium constraints.*, J. Optim. Theory Appl. **134** (2007), no. 3, 467 – 481.
- [40] C. Kanzow, *Inexact semismooth newton methods for large-scale complementarity problems.*, Optim. Method. Softw. **19** (2004), no. 3-4, 309 – 325.
- [41] C. Kanzow and H. Kleinmichel, *A new class of semismooth newton-type methods for nonlinear complementarity problems*, Comput. Optim. Appl. **11** (1998), no. 3, 227 – 251.
- [42] C. Kanzow, N. Yamashita, and M. Fukushima, *Levenberg-marquardt methods for constrained nonlinear equations with strong local convergence properties*, J. Comput. Appl. Math. **172** (2004), no. 2, 375 – 397.
- [43] T. Kelly, *Iterative methods for linear and nonlinear equations*, Society for Industrial and Applied Mathematics, 1995.
- [44] M. Kostreva, *Elasto-hydrodynamic lubrication: A non-linear complementarity problem*, Int. J. Numer. Methods Fluids **4** (1984), no. 4, 377 – 397.
- [45] C. E. Lemke and J. T. Howson, *Equilibrium points of bimatrix games*, J. Soc. Indust. Appl. Math **12** (1964), no. 2, 413 – 423.
- [46] Q. Li and D. Li, *A smoothing Newton method for nonlinear complementarity problems.*, Adv. Model. Optim. **13** (2011), no. 2, 141 – 152.
- [47] L. Liao and H. Qi, *A neural network for the linear complementarity problem*, Math. Comput. Model. **29** (1999), 9 – 18.
- [48] L. Liao, H. Qi, and L. Qi, *Solving nonlinear complementarity problems with neural networks*, Comput. Appl. Math. **131** (2001), no. 1, 343 – 359.
- [49] L. Lukšan, *Inexact trust region method for large sparse systems of nonlinear equations*, J. Optim. Theory Appl. **81** (1994), 569–590.
- [50] C. Ma, *A new smoothing quasi-newton method for nonlinear complementarity problems*, Appl. Math. Comput. **171** (2005), no. 2, 807 – 823.

- [51] L. Marini, B. Morini, and M. Porcelli, *Quasi-newton methods for convex constrained nonlinear systems and their application*, Comput. Optim. Appl. <https://doi.org/10.1007/s10589-018-9980-7> (2018).
- [52] K. Meintjes and A. Morgan, *Chemical equilibrium system as numerical test problems*, ACM Trans. Math. Softw. **16** (1990), no. 2, 143 – 151.
- [53] J. Nocedal and S. Wright, *Numerical optimization*, Springer Series in Operations Research, 2000.
- [54] J. Ortega and W. Rheinboldt, *Iterative solutions of nonlinear equations in several variables.*, Academic Press, 1970.
- [55] J. S. Pang and L. Qi, *Nonsmooth equations: Motivation and algorithms*, SIAM J. Optim. **3** (1993), no. 3, 443 – 465.
- [56] L. Qi and D. Sun, *A survey of some nonsmooth equations and smoothing newton methods*, Progress in Optimization: Contributions from Australasia, Springer US, 1999, pp. 121 – 146.
- [57] M. Querioz, J. Júdice, and Jr. Humes, *The symmetric eigenvalue complementarity problem.*, Math. Comput. **73** (2003), no. 1, 1849 – 1863.
- [58] S. P. Riu and C. X. Xu, *A smoothing inexact Newton method for nonlinear complementarity problems*, J. Comput. Appl. Math. **233** (2010), no. 9, 2332 – 2338.
- [59] V. Rocha, J. M. Martínez, and R. Pérez, *On the local convergence of quasi-newton methods for nonlinear complementary problems*, Appl. Numer. Math. **30** (1999), no. 1, 3 – 22.
- [60] T. Steihaug, *Damped inexact quasi newton methods*, Departament of Mathematical Science, Rice University (1984), TR81–3.
- [61] R. Tobin, *A variable dimension solution approach for the general spatial price equilibrium problem*, Math. Program. **40** (1988), no. 1, 33 – 51.
- [62] Z. Wang, H. Li, and S. Huang, *A smoothing inexact Newton method for nonlinear complementarity problems.*, Abstract Appl. Anal. **2015** (2015), 1 – 12.
- [63] J. G. Wardrop, *Road paper. some theoretical aspects of road traffic research*, Proceeding of the Institute of Civil Engineers **1** (1952), no. 3, 325 – 378.
- [64] Q. Xu and C. Dang, *A new homotopy method for solving non-linear complementarity problems*, Optimization **57** (2008), no. 5, 681 – 689.

-
- [65] L. Yong, *Nonlinear complementarity problem and solution methods*, Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence: Part I, Springer-Verlag, 2010, pp. 461 – 469.
- [66] X. Youshen and F. Gang, *A neural network for solving nonlinear projection equations*, Neural Netw. **20** (2007), no. 5, 577 – 589.