Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623



Artículo original

# Resolviendo el problema de valores propios complementarios mediante un algoritmo cuasi-Newton

## Solving the eigenvalue complementarity problem using a quasi-Newton algorithm

**Favián Arenas[1,*], Carlos Arias[1], Rosana Pérez[1]**

[1]Departamento de Matemáticas, Universidad del Cauca, Popayán, Colombia

## Resumen

En este artículo consideramos el problema de valores propios complementarios, de gran interés para muchos investigadores por sus numerosas aplicaciones en Ingeniería y Física, y abordamos su solución como un problema de complementariedad no lineal usando un método cuasi-Newton, un tipo de método que, hasta donde conocemos, no ha sido utilizado para dicho fin. Verificamos que el problema satisface ciertas hipótesis que permiten utilizar un algoritmo global cuasi-Newton y realizamos pruebas numéricas que muestran la eficacia del algoritmo utilizado y lo hacen una buena alternativa de solución para problemas de valores propios complementarios.

**Palabras clave:** valores propios complementarios; problema de complementariedad no lineal; programación no lineal; cuasi-Newton.

## Abstract

In this article we consider the complementarity eigenvalue problem, which is of great interest to many researchers due to its numerous applications in Engineering and Physics. We approach its solution as a nonlinear complementarity problem using a quasi-Newton method, a type of method that, as far as we know, has not been used for this purpose. We verify that the problem satisfies certain hypotheses that allow the use of a global quasi-Newton algorithm and we analyze its numerical performance. Numerical tests show the efficiency of the algorithm used and make it a good alternative to solve complementary eigenvalue problems.

**Keywords:** eigenvalue complementarity; nonlinear complementarity problem; nonlinear programming; quasi-Newton. AMS Subject Classification: 93B60; 90C30; 90C33.

**Citación:** Arenas F, Arias C, Pérez R. Resolviendo el problema de valores propios complementarios mediante un algoritmo cuasi-Newton. Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022. doi: https://doi.org/10.18257/raccefyn.1623

**Editor:** Luis Español González

***Correspondencia:**
Favian Arenas; farenas@unicauca.edu.co

**Recibido:** 25 de enero de 2022
**Aceptado:** 6 de mayo de 2022
**Publicado:** 28 de junio de 2022

Este artículo está bajo una licencia de Creative Commons Reconocimiento-NoComercial-Compartir Igual 4.0 Internacional

## Introducción

Sean $A$ y $B \in \mathbb{R}^{n\times n}$, con $B$ una matriz definida positiva. El problema de Valores Propios Complementarios, abreviadamente $VPC(A,B)$, consiste en encontrar un escalar positivo $\lambda$ y un vector $\boldsymbol{x} \in \mathbb{R}^n$, no nulo, tal que

$$\boldsymbol{x} \geq 0, \qquad (\lambda B - A)\boldsymbol{x} \geq 0, \qquad \boldsymbol{x}^T (\lambda B - A)\boldsymbol{x} = 0. \tag{1}$$

Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623

Este problema tiene numerosas aplicaciones en ingeniería y física tales como, el estudio de la frecuencia de resonancia en estructuras y la estabilidad de sistemas dinámicos (**Pinto da Costa, Martins, Figueiredo, y Júdice**, 2004).

Una solución al problema $VPC(A,B)$ es un par $(\lambda,\boldsymbol{x})$, donde $\lambda$ es llamado un *valor propio complementario*, y $\boldsymbol{x}$ es llamado *vector propio complementario* asociado a $\lambda$.

Como ocurre con los vectores propios convencionales, los vectores propios complementarios asociados a un valor propio complementario también forman un conjunto infinito, el cual es un cono convexo (**Iusem, Júdice, Sessa, y Sarabando**, 2019). Esto puede verse fácilmente con las siguientes matrices,

$$A = \begin{pmatrix} 18 & 0 \\ 0 & 1 \end{pmatrix} \text{ y } \quad B = \begin{pmatrix} 9 & 3 \\ 3 & 5 \end{pmatrix}.$$

En este caso, las soluciones del problema $VPC(A,B)$ son de la forma: $(\lambda_1,\boldsymbol{x}) = (2,(\alpha,0)^T)$ y $(\lambda_2,\boldsymbol{x}) = (\frac{1}{5},(0,\alpha)^T)$, con $\alpha > 0$. En efecto, si $\boldsymbol{x} = (x_1,x_2)^T$ entonces

$$\boldsymbol{x}^T(\lambda_B - A)\boldsymbol{x} = (9\lambda - 18)x_1^2 + 6x_1x_2 + (5\lambda - 1)x_2^2.$$

Teniendo en cuenta que $\boldsymbol{x} \geq 0$, tenemos que

- Si $\lambda = 2$ entonces $\boldsymbol{x}^T(\lambda B - A)\boldsymbol{x} = 0$ si, y solo si $x_2 = 0$ y $x_1 \geq 0$.
- Si $\lambda = 1/5$ entonces $\boldsymbol{x}^T(\lambda B - A)\boldsymbol{x} = 0$ si, y solo si $x_1 = 0$ y $x_2 \geq 0$.

Más aún, para los pares $(\lambda_1,\boldsymbol{x}) = (2,(\alpha,0)^T)$ y $(\lambda_2,\boldsymbol{x}) = (\frac{1}{5},(0,\alpha)^T)$, con $\alpha > 0$, es fácil verificar que $(\lambda B - A)\boldsymbol{x} \geq 0$. Por lo tanto, son soluciones del VPC(A, B).

Observe que, para cada valor propio hay infinitos vectores propios. Como es de esperarse, se prefiere un vector representante, como si fuera un "generador". Con el fin de obtener este vector propio complementario, el problema $VPC(A,B)$, se puede transformar en el siguiente problema equivalente (**Facchinei y Pang**, 2003).

Dado $p > 0$ encontrar un escalar positivo $\lambda$ y un vector $\boldsymbol{x} \in \mathbb{R}^n$ tal que

$$\boldsymbol{x} \geq 0, \qquad (\lambda B - A)\boldsymbol{x} \geq 0, \qquad \boldsymbol{x}^T(\lambda B - A)\boldsymbol{x} = 0, \quad \|\boldsymbol{x}\|_1 = p. \tag{2}$$

Por la nonegatividad del vector $\boldsymbol{x}$, se tiene que $\|\boldsymbol{x}\|_1 = \sum_{k=1}^n x_k$, lo cual se puede expresar como $\boldsymbol{e}^T\boldsymbol{x}$, donde $\boldsymbol{e}^T = (1,1,\ldots,1)$. Así, el problema (2) se transforma en

$$\boldsymbol{x} \geq 0, \qquad (\lambda B - A)\boldsymbol{x} \geq 0, \qquad \boldsymbol{x}^T(\lambda B - A)\boldsymbol{x} = 0, \quad \boldsymbol{e}^T\boldsymbol{x} = p \tag{3}$$

que llamaremos problema $VPC2(A,B)$.

El caso particular, en el cual las matrices $A$ y $B$ son simétricas, generalmente se analiza independientemente, ya que pueden aplicarse métodos especiales para su solución (**Iusem y cols.**, 2019); este tipo de problemas se les llama problema de valores propios complementarios simétrico. Cuando alguna de las matrices $A$ o $B$ no es simétrica, que es el caso más general, el problema es llamado no simétrico.

El problema $VPC(A,B)$ puede reformularse como un problema de minimización con restricciones de complementariedad (**Sherali y Tuncbilek**, 1992), como uno de minimización global (**Júdice, Sherali, y Ribeiro**, 2007) y como un problema de complementariedad no lineal (**Facchinei y Pang**, 2003). Nuestro interés está en esta última reformulación debido a que existen algoritmos competitivos, en particular, tipo cuasi-*Newton* que permiten resolver en forma eficiente problemas de complementariedad no lineal y no han sido utilizados, hasta donde conocemos, para resolver problemas de valores propios complementarios.

En este artículo, consideramos el problema de los valores propios complementarios y su reformulación como un problema de complementariedad no lineal. Verificamos que este

problema satisface ciertas hipótesis que permiten usar un algoritmo global cuasi-*Newton* propuesto en (**Arias, Martínez, y Pérez**, 2017) para resolver problemas de complementariedad no lineal en general. Implementamos dicho algoritmo adaptándolo al caso particular de los valores propios complementarios y realizamos un análisis numérico de su desempeño. Pruebas numéricas preliminares indican que es una buena alternativa para resolver dichos problemas.

El resto de este documento está organizado de la siguiente forma: en la siguiente sección, se reformula el problema de valores propios complementarios como uno de complementariedad no lineal, se mencionan métodos numéricos tradicionales para resolverlo y se introduce el algoritmo cuasi-*Newton* global que se usará para resolver indirectamente el VPC(A,B), luego, se verifica que el problema de valores propios complementarios satisface las hipótesis necesarias para que el algoritmo utilizado converja, posteriormente, se incluyen las pruebas numéricas que se realizaron para analizar la eficacia de esta estrategia y finalmente, se presentan algunos comentarios sobre el trabajo realizado.

## Reformulación del $VPC(A,B)$ como un problema de complementariedad no lineal

En general, dada $F\colon \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$, $F(\boldsymbol{y}) = (F_1(\boldsymbol{y}),\ldots,F_{n+1}(\boldsymbol{y}))^T$, continuamente diferenciable, el Problema de Complementariedad No Lineal consiste en encontrar un vector $\boldsymbol{y} \in \mathbb{R}^{n+1}$ que satisfaga las tres condiciones siguientes,

$$\boldsymbol{y} \geq 0, \;\; F(\boldsymbol{y}) \geq 0, \;\; \boldsymbol{y}^T F(\boldsymbol{y}) = 0. \tag{4}$$

Veamos a continuación que, en efecto, el problema $VPC2(A,B)$ puede reformularse como un problema de complementariedad no lineal.

A partir de (3), $VPC2(A,B)$ puede expresarse de la siguiente forma (**Júdice y cols.**, 2007): hallar $\lambda > 0$ y $\boldsymbol{x} \in \mathbb{R}^n$ tal que

$$\boldsymbol{x} \geq 0, \qquad \left(B - \frac{1}{\lambda}A\right)\boldsymbol{x} \geq 0, \qquad \boldsymbol{x}^T(\lambda B - A)\boldsymbol{x} = 0, \quad \boldsymbol{e}^T\boldsymbol{x} = p.$$

Haciendo un cambio de variable, se definen $F\colon \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ y $\boldsymbol{y} \in \mathbb{R}^{n+1}$ tales que,

$$y_i = \begin{cases} x_i, & \text{si} \quad 1 \leq i \leq n \\ \frac{1}{\lambda}, & \text{si} \quad i = n+1, \end{cases} \qquad F(\boldsymbol{y}) = \begin{cases} [B - y_{n+1}A]_i\boldsymbol{x}, & \text{si } 1 \leq i \leq n \\ \boldsymbol{e}^T\boldsymbol{x} - p, & \text{si } i = n+1, \end{cases} \tag{5}$$

con lo cual, el problema $VPC2(A,B)$, es de la forma (4); es decir, es un problema de complementariedad lineal.

Una estrategia de solución para problemas de complementariedad no lineal, tal vez, la más popular y ampliamente usada, consiste en reformularlo como un sistema de ecuaciones no lineales mediante ciertas funciones especiales, $\varphi\colon \mathbb{R}^2 \to \mathbb{R}$, llamadas *funciones de complementariedad* (**Yong**, 2010) que satisfacen la equivalencia, $\varphi(a,b) = 0 \Longleftrightarrow a \geq 0, \;\; b \geq 0, \;\; ab = 0$.

En este artículo usaremos la familia de funciones de complementariedad introducida en (**Kanzow y Kleinmichel**, 1998), y definida por,

$$\varphi_\tau(a,b) = \sqrt{(a-b)^2 + \tau ab} - a - b, \qquad \tau \in (0,4), \tag{6}$$

la cual ha resultado competitiva en la reformulación de problemas de complementariedad (**Arenas, Martínez, y Pérez**, 2014) (**Pérez, Arenas, Martínez, y Arias**, 2019). Así, la reformulación del problema de valores propios complementarios puede, a su vez, reformularse

Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623

como el siguiente sistema de ecuaciones no lineales no diferenciable

$$\Phi_\tau(\boldsymbol{y}) = \begin{pmatrix} \varphi_\tau(y_1, F_1(\boldsymbol{y})) \\ \vdots \\ \varphi_\tau(x_{n+1}, F_{n+1}(\boldsymbol{y})) \end{pmatrix} = \mathbf{0}. \tag{7}$$

Este sistema se puede resolver mediante un método tipo *Newton* generalizado (**Ferris y Pang**, 1997; **Facchinei y Pang**, 2003; **Kanzow y Kleinmichel**, 1998) o cuasi-*Newton* (**Arenas y cols.**, 2014; **Arias y cols.**, 2017), entre otros (**Arenas, Martínez, y Pérez**, 2020; **Sánchez, Pérez, y Martínez**, 2021). En el primer caso, la iteración básica, es de la forma

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_k - H_k^{-1}\Phi_\tau(\boldsymbol{y}_k),$$

donde $H_k \in \partial\Phi_\tau(\boldsymbol{y}_k)$, es decir $B_k$ es una matriz en el jacobiano generalizado de $\Phi_\tau$ en $\boldsymbol{y}_k$ (**Clarke**, 1975). En el caso, de un método cuasi-*Newton*, la iteración básica, es de la forma

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_k - B_k^{-1}\Phi_\tau(\boldsymbol{y}_k),$$

donde $B_k \approx H_k \in \partial\Phi_\tau(\boldsymbol{y}_k)$.

Por otra parte, con el fin de globalizar este tipo de algoritmos que resuelven sistemas de ecuaciones no lineales como el planteado en (7), se introduce una función de mérito y se resuelve un problema de minimización (**Dennis y Schnabel**, 1996). La función de mérito natural y ampliamente usada es $\Psi_\tau$ definida por $\Psi_\tau(\boldsymbol{y}) = \frac{1}{2}\Phi_\tau(\boldsymbol{y})^T\Phi_\tau(\boldsymbol{y})$, y el problema de minimización es el siguiente:

$$\begin{array}{ll} Minimizar & \Psi_\tau(\boldsymbol{y}). \\ \boldsymbol{y} \in \mathbb{R}^{n+1} & \end{array} \tag{8}$$

En (**Arias y cols.**, 2017), los autores proponen un algoritmo global cuasi-*Newton*, concretamente un algoritmo *secante de cambio mínimo*, para resolver problemas de complementariedad no lineal, como el problema (8), el cual usa búsqueda lineal como estrategia de globalización. Este algoritmo mostró resultados excelentes al ser probado con funciones de difícil comportamiento numérico. Por ello, consideramos que es la mejor opción para resolver el problema $VPC(A,B)$ vía la reformulación como un problema de complementariedad no lineal. En la experimentación numérica, presentamos más detalles de este algoritmo; en lo que sigue, nos referiremos a él como **Algoritmo 1**.

## Verificación de hipótesis

El **Algoritmo 1** converge con tasa superlineal, y esta convergencia está condicionada a que satisfaga las siguientes hipótesis:

**H.1** Existe una solución $\boldsymbol{y}^*$ del sistema $\Phi_\tau(\boldsymbol{y}) = 0$.

**H.2** Las funciones $F_i$ son continuamente diferenciables y su gradiente es una función semisuave.

**H.3** La matriz jacobiana de $F$ es *Lipschitz* continua en una vecindad de $\boldsymbol{y}^*$.

A continuación, analizaremos hasta qué punto la función $F$ satisface las hipótesis que requiere el **Algoritmo 1** para alcanzar la convergencia. Para ello, usamos (5) y deducimos que la matriz Jacobiana de $F$, está dada por

$$F'(\boldsymbol{y}) = \begin{bmatrix} B - y_{n+1}A & -A\boldsymbol{x} \\ \boldsymbol{e}^T & 0 \end{bmatrix}. \tag{9}$$

Estas expresiones nos facilitarán la verificación de las hipótesis que garantizan la viabilidad del **Algoritmo 1** para este problema particular.

**H.1** En general, para determinar si un problema de complementariedad tiene solución, bastaría verificar que la función que define el problema es estrictamente monótona (**Ferris y Pang**, 1997); sin embargo, para el problema (4) con $F$ definida en (5), este camino conduce a imponer condiciones muy fuertes sobre $A$ y $B$ para garantizar la monotonía de $F$ y con ello, la solución al problema.

Una alternativa para garantizar la existencia de al menos una solución del problema $VPC(A,B)$ es imponer condiciones menos restrictivas sobre $A$ y $B$. Esto fue hecho en (**Queiroz, Júdice, y Humes**, 2004), donde los autores demuestran que, si todas las componentes de $A$ son positivas y $B = I_n$, el problema $VPC(A,B)$ tiene solución. Adicionalmente, si $A$ es simétrica la solución es única. Este resultado, nos permite verificar la hipótesis **H1** y diseñar experimentos numéricos, como veremos más adelante.

Vale la pena mencionar que la cantidad máxima de soluciones que puede tener el problema $VPC(A,B)$ es $n2^{n-1}$ (**Seeger**, 1999).

**H.2** Las funciones $F_i$ son diferenciables. En efecto, cuando $1 \leq i \leq n$, su respectivo gradiente está dado por

$$\nabla F_i(\boldsymbol{y}) = \begin{pmatrix} B_{i1} + y_{n+1}A_{i1} \\ \vdots \\ B_{in} + y_{n+1}A_{n1} \\ -A_{i1}y_1 - \cdots - A_{in}y_n \end{pmatrix}$$

y, para $n+1$, el $\nabla F_i(\boldsymbol{y})$ es el vector de unos de $n+1$ componentes. Cada uno diferenciable, con además para $1 \leq i \leq n$

$$\nabla^2 F_i(\boldsymbol{y}) = \begin{pmatrix} 0 & \cdots & 0 & A_{i1} \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & A_{in} \\ 1 & \cdots & 1 & 0 \end{pmatrix}$$

y, para $i = n+1$, la $\nabla^2 F_i(\boldsymbol{y})$ es la matriz nula de orden $n+1$. Esto demuestra que cada gradiente $\nabla F_i$ es continuamente diferenciable; por lo tanto, es también una función semisuave[1].

**H.3** Probemos que $F'$ es *Lipschitz* continua en todo su dominio. Sean $\boldsymbol{y}, \boldsymbol{z} \in \mathbb{R}^{n+1}$ de tal forma que

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{x} \\ y_{n+1} \end{bmatrix} \text{ y } \boldsymbol{z} = \begin{bmatrix} \boldsymbol{w} \\ z_{n+1} \end{bmatrix}$$

Considérese la diferencia

$$\begin{aligned} \left\|F'(\boldsymbol{y}) - F'(\boldsymbol{z})\right\|_1 &= \left\| \begin{bmatrix} (B - y_{n+1}A) & -A\boldsymbol{x} \\ \boldsymbol{e}^T & 0 \end{bmatrix} - \begin{bmatrix} (B - z_{n+1}A) & -A\boldsymbol{w} \\ \boldsymbol{e}^T & 0 \end{bmatrix} \right\|_1 \\ &= \left\| \begin{bmatrix} (z_{n+1} - y_{n+1})A & -A(\boldsymbol{x} - \boldsymbol{w}) \\ 0^T & 0 \end{bmatrix} \right\|_1 \\ &= \max\left\{ \left\|(z_{n+1} - y_{n+1})A\right\|_1, \left\|-A(\boldsymbol{x} - \boldsymbol{w})\right\|_1 \right\} \end{aligned}$$

[1] Sea $G\colon \mathbb{R}^n \to \mathbb{R}^n$ una función localmente *Lipschitz* continua, se dice que $G$ es semisuave en $\boldsymbol{x} \in \mathbb{R}^n$, si $G$ en $\boldsymbol{x}$ es direccionalmente diferenciable y,

$$G(\boldsymbol{x} + h) - G(\boldsymbol{x}) - Hh = o(\|h)\|)$$

para todo $H \in \partial_C G(\boldsymbol{x} + h)$ cuando $h \longrightarrow 0$.

Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623

**Caso 1.** Si máx $\{\|(z_{n+1}-y_{n+1})A\|_1, \|-A(\boldsymbol{x}-\boldsymbol{w})\|_1\} = \|(z_{n+1}-y_{n+1})A\|_1$

$$\begin{aligned}\left\|F'(\boldsymbol{y})-F'(\boldsymbol{z})\right\|_1 &= \|(z_{n+1}-y_{n+1})A\|_1 \\ &= |y_{n+1}-z_{n+1}|\,\|A\|_1 \\ &\leq \|A\|_1\,(|y_{n+1}-z_{n+1}|+\|\boldsymbol{x}-\boldsymbol{w}\|_1) \\ &\leq \|A\|_1\,\|\boldsymbol{y}-\boldsymbol{z}\|_1\,.\end{aligned}$$

**Caso 2.** Si máx $\{\|(z_{n+1}-y_{n+1})A\|, \|-A(\boldsymbol{x}-\boldsymbol{w})\|\} = \|-A(\boldsymbol{x}-\boldsymbol{w})\|_1$

$$\begin{aligned}\left\|F'(\boldsymbol{y})-F'(\boldsymbol{z})\right\|_1 &= \|-A(\boldsymbol{x}-\boldsymbol{w})\|_1 \\ &\leq \|A\|_1\,\|\boldsymbol{x}-\boldsymbol{w}\|_1 \\ &\leq \|A\|_1\,(\|\boldsymbol{x}-\boldsymbol{w}\|_1+|y_{n+1}-z_{n+1}|) \\ &\leq \|A\|_1\,\|\boldsymbol{y}-\boldsymbol{z}\|_1\,,\end{aligned}$$

por lo tanto, $F'$ es *Lipschitz* continua con constante $\|A\|_1$.

Al satisfacer de manera razonable las hipótesis del algoritmo, se procede a mostrar los resultados numéricos.

## Experimentación numérica

En esta sección, resolvemos numéricamente la reformulación del problema $VPC2(A,B)$ como un problema de complementariedad no lineal, el cual a su vez, lo hemos reformulado como el problema de minimización (8). Para ello usaremos el algoritmo propuesto en (**Arias y cols.**, 2017), el cual hemos denominado en este documento **Algoritmo 1**. Este es un algoritmo cuasi-*Newton*, exactamente secante de cambio mínimo (**Martínez**, 1992), globalizado mediante una estrategia de búsqueda lineal (**Dennis y Schnabel**, 1996). Para mayor claridad en la lectura de este documento lo incluimos a continuación.

**Algoritmo 1** Cuasi *Newton*

**Entrada:** vector inicial $y_0 \in \mathbb{R}^{n+1}$; $\varepsilon$ : tolerancia; $N$ : número máximo de iteraciones.
**Salida:** vector solución $y^* \in \mathbb{R}^{n+1}$ o mensaje de error.

$k \leftarrow 0$
Calcule $B_k \in \partial\Phi_\tau(\boldsymbol{y}_k)$ como en (15) de (**Arenas, Martínez, y Pérez**, 2015).
**mientras** $\|B_k\Phi_\tau(\boldsymbol{y}_k)\| < \varepsilon$ o $k < N$ **hacer**
  **si** el sistema $B_k\boldsymbol{d}_k = -\Phi_\tau(\boldsymbol{y}_k)$ tiene solución **entonces**
    calcular $\boldsymbol{d}_k$ del sistema $B_k\boldsymbol{d}_k = -\Phi_\tau(\boldsymbol{y}_k)$
  **si no**
    $\boldsymbol{d}_k \leftarrow B_k\Phi_\tau(\boldsymbol{y}_k)$
  **fin si**
  **mientras** $\Psi_\tau(\boldsymbol{y}_k+\alpha\boldsymbol{d}_k) \leq \Psi_\tau(\boldsymbol{y}_k)+\sigma\alpha\Phi_\tau(\boldsymbol{y}_k)^T B_k^T\boldsymbol{d}_k$ **hacer**
    $\alpha \leftarrow \frac{1}{2}\alpha$
  **fin mientras**
  $\boldsymbol{y}_k \leftarrow \boldsymbol{y}_k+\alpha\boldsymbol{d}_k$
  $\boldsymbol{s}_k \leftarrow \alpha\boldsymbol{d}_k$
  $\boldsymbol{z}_k \leftarrow \Psi_\tau(\boldsymbol{y}_{k+1})-\Psi_\tau(\boldsymbol{y}_k)$
  $A_{k+1} \leftarrow Actualizar(A_k,\boldsymbol{y}_k,\boldsymbol{s}_k,\boldsymbol{z}_k,op)$ mediante el **Algoritmo 2**.
  $k \leftarrow k+1$
**fin mientras**
**devolver** $\boldsymbol{y}_k$

El **Algoritmo 1**, hace un llamado al **Algoritmo 2**, que selecciona la actualización secante, que actualizará la matriz $B_k$ y que será usada en la siguiente iteración. Las actualizaciones que consideraremos son las siguientes: buena de *Broyden* (**Broyden**, 1965), mala de *Broyden* (**Brown y Saad**, 1994), *Schubert* (**Schubert**, 1970), COLUM (**Martínez**, 2000) e ICUM (**Martínez**, 1992).

**Algoritmo 2** Función Actualizar

**Entrada:** $A_k \in \mathbb{R}^{n\times n}$, vectores $\boldsymbol{x}_k, \boldsymbol{s}_k, \boldsymbol{y}_k \in \mathbb{R}^{n+1}$, el método: $op$
**Salida:** $A_{k+1} \in \mathbb{R}^{n\times n}$,

**si** $op = 1$ **entonces**
Actualización buena de *Broyden*(**Broyden**, 1965)

$$\Delta \leftarrow \frac{(\boldsymbol{y}_k - A_k\boldsymbol{s}_k)\,\boldsymbol{s}_k^T}{\boldsymbol{s}_k^T\boldsymbol{s}_k}$$

**si no**, **si** $op = 2$ **entonces**
Actualización mala de *Broyden* (**Brown y Saad**, 1994)

$$\Delta \leftarrow \frac{(\boldsymbol{y}_k - A_k\boldsymbol{s}_k)\,\boldsymbol{y}_k^T A_k}{\boldsymbol{y}_k^T A_k\boldsymbol{s}_k}$$

**si no**, **si** $op = 3$ **entonces**
Actualización de *Schubert* (**Schubert**, 1970)

$$\Delta \leftarrow \frac{F(\boldsymbol{x}_{k+1})\,\boldsymbol{s}_k^T}{\boldsymbol{s}_k^T\boldsymbol{s}_k}$$

**si no**, **si** $op = 4$ **entonces**
Actualización de COLUM (**Martínez**, 2000). Sea $j$ tal que $\|\boldsymbol{s}_k\|_\infty = |\,[s_k]_j\,|$

$$\Delta \leftarrow \frac{(\boldsymbol{y}_k - A_k\boldsymbol{s}_k)\,\boldsymbol{e}_{jk}^T}{\boldsymbol{e}_{jk}^T\boldsymbol{s}_k}$$

**si no**, **si** $op = 5$ **entonces**
Actualización de ICUM (**Martínez**, 1992). Sea $j$ tal que $\|\boldsymbol{y}_k\|_\infty = |\boldsymbol{e}_{jk}^T y_k|$

$$\Delta \leftarrow \frac{(\boldsymbol{y}_k - A_k\boldsymbol{s}_k)\,\boldsymbol{e}_{jk}^T A_k}{\boldsymbol{e}_{jk}^T A_k\boldsymbol{s}_k}$$

**fin si**
$A_{k+1} \leftarrow A_k + \Delta$
**devolver** $A_k$

Se consideró como algoritmo para hacer las comparaciones respectivas el método de *Newton* generalizado, el cual llamaremos **Algoritmo 3**, y lo incluimos a continuación para claridad en la lectura de la sección. Por supuesto, aprovechamos la estructura del jacobiano de $F$ indicada en (9) ya que, no es muy costosa su evaluación, ni su implementación.

Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623

**Algoritmo 3** *Newton* generalizado

**Entrada:** vector inicial $y_0 \in \mathbb{R}^{n+1}$; $\varepsilon$ : tolerancia; $N$ : número máximo de iteraciones.
**Salida:** vector solución $y^* \in \mathbb{R}^{n+1}$ o mensaje de error.

$k \leftarrow 0$
Calcule $H_k \in \partial\Phi_\tau(\boldsymbol{y}_k)$ como en (14) de (**Arenas y cols.**, 2015).
**mientras** $\|H_k\Phi_\tau(\boldsymbol{y}_k)\| < \varepsilon$ o $k < N$ **hacer**
  **si** el sistema $H_k\boldsymbol{d}_k = -\Phi_\tau(\boldsymbol{y}_k)$ tiene solución **entonces**
    calcular $\boldsymbol{d}_k$ del sistema $H_k\boldsymbol{d}_k = -\Phi_\tau(\boldsymbol{y}_k)$
  **si no**
    $\boldsymbol{d}_k \leftarrow H_k\Phi_\tau(\boldsymbol{y}_k)$
  **fin si**
  **mientras** $\Psi_\tau(\boldsymbol{y}_k + \alpha\boldsymbol{d}_k) \leq \Psi_\tau(\boldsymbol{y}_k) + \sigma\alpha\Phi_\tau(\boldsymbol{y}_k)^T H_k^T\boldsymbol{d}_k$ **hacer**
    $\alpha \leftarrow \frac{1}{2}\alpha$
  **fin mientras**
  $\boldsymbol{y}_k \leftarrow \boldsymbol{y}_k + \alpha\boldsymbol{d}_k$
  $k \leftarrow k+1$
  Calcule $H_k \in \partial\Phi_\tau(\boldsymbol{y}_k)$ como en (14) de (**Arenas y cols.**, 2015).
**fin mientras**
**devolver** $\boldsymbol{y}_k$

### Construcción de los problemas

Para construir los problemas $VPC(A,B)$ usamos el lineamiento descrito en (**Júdice y cols.**, 2007); es decir, dos categorías de problemas: simétricos y no simétricos; en todos, la matriz $B$ es la identidad de tamaño $n \times n$. La matriz $A$ se construye como se indica en la **Tabla 1**, donde *Escala* significa el valor más grande que puede tomar cada componente de $A$, *randn(n)* es una matriz aleatoria de tamaño $n \times n$ y $p$ está dado por (3), generando un total de ocho configuraciones de parámetros.

**Tabla 1.** Selección de las matrices $A$ y $B$ en la experimentación.

| Tipo de problema | $p$ | Escala | $A$ | $B$ |
|---|---|---|---|---|
| Simétrico | 1 y 6 | 1 y 50 | $V = Escala \times randn(n)$<br>$U = triu(V)$<br>$A = U + U^T$ | $I_n$ |
| No simétrico | 1 y 6 | 1 y 50 | $A = Escala \times randn(n)$ | $I_n$ |

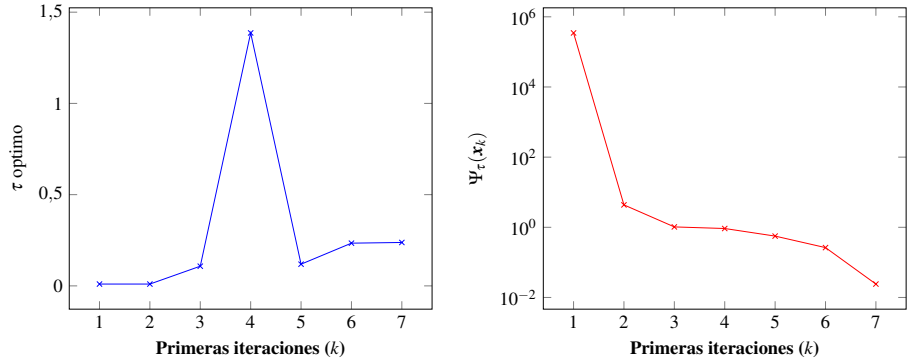
Para escribir los códigos de los algoritmos y de las funciones objetivo de cada problema usamos el software MATLAB®. Realizamos las pruebas numéricas en un computador Intel (R) Core (TM) i7-9700F CPU de 3.00 GHz.

### Experimento preliminar

Uno de los interrogantes planteados antes de comenzar la experimentación fue el siguiente, ¿cuál es el valor de $\tau$ adecuado para la función $\varphi_\tau$ definida en (6)? En (**Kanzow y Kleinmichel**, 1998) se presentó un algoritmo dinámico para variar $\tau$ con buenos resultados, sin embargo, en el caso particular del problema $VPC(A,B)$, en pruebas preliminares no se obtuvo un decrecimiento eficaz, al compararlo con el valor de $\tau = 2$ estático, el cual es un valor muy utilizado en funciones de complementariedad, esto llevó a diseñar el siguiente experimento.

Encontrar en cada paso el valor de $\tau$ en el cual se pueda obtener el mayor decrecimiento de

la función objetivo $\Psi_\tau$. Para ello, se utilizaron diferentes valores de $\tau$ desde 0.01 hasta 3.99 y se seleccionó para cada paso aquel valor de $\tau$ en que se obtuvo el máximo descenso de $\Psi_\tau$ utilizando el método de *Newton* generalizado. Luego se repitió esta experiencia 10000 veces, los promedios de los 9110 ejecuciones exitosas se muestran en la **Figura 1**.




**(a)** Valores de $\tau$ con mayor descenso

**(b)** Valores de $\Psi_\tau$ obtenidos

**Figura 1.** Valores de $\tau$ que producen mayor decrecimiento y su respectivo $\Psi_\tau(\boldsymbol{x}_k)$ en las primeras 7 iteraciones.

Podemos observar que el mayor decrecimiento se obtiene en las primeras 3 iteraciones, en los cuales $\lambda$ toma valores muy pequeños, es por ello que hemos decidido realizar todos los experimentos con un valor de $\lambda = 10^{-2}$.

### Experimento principal

Realizamos para cada valor de $n$, cada valor de *Escala* y cada valor de $p$, 1000 ejecuciones; en cada ejecución se generó aleatoriamente un problema. El punto inicial $\boldsymbol{y}_0$ también se generó aleatoriamente con el comando $randn(n+1,1)$. Cada uno de estos problemas, se resolvió con los algoritmos 1 y 2 respectivamente.

Los resultados más relevantes obtenidos en esta experimentación, se presentan en las **Tablas 2** a **4**, donde **n**, es el tamaño de las matrices $A$ y $B$, $p$, es el valor de la constante dada en (3); *escala*, es el valor más grande que puede tener cada componente de la matriz $A$; en cada una de las actualizaciones **E**, indica el porcentaje de éxitos con cada una de las actualizaciones; es decir, el número de veces que se obtuvo convergencia entre 1000 expresado como porcentaje; **Iter**, indica el promedio de iteraciones redondeado a un entero de las ejecuciones que tuvieron éxito; finalmente, **tiempo** indica el promedio de tiempo CPU redondeado a cuatro decimales en las ejecuciones que tuvieron éxito.

Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623

**Tabla 2.** Caso asimétrico con $p = 1$ y $Escala = 1$.

| | | *Newton* | | | *Broyden B* | | | *Broyden M* | | |
|---|---|---|---|---|---|---|---|---|---|---|
| n | p | E | Iter | tiempo | E | Iter | tiempo | E | Iter | tiempo |
| 2 | 1 | 97.1 | 16 | 0.0012 | 85.1 | 23 | 0.0016 | 27.9 | 10 | 0.0003 |
| 3 | 1 | 99.1 | 14 | 0.0009 | 91.4 | 24 | 0.0017 | 28.1 | 12 | 0.0003 |
| 4 | 1 | 99.4 | 13 | 0.0008 | 91.1 | 22 | 0.0014 | 24.2 | 13 | 0.0004 |
| 5 | 1 | 99.5 | 12 | 0.0007 | 90.8 | 24 | 0.0016 | 24.9 | 14 | 0.0004 |
| 10 | 1 | 99.7 | 13 | 0.0009 | 92.1 | 34 | 0.0025 | 17.9 | 18 | 0.0006 |
| 20 | 1 | 99.7 | 16 | 0.0015 | 89.1 | 56 | 0.0063 | 9.9 | 22 | 0.0011 |
| 30 | 1 | 99.9 | 20 | 0.0026 | 85.6 | 70 | 0.0098 | 8.3 | 24 | 0.0016 |
| 50 | 1 | 98.1 | 23 | 0.0052 | 72.5 | 105 | 0.0246 | 5.1 | 27 | 0.0030 |
| 100 | 1 | 92.5 | 26 | 0.0297 | 47.3 | 123 | 0.1364 | 2.1 | 31 | 0.0207 |
| | | *Schubert* | | | COLUM | | | ICUM | | |
| n | p | E | Iter | tiempo | E | Iter | tiempo | E | Iter | tiempo |
| 2 | 1 | 35.7 | 13 | 0.0007 | 76.9 | 21 | 0.0017 | 34.7 | 9 | 0.0003 |
| 3 | 1 | 33.4 | 15 | 0.0008 | 80.3 | 19 | 0.0013 | 30.2 | 11 | 0.0004 |
| 4 | 1 | 33.8 | 16 | 0.0007 | 77.5 | 19 | 0.0012 | 28.8 | 11 | 0.0004 |
| 5 | 1 | 32.0 | 17 | 0.0007 | 74.7 | 22 | 0.0015 | 24.2 | 13 | 0.0004 |
| 10 | 1 | 17.9 | 26 | 0.0019 | 52.0 | 27 | 0.0018 | 13.1 | 15 | 0.0006 |
| 20 | 1 | 5.9 | 41 | 0.0043 | 21.0 | 30 | 0.0024 | 8.5 | 17 | 0.0010 |
| 30 | 1 | 2.5 | 47 | 0.0073 | 14.5 | 31 | 0.0033 | 7.1 | 18 | 0.0013 |
| 50 | 1 | 0.7 | 39 | 0.0068 | 7.4 | 33 | 0.0061 | 4.0 | 20 | 0.0024 |
| 100 | 1 | 0.1 | 27 | 0.0254 | 4.5 | 36 | 0.0418 | 2.5 | 19 | 0.0138 |

En los resultados presentados en la **Tabla 2** se destacan las actualizaciones *Broyden M*. e ICUM por su rapidez de convergencia, sin embargo, tienen poco éxito, por lo cual afirmamos que no son muy efectivas al tomar valores iniciales aleatorios, si el problema tuviese una forma de iniciar el algoritmo en una vecindad pequeña de la solución esta falencia pudiera ser superada. Por otro lado, los métodos de *Newton* generalizado y *Broyden B* son los métodos más eficaces para este caso no simétrico con valores pequeños de las entradas de la matriz $A$.

**Tabla 3.** Caso asimétrico con $p = 6$ y $Escala = 50$.

| | | *Newton* | | | *Broyden B* | | | *Broyden M* | | |
|---|---|---|---|---|---|---|---|---|---|---|
| n | p | E | Iter | tiempo | E | Iter | tiempo | E | Iter | tiempo |
| 2 | 6 | 77.4 | 14 | 0.0007 | 24.0 | 54 | 0.0030 | 0.8 | 18 | 0.0013 |
| 3 | 6 | 78.6 | 14 | 0.0006 | 10.0 | 59 | 0.0047 | 1.1 | 18 | 0.0012 |
| 4 | 6 | 80.6 | 14 | 0.0007 | 4.4 | 59 | 0.0050 | 0.6 | 35 | 0.0031 |
| 5 | 6 | 81.1 | 14 | 0.0007 | 2.7 | 42 | 0.0033 | 1.3 | 24 | 0.0024 |
| 10 | 6 | 82.4 | 14 | 0.0008 | 1.7 | 41 | 0.0035 | 0.8 | 54 | 0.0064 |
| 20 | 6 | 84.5 | 15 | 0.0012 | 0.5 | 81 | 0.0129 | 0.5 | 42 | 0.0067 |
| 30 | 6 | 82.9 | 16 | 0.0017 | 0.4 | 106 | 0.0246 | 0.2 | 51 | 0.0088 |
| 50 | 6 | 83.3 | 17 | 0.0030 | 0.5 | 107 | 0.0327 | 0.4 | 75 | 0.0293 |
| 100 | 6 | 86.9 | 19 | 0.0178 | 1.3 | 52 | 0.0793 | 0.3 | 101 | 0.1431 |
| | | *Schubert* | | | COLUM | | | ICUM | | |
| n | p | E | Iter | tiempo | E | Iter | tiempo | E | Iter | tiempo |
| 2 | 6 | 0.7 | 23 | 0.0021 | 34.9 | 49 | 0.0040 | 0.4 | 20 | 0.0012 |
| 3 | 6 | 0.0 | - | - | 15.4 | 66 | 0.0056 | 0.5 | 11 | 0.0004 |
| 4 | 6 | 0.2 | 50 | 0.0056 | 7.5 | 50 | 0.0037 | 0.4 | 12 | 0.0004 |
| 5 | 6 | 0.5 | 15 | 0.0004 | 3.3 | 56 | 0.0044 | 0.1 | 12 | 0.0004 |
| 10 | 6 | 0.1 | 17 | 0.0007 | 1.1 | 27 | 0.0015 | 0.2 | 18 | 0.0007 |
| 20 | 6 | 0.2 | 13 | 0.0006 | 0.7 | 29 | 0.0018 | 0.1 | 22 | 0.0011 |
| 30 | 6 | 0.0 | - | - | 0.3 | 13 | 0.0010 | 0.2 | 13 | 0.0009 |
| 50 | 6 | 0.4 | 25 | 0.0031 | 0.0 | - | - | 0.0 | - | - |
| 100 | 6 | 0.0 | - | - | 0.1 | 40 | 0.0361 | 0.0 | - | - |

Los resultados de la **Tabla 3** muestran que el método más eficiente en el caso no simétrico con entradas menores o iguales que 50, sigue siendo el método de *Newton* generalizado, seguido por la actualización de *Broyden B* también la actualización ICUM es la que converge más rápidamente, aunque tiene poco éxito.

Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623

**Tabla 4.** Caso simétrico con $p = 1$ y $Escala = 1$.

| | | Newton | | | *Broyden B* | | | *Broyden M* | | |
|---|---|---|---|---|---|---|---|---|---|---|
| n | p | E | Iter | tiempo | E | Iter | tiempo | E | Iter | tiempo |
| 2 | 1 | 100.0 | 13 | 0.0032 | 90.0 | 21 | 0.0017 | 50.0 | 10 | 0.0003 |
| 3 | 1 | 100.0 | 14 | 0.0009 | 90.0 | 17 | 0.0009 | 20.0 | 9 | 0.0002 |
| 4 | 1 | 100.0 | 10 | 0.0004 | 90.0 | 34 | 0.0035 | 0.0 | - | - |
| 5 | 1 | 100.0 | 10 | 0.0004 | 90.0 | 29 | 0.0018 | 20.0 | 14 | 0.0004 |
| 10 | 1 | 100.0 | 16 | 0.0010 | 100.0 | 44 | 0.0042 | 10.0 | 15 | 0.0005 |
| 20 | 1 | 100.0 | 40 | 0.0055 | 80.0 | 46 | 0.0047 | 0.0 | - | - |
| 30 | 1 | 100.0 | 32 | 0.0052 | 60.0 | 68 | 0.0091 | 0.0 | - | - |
| 50 | 1 | 90.0 | 42 | 0.0115 | 70.0 | 116 | 0.0302 | 20.0 | 28 | 0.0033 |
| 100 | 1 | 70.0 | 17 | 0.0109 | 30.0 | 160 | 0.1602 | 0.0 | - | - |
| | | *Schubert* | | | COLUM | | | ICUM | | |
| n | p | E | Iter | tiempo | E | Iter | tiempo | E | Iter | tiempo |
| 2 | 1 | 30.0 | 17 | 0.0017 | 70.0 | 23 | 0.0024 | 30.0 | 6 | 0.0014 |
| 3 | 1 | 10.0 | 7 | 0.0002 | 70.0 | 23 | 0.0017 | 70.0 | 11 | 0.0004 |
| 4 | 1 | 0.0 | - | - | 50.0 | 16 | 0.0006 | 30.0 | 14 | 0.0005 |
| 5 | 1 | 30.0 | 14 | 0.0004 | 60.0 | 27 | 0.0015 | 30.0 | 12 | 0.0004 |
| 10 | 1 | 10.0 | 16 | 0.0005 | 50.0 | 70 | 0.0114 | 10.0 | 23 | 0.0008 |
| 20 | 1 | 0.0 | - | - | 20.0 | 23 | 0.0018 | 0.0 | - | - |
| 30 | 1 | 0.0 | - | - | 10.0 | 18 | 0.0015 | 10.0 | 26 | 0.0022 |
| 50 | 1 | 0.0 | - | - | 10.0 | 21 | 0.0026 | 0.0 | - | - |
| 100 | 1 | 0.0 | - | - | 0.0 | - | - | 0.0 | - | - |

De la **Tabla 4** deducimos que para el caso simétrico, el método de *Newton* generalizado es el mas indicado, por efectividad y eficiencia, en contraste, el método de *Shubert* no parece ser recomendable para problemas de tamaño mayor a 5. También llama la atención que en muchos casos no se alcanza la convergencia en algunos métodos cuasi-*Newton*, salvo el método *Broyden B*ueno, que es muy confiable.

## Comentarios finales

En este artículo, utilizamos un método cuasi-*Newton* para resolver el problema de valores propios complementarios basado en su reformulación como un problema de complementariedad no lineal. Un método de este tipo es una buena alternativa cuando las derivadas del sistema son muy costosas o difíciles de obtener. Particularmente en este caso no es muy costoso su cómputo, aún así, por las valiosas aplicaciones que tiene el reducir el tiempo de ejecución es un incentivo para utilizar esta propuesta.

Como pudimos verificar en los resultados de las pruebas numéricas, la cantidad de experimentos numéricos que se hicieron fue bastante amplia, por lo cual podemos concluir que los métodos cuasi-*Newton* aplicados a este problema pueden ser competitivos frente al método de *Newton* generalizado, especialmente el método de *Broyden B*ueno. Por otro lado también observamos que al aumentar la dimensión del problema el número de éxito se reduce considerablemente, creemos que la razón de este fenómeno es que la norma del vector solución es igual a $p$ y al ser muy pequeño (1 o 6) y la dimensión muy grande, obliga a que cada componente del vector solución sea casi cero, lo cual genera problemas de estabilidad en el algoritmo.

Por otro lado, en cuanto a la selección del "mejor" valor de $\tau$ en (6) todavía que da mucho por hacer. Dicho lo anterior concluimos que este algoritmo es una buena alternativa, pues el costo computacional es menor en comparación con el método de *Newton* generalizado.

## Agradecimientos

Expresamos nuestro agradecimiento a la Universidad del Cauca por brindar tiempo para este trabajo a través del proyecto de investigación VRI ID 5247 y a los árbitros anónimos por las sugerencias constructivas lo cual nos permitió mejorar la presentación de este artículo.

## Contribución de los autores

FA: programación, diseño de experimentos, análisis de resultados, discusión y conclusiones; CA: Construcción y desarrollo de la propuesta de investigación, análisis de resultados; RP: Organización de la información, análisis de resultados, revisión de forma y contenido del documento, discusión y conclusiones.

## Conflicto de intereses

Los autores declaran no tener conflicto de intereses con respecto al contenido de este artículo.

## Referencias

**Arenas, F., Martínez, H. J., y Pérez, R.** (2014). Redefinición de la función de complementariedad de Kanzow. *Revista de Ciencias*, *18*(2), 111-122. doi: 10.25100/rc.v18i2.6105

**Arenas, F., Martínez, H. J., y Pérez, R.** (2015). Least change secant update methods for nonlinear complementarity problem. *Ingeniería y Ciencia*, *11*(21), 11-36. doi: 10.17230/ingciencia.11.21.1

**Arenas, F., Martínez, H., y Pérez, R.** (2020, May). A local jacobian smoothing method for solving nonlinear complementarity problems. *Universitas Scientiarum*, *25*(1), 149-174. doi: 10.11144/Javeriana.SC25-1.aljs

**Arias, C., Martínez, H., y Pérez, R.** (2017). A nonsmooth global cuasi-newton method for nonlinear complementarity problems. *Computer Methods in Applied Mechanics and Engineering*, *13*(1), 1-15.

**Brown, P. N., y Saad, Y.** (1994). Convergence theory of nonlinear Newton–Krylov algorithms. *SIAM Journal on Optimization*, *4*(2), 297-330.

**Broyden, C. G.** (1965). A Class of Methods for Solving Nonlinear Simultaneous Equations. *Mathematics of Computation*, *19*(92), 577-593.

**Clarke, F. H.** (1975). Generalized gradients and applications. *Transactions of the american society*, *205*(1), 247-262.

**Dennis, J. E., y Schnabel, R. B.** (1996). *Numerical methods for unconstrained optimization and nonlinear equations*. Society for Industrial and Applied Mathematics.

**Facchinei, F., y Pang, J.** (2003). *Finite-dimensional variational inequalities and complementarity problems* (Vol. II; Springer, Ed.). Springer-Verlag. doi: 10.1007/b97544

**Ferris, M. C., y Pang, J. S.** (1997). Engineering and economic applications of complementarity problems. *SIAM Review*, *39*(1), 669-713.

**Iusem, A. N., Júdice, J. J., Sessa, V., y Sarabando, P.** (2019). Splitting methods for the eigenvalue complementarity problem. *Optimization Methods and Software*, *34*(6), 1184-1212. doi: 10.1080/10556788.2018.1479408

**Júdice, J. J., Sherali, H. D., y Ribeiro, I.** (2007). The eigenvalue complementarity problem. *Computational Optimization and Applications*, *37*(2), 139-156. doi: 10.1007/s10589-007-9017-0

**Kanzow, C., y Kleinmichel, H.** (1998). A new class of semismooth Newton-type methods for nonlinear complementarity problems. *Computational Optimization and Applications*, *11*(3), 227-251. doi: 10.1023/A:1026424918464

**Martínez, J. M.** (1992). On the relation between two local convergence theories of least change secant update methods. *Mathematics of Computation*, *59*(200), pp. 457-481.

**Martínez, J. M.** (2000). Practical quasi-Newton methods for solving nonlinear systems. *Journal of Computational and Applied Mathematics*, *124*(1), 97-121. (Nu- merical Analysis 2000. Vol. IV: Optimization and Nonlinear Equations) doi: https://doi.org/10.1016/S0377-0427(00)00434-9

**Pérez, R., Arenas, F., Martínez, H., y Arias, C.** (2019). *El problema de complementariedad no lineal* (Vol. 1). Programa editorial Universidad del Valle.

Rev. Acad. Colomb. Cienc. Ex. Fis. Nat. 46(179):325-338, abril-junio de 2022
doi: https://doi.org/10.18257/raccefyn.1623

**Pinto da Costa, A., Martins, J., Figueiredo, I. N., y Júdice, J.** (2004). The directional instability problem in systems with frictional contacts. *Computer Methods in Applied Mechanics and Engineering*, *193*(3), 357-384. doi: 10.1016/j.cma.2003.09.013

**Queiroz, M., Júdice, J., y Humes, C. J.** (2004). The symmetric eigenvalue complementarity problem. *Mathematics of Computation*, *73*(248), 1849-1863.

**Schubert, L. K.** (1970). Modification of a quasi-Newton method for nonlinear equations with a sparse jacobian. *Mathematics of Computation*, *24*, 27-30.

**Seeger, A.** (1999). Eigenvalue analysis of equilibrium processes defined by linear complementarity conditions. *Linear Algebra and its Applications*, *292*(1), 1-14. doi: https://doi.org/10.1016/S0024-3795(99)00004-X

**Sherali, H. D., y Tuncbilek, C. H.** (1992). A global optimization algorithm for polyno- mial programming problems using a reformulation-linearization technique. *Journal of Global Optimization*, *2,* 101-112. doi: 10.1007/BF00121304

**Sánchez, W., Pérez, R., y Martínez, H.** (2021). Un algoritmo global con jacobiano suavizado para problemas de complementariedad no lineal. *Revista integración, temas de matemáticas*, *39*(2), 191-215. doi: 10.18273/revint.v39n2-20210004

**Yong, L.** (2010). Nonlinear complementarity problem and solution methods. En *Proceedings of the 2010 international conference on artificial intelligence and computational intelligence: Part i* (p. 461-469). Springer-Verlag.