

**EVALUACIÓN Y COMPARACIÓN DE LOS METODOS UPSO Y NEWTON
RAPHSON PARA EL ANALISIS DE FLUJO DE CARGAS EN UN SISTEMA DE
POTENCIA**

**CAMILO ANDRÉS PINZÓN CEBALLOS
EDWIN ARDILA PEDRAZA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRONICA Y
TELECOMUNICACIONES
BUCARAMANGA**

2013

**EVALUACIÓN Y COMPARACIÓN DE LOS METODOS UPSO Y NEWTON
RAPHSON PARA EL ANALISIS DE FLUJO DE CARGAS EN UN SISTEMA DE
POTENCIA**

**CAMILO ANDRÉS PINZÓN CEBALLOS
EDWIN ARDILA PEDRAZA**

**Trabajo de Grado para optar al título de
Ingeniero Electricista**

**Director
PhD. CARLOS RODRIGO CORREA CELY**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRONICA Y
TELECOMUNICACIONES
BUCARAMANGA**

2013

DEDICATORIA

A mi familia por mostrarme que el camino a la superación es la educación.

A Adriana por ayudarme y apoyarme en esos momentos de intranquilidad.

A mis amigos, por mostrarme que hay mucho más que aprender en la universidad y que no es solo para estudiar.

A todas esas personas importantes en mi vida, que entre otras cosas, me tuvieron mucha paciencia.

Camilo Andrés Pinzón

DEDICATORIA

A mi hijo Adrián quien fue mi fortaleza en momentos de debilidad.

A mi amada por su comprensión y apoyo incondicional

*A toda mi familia que siempre estuvo ahí para apoyarme y nunca
dudaron que esto sería una realidad*

*A mis compañeros quienes aportaron sabiduría e hicieron de esta
etapa algo sencillamente inolvidable*

Edwin Ardila Pedraza

AGRADECIMIENTOS

A todas las personas, compañeros y familiares, que de alguna u otra forma colaboraron en esta empresa, e hicieron posible que este proyecto terminara.

A nuestros maestros, que cada semestre nos dieron las herramientas necesarias para hacer esto posible.

Al profesor Carlos Rodrigo Correa Cely, por su valiosa ayuda y colaboración en la realización de este proyecto.

A Gokú por salvarnos tantas veces de la aniquilación.

RESUMEN

TÍTULO

EVALUACIÓN Y COMPARACIÓN DE LOS METODOS UPSO Y NEWTON RAPHSON PARA EL ANALISIS DE FLUJO DE CARGAS EN UN SISTEMA DE POTENCIA*

AUTORES

Camilo Andrés Pinzón Ceballos**

Edwin Ardila Pedraza**

PALABRAS CLAVE

UPSO, Newton-Raphson, DIgSILENT, Flujos de Carga, Sistemas de Potencia

El presente documento contiene la aplicación del algoritmo estocástico de Optimización por Enjambre de Partículas Unificado (UPSO) en ejemplos tipo de cálculo de flujos de carga para sistemas de potencia. Se inicia con una breve explicación sobre la fundamentación teórica del método y luego se procede a describir los resultados de la solución del problema de flujo de cargas en sistemas de 3, 6 y 9 barras, mediante el software de simulación DIgSILENT PowerFactory®, con el fin de realizar una comparación con los resultados obtenidos en UPSO. A continuación se presenta el planteamiento de la función objetivo a analizar, a partir del sistema de ecuaciones no lineales propias de cada sistema. El algoritmo se describe mediante un diagrama de flujo, mostrando sus etapas de funcionamiento, y fue implementado en la herramienta de software matemático MATLAB®. Sus resultados son compilados en gráficas y tablas obtenidas al realizar una serie de evaluaciones iterativas del proceso. Se realiza un análisis estadístico sobre el funcionamiento del método, como lo es tiempo promedio de una serie de iteraciones, la convergencia del método, la cantidad de iteraciones de acuerdo con el número de partículas evaluadas. Se presentan consideraciones encontradas de forma empírica para tener en cuenta sobre los parámetros propios del método y observaciones sobre el proceso realizado en los diferentes sistemas. Finalmente, se presentan las conclusiones planteadas a partir del estudio llevado a cabo.

* Este trabajo de grado hace parte integral de la línea de investigación del grupo CEMOS y aporta sus resultados con el objeto de consolidarla

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones.
Director: Carlos Rodrigo Correa Cely

ABSTRACT

TITLE

EVALUATION AND COMPARISON OF THE UPSO AND NEWTON RAPHSON METHODS ON THE ANALYSIS OF POWER FLOW ON ONE POWER SYSTEM*

AUTHORS

Camilo Andrés Pinzón Ceballos**
Edwin Ardila Pedraza**

KEYWORDS

UPSO, Newton-Raphson, DIgSILENT, Steady State Power Flow, Power Systems

The following document presents the application of the stochastic algorithm Unified Particle Swarm Optimization (UPSO) on typical examples of steady state power flow calculations on power systems. It starts with a brief explanation about the theoretical foundations of the method, and then it proceeds to describe the results of the power flow solution on systems consisting of 3, 6 y 9 buses using the simulation software DIgSILENT PowerFactory®, this was made with the purpose of make a comparison with the results obtained after with UPSO. Next it's presented the objective function approach to use from the non-linear equation systems proper for each power system. The algorithm is described in a flowchart, where the different steps are shown, and it was implemented in the mathematical software tool MATLAB®. The results are shown through graphics and tables obtained from a series of iterative processes. A statistical analysis is made about the method performance, like average time from a series of iterations, the method's convergence and the quantity of iterations according to the number of particles used. Some considerations found empirically are shown about the own parameters of the method and observations about the process made on the several systems. Finally, some conclusions obtained from the analysis are shown.

* This Degree Work is an integral part of the CEMOS group and provides its results with the purpose of consolidate it.

** Faculty of Physic-Mechanical Engineering. School of Electric, Electronic and Telecommunications Engineering. Project Director: Carlos Rodrigo Correa Cely.

TABLA DE CONTENIDO

	Pág.
INTRODUCCION	17
1. MARCO TEÓRICO.....	20
1.1 MÉTODO DE OPTIMIZACION POR ENJAMBRE DE PARTÍCULAS (PSO - PARTICLE SWARM OPTIMIZATION)	21
1.2 OPTIMIZACIÓN POR ENJAMBRE DE PARTÍCULAS UNIFICADO (UPSO - UNIFIED PARTICLE SWARM OPTIMIZATION)	23
1.3 FUNCIÓN OBJETIVO.....	26
1.4 FLUJO DE CARGAS EN SISTEMAS DE POTENCIA	26
2. FASE EXPERIMENTAL	29
2.1 PROGRAMACIÓN DEL ALGORITMO.....	29
2.2 PRUEBA DEL ALGORITMO FRENTE A PROBLEMAS TÍPICOS	31
2.2.1 Parábola tridimensional con simetría esférica.....	32
2.2.2 Ecuación de cuarto grado de Rosenbrock	32
2.2.3 Función escalón penta-dimensional:.....	35
2.2.4 Solución a circuito de 3 mallas:.....	36
2.2.5 Solución a circuito de 2 nodos de respuesta compleja:.....	38
2.3 FLUJO DE CARGAS EN SISTEMAS DE POTENCIA	40
2.4 COEFICIENTE DE UNIFICACIÓN u	43

2.5	PROBLEMA TIPO DE 3 BARRAS.....	43
2.6	PROBLEMA TIPO DE 6 BARRAS.....	46
2.7	PROBLEMA TIPO DE 9 BARRAS.....	50
3.	CONSIDERACIONES	54
4.	CONCLUSIONES - RECOMENDACIONES.....	55
	BIBLIOGRAFÍA.....	58
	ANEXO	59

LISTA DE TABLAS

	Pág.
Tabla 1. Clasificación general de las barras en un sistema de potencia.	27
Tabla 2. Datos de simulación sistema tipo de 3 barras	40
Tabla 3. Datos de simulación sistema tipo de 6 barras	41
Tabla 4. Datos de simulación sistema tipo de 9 barras	42
Tabla 5. Resultados UPSO problema tipo de 9 barras	52

LISTAS DE FIGURAS

	Pág.
Figura 1. Diagrama de flujo algoritmo PSO.....	23
Figura 2. Diagrama de flujo algoritmo UPSO	25
Figura 3. Matriz correspondiente a $Ybarra$	28
Figura 4. Captura de pantalla de la salida del programa.....	31
Figura 5. Iteraciones vs Intentos (Parábola)	33
Figura 6. Iteración vs Mejor posición global (Parábola)	33
Figura 7. Iteraciones vs Intentos (Rosenbrock).....	34
Figura 8. Iteración vs Mejor posición global (Rosenbrock).....	34
Figura 9. Iteración vs Mejor posición global 1 tomada de [8]	35
Figura 10. Iteración vs Mejor posición de 2.2.3.....	36
Figura 11. Circuito de 3 mallas.....	37
Figura 12. Iteración vs Mejor posición de 2.2.4.....	37
Figura 13. Circuito de 2 nodos de respuesta compleja	38
Figura 14. Iteración vs Mejor posición de 2.2.5.....	39
Figura 15. Sistema de potencia tipo de 3 barras.....	40
Figura 16. Sistema de potencia tipo de 6 barras.....	41
Figura 17. Sistema de potencia tipo de 9 barras.....	42
Figura 18. Iteraciones vs Factor de Unificación UPSO	43
Figura 19. Matriz $Ybarra$ problema tipo de 3 barras	43
Figura 20. Iteraciones vs partículas en UPSO sistema de 3 barras.....	45

Figura 21. Tiempo vs partículas en UPSO sistema de 3 barras.....	45
Figura 22. Acercamiento a la respuesta con UPSO en sistema de 3 barras	46
Figura 23. Matriz <i>Ybarra</i> problema tipo de 6 barras	46
Figura 24. Iteraciones vs partículas en UPSO sistema de 6 barras	48
Figura 25. Tiempo vs partículas en UPSO sistema de 6 barras.....	48
Figura 26. Acercamiento a la respuesta con UPSO en sistema de 6 barras	49
Figura 27. Matriz <i>Ybarra</i> problema tipo de 9 barras	50
Figura 28. Respuesta Newton-Raphson vs UPSO.....	51
Figura 29. Respuestas UPSO óptima vs no óptima	53
Figura 30. Respuesta 1 - UPSO sistema de 9 barras	60
Figura 31. Respuesta 2 - UPSO sistema de 9 barras	60
Figura 32. Respuesta 3 - UPSO sistema de 9 barras	61
Figura 33. Respuesta 4 - UPSO sistema de 9 barras	61
Figura 34. Respuesta 5 - UPSO sistema de 9 barras	62
Figura 35. Respuesta 6 - UPSO sistema de 9 barras	62
Figura 36. Respuesta 7 - UPSO sistema de 9 barras	63
Figura 37. Respuesta 8 - UPSO sistema de 9 barras	63
Figura 38. Respuesta 9 - UPSO sistema de 9 barras	64
Figura 39. Respuesta 10 - UPSO sistema de 9 barras	64
Figura 40. Respuesta 11 - UPSO sistema de 9 barras	65

INTRODUCCION

Los sistemas eléctricos comprenden una gran variedad de elementos conectados entre sí, iniciando en la generación, pasando por las redes de transmisión, distribución y finalizando en el usuario. Cada una de estas partes se puede modelar de manera sencilla, permitiendo simplificar el modelo en un sistema completo de gran tamaño. Estos sistemas se llevan a un conjunto de ecuaciones no lineales, en términos de la potencia inyectada en las barras.

Los cálculos de *Flujos de Carga*, o *Flujos de Potencia*, caracterizan el sistema eléctrico para unas condiciones carga, generación y tensión determinadas, y en condiciones de estado estable. Sin embargo, son importantes para el diseño, la planeación, la expansión y la operación de redes eléctricas; para realizar el análisis del rendimiento en régimen permanente del sistema; realizar análisis de estabilidad y análisis de contingencias, en caso de presentarse salidas inesperadas de algún (o algunos) elemento(s) del sistema.

En la actualidad se observa una tendencia a la investigación de algoritmos estocásticos, como lo es *Unified Particle Swarm Optimization (UPSO)* [1]. Este tipo de técnicas basan su desarrollo en la solución de una función objetivo que se obtiene a partir de la suma del cuadrado de las ecuaciones que forman el sistema. En este documento se hará uso del algoritmo UPSO para la solución de flujo de cargas, a título de ejemplo, en un sistema de potencia de nueve (9) barras.

El presente informe de investigación se compone de una introducción, seguido del planteamiento de los objetivos ya aprobados en el plan de trabajo de grado I. Posteriormente, en el capítulo 1 se describen sucintamente los fundamentos conceptuales referidos para el desarrollo y comprensión del tema en forma integral. En el capítulo 2, la fase experimental, se muestran los principales resultados obtenidos en la aplicación del programa desarrollado en problemas tipo

de optimización, junto con los resultados obtenidos al desarrollar ejemplos de 3, 6 y 9 barras dentro del cálculo de flujo de cargas. Finalmente se presentan algunas consideraciones y conclusiones obtenidas.

OBJETIVOS¹

GENERAL

Realizar un análisis comparativo para la solución de un problema de Flujo de carga en sistemas de potencia, entre el método numérico de Newton-Raphson y el método estocástico de Optimización por Enjambre de Partículas Unificado (UPSO - *Unified Particle Swarm Optimization*), desde el punto de vista de precisión en los resultados.

ESPECÍFICOS

1. Implementar UPSO en MATLAB® que sea capaz de solucionar sistemas de ecuaciones no lineales.
2. Contrastar los resultados de UPSO obtenidos de un problema tipo de 9 barras, con la herramienta de simulación DIGSILENT®.

¹ Esto constituye un aporte intelectual consensuado entre los estudiantes Pinzón, Ardila y el profesor Rodrigo Correa. Su formulación fue la base para el desarrollo del presente trabajo de grado.

1. MARCO TEÓRICO²

El problema de flujo de cargas consiste en conocer, dadas determinadas condiciones, qué tensiones presentan y qué potencias fluyen por las diferentes barras o nodos de un sistema de potencia específico, y es de notable importancia en la ingeniería eléctrica; existiendo múltiples opciones para su solución planteadas en la literatura [2], [3]. Recientemente se han desarrollado alternativas para diferentes problemas, como UPSO, que proponen un acercamiento a la solución desde el punto de vista estocástico y que han demostrado un buen desempeño en una gran variedad de situaciones en diferentes áreas [4–10]. Estas técnicas son apropiadas cuando la optimización del problema es complejo, no lineal o estocástico, y con un conocimiento mínimo del sistema [5].

Uno de los grandes méritos de los algoritmos estocásticos es la habilidad de inspeccionar en diferentes direcciones (o posibles soluciones) de manera simultánea, a diferencia de los métodos clásicos que encuentran la solución de forma secuencial, basados en la búsqueda en una sola dirección, lo que genera que, si se inicia en un punto relativamente lejos de la respuesta, el método diverja y se deba reiniciar el proceso.

Los algoritmos estocásticos no se centran en los comportamientos de los sistemas estudiados, ni utilizan información detallada del mismo, salvo el análisis previo de la matriz de admitancias, como por ejemplo, la continuidad o no de su función objetivo, sino que utiliza el azar para encontrar las soluciones modificando las obtenidas mediante ecuaciones de múltiples variables y evaluando si es la óptima o no. Antes de entrar a analizar el método Unified Particle Swarm Optimization, es necesario entrar en detalle al método que lo antecedió, Particle Swarm Optimization.

² Este texto fue escrito como aporte intelectual colaborativo de los estudiantes Pinzón, Ardila y el profesor Rodrigo Correa.

1.1 MÉTODO DE OPTIMIZACION POR ENJAMBRE DE PARTÍCULAS (PSO - PARTICLE SWARM OPTIMIZATION)

Es un método estocástico de optimización basado en poblaciones. Fue desarrollado a partir de la simulación de un modelo social simplificado. Los agentes fueron pensados como aves a prueba de colisión, y la idea original era "simular gráficamente la armoniosa pero impredecible coreografía de una bandada" [6]. En el documento de Kennedy y Eberhart [6] se presentan, tanto las consideraciones generales del método, como los pasos a través de los cuales se llegó desde el modelo de aves ya mencionado, al de partículas. Esta es una versión simplificada que no considera el tamaño de cada partícula sino únicamente su posición.

En su versión simplificada, el algoritmo consiste en: dada una población inicial, evaluarla en la función objetivo, determinando con esto los puntos que pueden ser considerados como la mejor posición (propia) que cada partícula presenta (p_i), y comparando entre éstas cuál pasa a ser la mejor posición global (p_{g_i}). El cambio de posición de cada agente se rige por una variable v_i , asociada a la velocidad de cada agente y que depende de su posición actual, de la mejor posición propia y de la mejor posición global, y que se modela de acuerdo con las siguientes ecuaciones [7]:

$$v_i^{(k+1)} = \mathcal{X} * [v_i^{(k)} + c_1 r_1 (p_i^{(k)} - x_i^{(k)}) + c_2 r_2 (p_{g_i}^{(k)} - x_i^{(k)})] \quad (1)$$

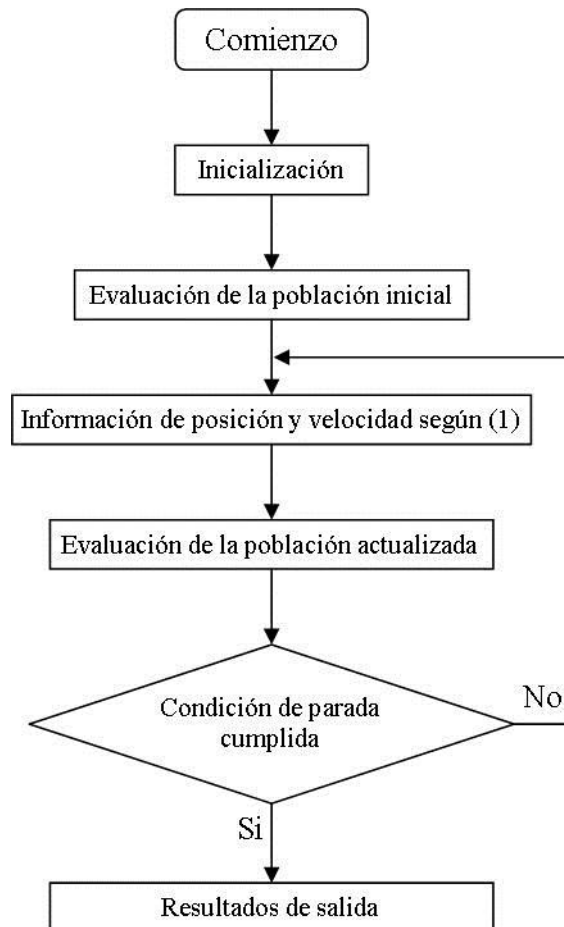
$$x_i^{(k+1)} = x_i^{(k)} + v_i^{(k+1)} \quad (2)$$

donde g_i es el índice de la partícula mejor posicionada en la vecindad de x_i ; $i = 1, 2, \dots, N$ número de partículas; k corresponde al contador de iteraciones; \mathcal{X} es un parámetro llamado *factor de restricción*, que controla la magnitud de la

velocidad; c_1 y c_2 son parámetros positivos de aceleración, llamados parámetro *cognitivo* y *social* respectivamente; y r_1 y r_2 son vectores aleatorios, de valores entre $[0,1]$ [7]. Por el significado de p_{g_i} en el algoritmo, se comprende el intercambio de información entre agentes. Para implementar este algoritmo se sigue el diagrama de flujo de la Figura 1, el cual está constituido por cuatro pasos descritos a continuación [5]:

- (i) *Inicialización*: Su objetivo es determinar el intervalo en el cual se encuentran las posiciones iniciales de las partículas. La población inicial se ubica de manera aleatoria en el dominio, la mejor posición p_i inicial, correspondiente al punto más cercano a la solución, es la posición actual. Se determinan también los parámetros de la ecuación (1).
- (ii) *Evaluación de la población inicial*: Esta etapa consiste en evaluar todas las partículas, utilizando para ello la función objetivo. De aquí se obtienen los valores de p_i y p_{g_i} para el cálculo de las velocidades; donde p_i es la mejor posición de cada partícula y p_{g_i} es la mejor posición entre todas.
- (iii) *Información de posición y velocidad*: Con la ecuación (1) se calcula la velocidad de cada partícula; para luego reubicar cada partícula en el espacio, con la ayuda de la ecuación (2).
- (iv) *Evaluación de la población actualizada*: Luego de reubicar las partículas, se evalúan de nuevo en la función objetivo para analizar su cercanía a la respuesta (como se observa en [8], la función objetivo evaluada en las partículas debe tender a $F_{obj} = 0$) de acuerdo con el criterio de parada establecido. Si se cumple, la respuesta final es p_{g_i} ; de lo contrario, se realiza de nuevo el paso (iii) hasta cumplir con el criterio o abortar el proceso.

Figura 1. Diagrama de flujo algoritmo PSO



Fuente: [5]

1.2 OPTIMIZACIÓN POR ENJAMBRE DE PARTÍCULAS UNIFICADO (UPSO - UNIFIED PARTICLE SWARM OPTIMIZATION)

El desempeño de un algoritmo basado en poblaciones, como lo es PSO, tiene dos variantes: la habilidad de realizar búsquedas globales, o *exploración*, y la habilidad de realizar búsquedas locales más refinadas, conocida como *explotación*. En UPSO se toman estas dos variantes y se enlazan incrementando así sus capacidades de búsquedas del óptimo global [7].

Partiendo de las ecuaciones (1) y (2) vistas en la sección 1.1 y tomando $G_i^{(k+1)}$ como la actualización de la velocidad de la partícula i -ésima, x_i , en la variante global de PSO, mientras que $L_i^{(k+1)}$ es la actualización de velocidad de la variante local, se tiene:

$$G_i^{(k+1)} = \mathcal{X} * [v_i^{(k)} + c_1 r_1 (p_i^{(k)} - x_i^{(k)}) + c_2 r_2 (p_g^{(k)} - x_i^{(k)})] \quad (3)$$

$$L_i^{(k+1)} = \mathcal{X} * [v_i^{(k)} + c_1 r_1' (p_i^{(k)} - x_i^{(k)}) + c_2 r_2' (p_{g_i}^{(k)} - x_i^{(k)})] \quad (4)$$

donde, k denota el número de la iteración, g es el índice de la mejor partícula (mejor posición variante global), y g_i es el índice de la mejor partícula local (mejor posición variante local). Combinando las ecuaciones (3) y (4) se obtienen las fórmulas de cambio de velocidad y posición respectivamente para UPSO:

$$U_i^{(k+1)} = u G_i^{(k+1)} + (1 - u) L_i^{(k+1)} \quad (5)$$

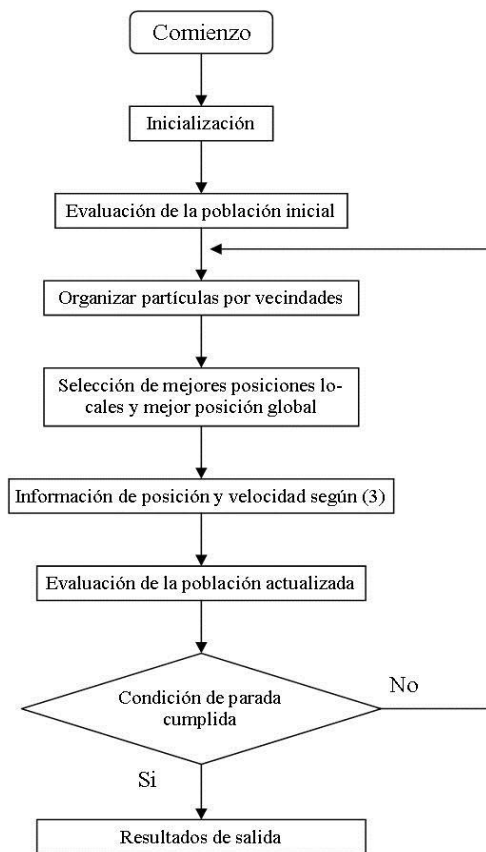
$$x_i^{(k+1)} = x_i^{(k)} + U_i^{(k+1)} \quad (6)$$

donde $u \in [0,1]$ es un parámetro llamado factor de unificación, que determina la influencia de los componentes de las variables local y global en la ecuación (5). Para $u = 1$, la ecuación (5) es igual a la variante global, mientras que para $u = 0$, es igual a la variante local. Para valores $[0,1]$, se obtienen variantes compuestas de PSO que combinan las características antes mencionadas de exploración y explotación de sus variantes.

La implementación de este algoritmo es similar a la de *PSO*, con la diferencia que ahora se establecerán vecindades de partículas a partir del índice de cada una. Es decir, para establecer una vecindad de, por ejemplo 10 partículas, se ordenan de acuerdo con su índice. Entonces, las partículas de índices $i =$

1,2,3,...,10 forman la primera vecindad, las de índice $i = 11,12,13,...,20$ forman la segunda vecindad y así sucesivamente. Luego, de cada vecindad se tomará la partícula con mejor posición (p_{gi}) y de estas posiciones la que esté más cercana a $F_{obj} = 0$ será la mejor posición global (p_g), para luego aplicar las ecuaciones (5) y (6). El factor de unificación (u) es propio de cada caso bajo estudio y se debe buscar el valor que presente una mejor eficiencia de manera empírica; pues aún no hay herramientas matemáticas que ayuden en su búsqueda.

Figura 2. Diagrama de flujo algoritmo UPSO



Fuente: Elaboración propia

1.3 FUNCIÓN OBJETIVO

Con el fin de determinar las condiciones de operación de cada sistema de potencia, el algoritmo se evaluará mediante la función objetivo. Esta función objetivo consiste en elevar al cuadrado las ecuaciones características del problema y además sumarlas entre sí, como se describe en [8]:

$$F_{obj}(x) = \sum_{i=1}^m (f_i(X))^2 \quad (7)$$

donde: $X = (x_1, x_2, \dots, x_n)$

Dicha función simboliza qué tan cerca están de la solución real los datos obtenidos por el método empleado. Para este caso, la función objetivo evaluará las posiciones de cada partícula. Las ecuaciones utilizadas para el sistema de potencia están dadas por $S = VI^*$, dado en valores en por unidad [pu], y aplicadas para la función objetivo en general son:

$$f_i(X) = V_i I_{ij}^* - S_i \quad (8)$$

donde: $I_{ij}^* = \sum (Y_{ij} V_j)^*$

Y_{ij} : término correspondiente a la Ybarra del sistema.

1.4 FLUJO DE CARGAS EN SISTEMAS DE POTENCIA

“Un flujo de cargas es el modo de expresar la solución de estado estable de una red, en el lenguaje convencional de los sistemas de potencia. Esta solución no difiere esencialmente de la correspondiente a cualquier otro tipo de red, excepto en ciertas limitaciones peculiares de los suministros de energía” [3].

Como se mencionó anteriormente, se modelarán los sistemas de potencia en términos de circuitos equivalentes. Así, en lugar de tener, por ejemplo, una línea de transmisión de una distancia dada, con uno (o más) conductor(es) definidos y con cierta disposición, se tendrá un modelo simplificado con una impedancia serie y un par de admitancias en paralelo. Los valores del sistema están en por unidad, pues así los datos hallados son más significativos con respecto al sistema.

Existen en general 4 tipos de barra [2], los cuales se muestran en la Tabla 1. Cada una de ellas tiene asociadas dos variables conocidas (✓) y dos desconocidas (?), de las cuatro posibles variables a trabajar. Las cuatro variables son: potencia activa inyectada (P), potencia reactiva inyectada (Q), tensión en la barra (V) y ángulo de ésta (δ). Por ejemplo, si se desea analizar una barra tipo generación, se conocerán la potencia activa inyectada y la magnitud de la tensión en ella, mientras que las incógnitas serán potencia reactiva y ángulo de potencia (o de operación).

Tabla 1. Clasificación general de las barras en un sistema de potencia.

Tipo	Código	P	Q	V	δ	# aproximado
Referencia	0	?	?	✓	✓	1
Carga	1	✓	✓	?	?	85%
Generadora	2	✓	?	✓	?	10%
V controlado	3	✓	✓	✓	?	5%

Fuente: [2]

El valor de la columna “# aproximado” es una estimación de en qué cantidad se presentan dichos tipos de barra en un sistema de potencia.

De acuerdo con lo visto hasta ahora, el problema de flujo de cargas en el análisis de sistemas de potencia consistirá en resolver un conjunto de igual número de ecuaciones no lineales e incógnitas. Su solución se llevará a cabo con la ayuda

del método presentado en la sección 1.2. Para esto es necesario conocer la matriz Y_{barra} de cada sistema, la cual está dada por:

Figura 3. Matriz correspondiente a Y_{barra}

$$Y_{barra} = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & \cdots & Y_{1n} \\ Y_{21} & Y_{22} & Y_{23} & \cdots & Y_{2n} \\ Y_{31} & Y_{32} & Y_{33} & \cdots & Y_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{m1} & Y_{m2} & Y_{m3} & \cdots & Y_{mn} \end{bmatrix}$$

Fuente: Elaboración propia

donde Y_{ii} es la suma de las admitancias conectadas a la barra i y los términos restantes de la matriz el negativo de la suma de las admitancias compartidas entre las diferentes barras.

2. FASE EXPERIMENTAL

La metodología de desarrollo de este trabajo se realizó en tres etapas.

2.1 PROGRAMACIÓN DEL ALGORITMO

El algoritmo fue escrito en el lenguaje propio del software MATLAB 7.4 de MathWorks, disponible en las instalaciones del CENTIC de la UIS. Consiste de dos scripts, donde uno define la función objetivo a partir de las ecuaciones características de cada situación bajo estudio, y el otro llama dicha función objetivo para evaluarla en las partículas que conforman el enjambre.

La función objetivo, como se plantea en la ecuación (7) de la sección 1.3, se fundamenta en sumar las diferentes ecuaciones del sistema, luego de haberlas elevado al cuadrado. Esto se logra planteando un vector columna, donde cada elemento corresponde a una ecuación del sistema; luego se hace un producto punto entre el vector y él mismo, ya que: $F \cdot F = |F||F| = |F|^2 = (F_1, F_2, \dots, F_n)^2 = F_1^2 + F_2^2 + \dots + F_n^2$.

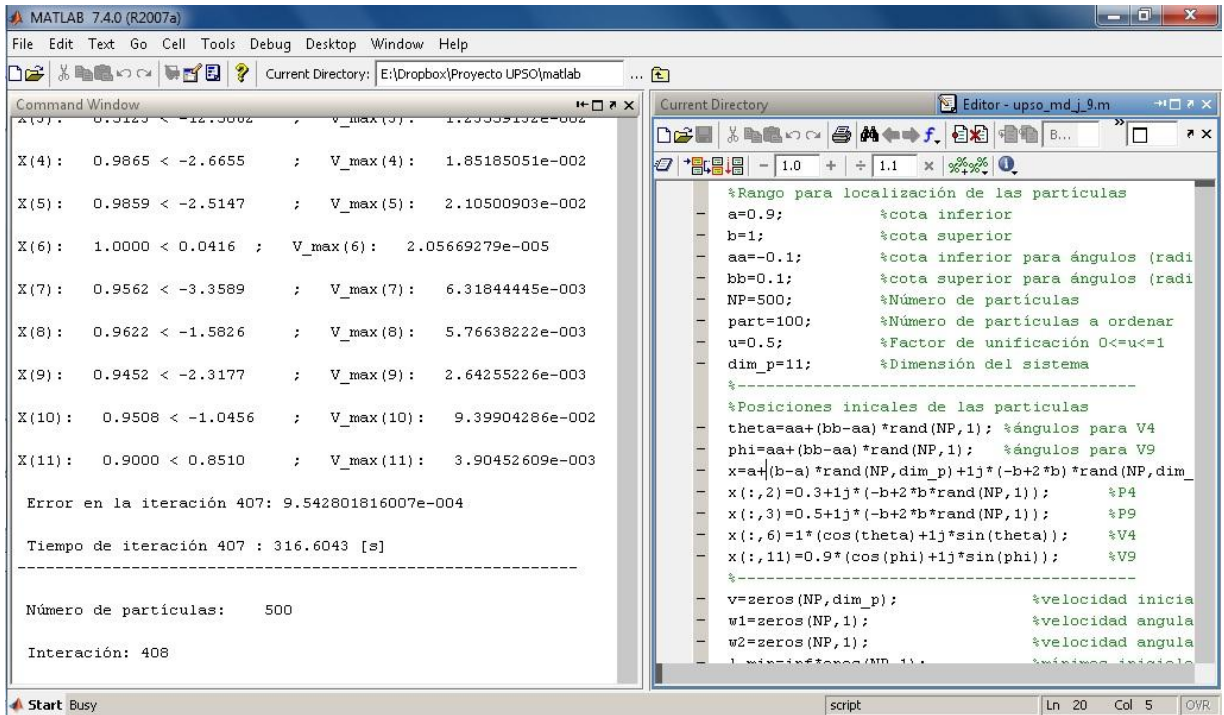
El script principal del algoritmo tiene tres funciones centrales. La primera consiste en generar partículas con valores de posición al azar en un intervalo determinado. Esto se agrupa en una matriz de NP filas por dim columnas, donde NP corresponde al número de partículas a utilizar, y dim a la dimensión o número de incógnitas del problema a resolver. Con esto, si se tiene un problema de seis incógnitas, se genera una matriz de, por ejemplo, 500×6 para un enjambre de 500 partículas de 6 coordenadas en el espacio. La segunda función se basa en evaluar cada partícula en la función objetivo mencionada en el párrafo anterior, obteniendo de esto un valor escalar. La solución al sistema la da la partícula que al ser evaluada de un valor cercano a cero, dentro de un margen de tolerancia. La

precisión trabajada en este experimento es de $1 * 10^{-9}$. La tercera función del script consta en identificar la mejor posición en la variante global y las mejores posiciones en las variantes locales, para aplicar la ecuación (5) y calcular las velocidades de cada partícula. Con esas velocidades se determinan las nuevas posiciones de las partículas. Seguido esto, se comprueba el criterio de parada. Si no se satisface dicho criterio, se reinicia el proceso con las nuevas posiciones.

Para conocer las mejores posiciones en las variantes locales, es necesario dividir las partículas en vecindades, lo que se logra creando una matriz de un número determinado de filas (conocido en el script como el parámetro "*part*"). Dicha matriz tendrá un número de columnas correspondiente al número de incógnitas del sistema, más una columna correspondiente a la función evaluada en cada fila, esto multiplicado por el número de vecindades ("*NP/part*" en el script). Así, para un sistema de seis incógnitas, con un enjambre de 500 partículas dividido en vecindades de 10, se tendrá una matriz resultante de 10 filas y $(6 + 1) * (500/10) = 350$ columnas. Luego, se examinan las menores funciones de cada grupo o vecindad y las coordenadas de la partícula correspondiente a dicho valor pasa a ser la mejor posición local de esa vecindad.

Con el fin de conocer el progreso del algoritmo, en cada iteración se imprimen los resultados, el número de iteraciones, el error de cada iteración, los valores de la mejor posición global y el tiempo; como se ilustra en la siguiente captura de pantalla:

Figura 4. Captura de pantalla de la salida del programa



Fuente: Elaboración propia.

2.2 PRUEBA DEL ALGORITMO FRENTE A PROBLEMAS TÍPICOS

Para probar la efectividad del algoritmo se examinaron dos tipos de problemas: por una parte, problemas clásicos de optimización, donde se tomaron unos ejemplos típicos [9], y por otro lado se tomaron ejercicios de apuntes realizados en clases de la asignatura Sistemas de Potencia I a cargo del profesor Gilberto Carrillo, para hallar la solución a sistemas de ecuaciones lineales con respuesta real y compleja, con el fin de evaluar la capacidad del algoritmo para resolver estas dos situaciones.

2.2.1 Parábola tridimensional con simetría esférica

Esta función está definida por la ecuación:

$$F(X) = \sum_{i=1}^3 X_i^2 \quad (9)$$

donde X_1, X_2 y X_3 representan coordenadas en cada una de las tres dimensiones. Dicha función presenta mínimo en $F(0,0,0) = 0$. Para fines de experimentación de convergencia, se realizaron veinte intentos; inicializando las partículas en el mismo intervalo utilizado en [9] ($-5.12 \leq X_i \leq 5.12$). En la Figura 6 se observa el movimiento de las partículas a través de las iteraciones.

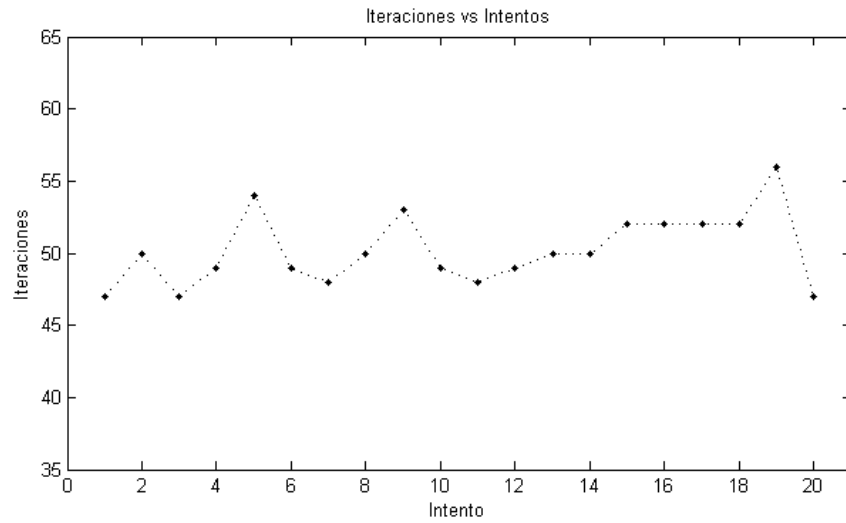
2.2.2 Ecuación de cuarto grado de Rosenbrock

Esta función está definida por la ecuación:

$$F(X) = 100 * (X_1^2 - X_2)^2 + (1 - X_1)^2 \quad (10)$$

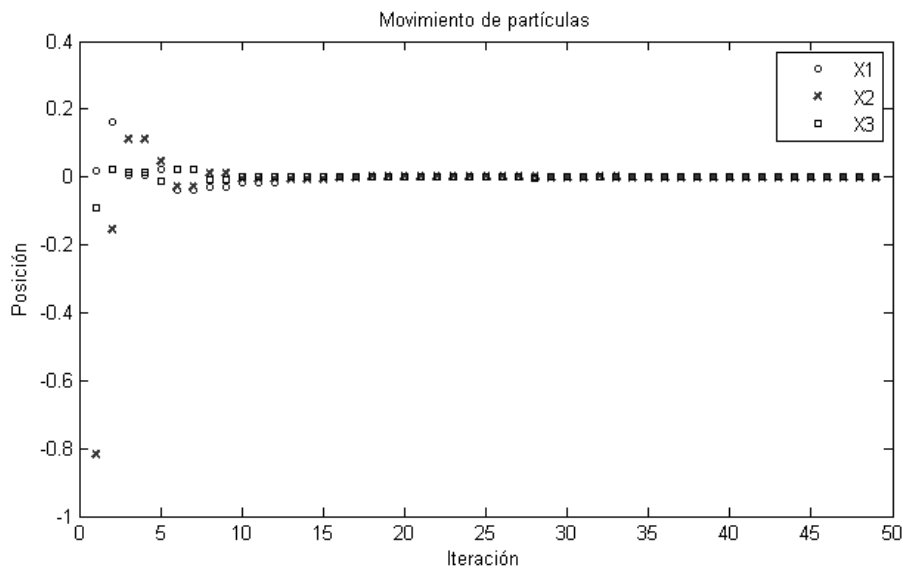
La ecuación de Rosenbrock es de cuarto grado, unimodal, de dos dimensiones, que presenta un mínimo en el punto $F(1,1) = 0$. Al igual que en el caso anterior, se realizaron 20 intentos para examinar la convergencia del método en este problema típico, teniendo en cuenta que el intervalo de inicio ahora es $-2.048 \leq X_i \leq 2.048$, y los resultados se encuentran en la Figura 7. En la Figura 8 se observa el movimiento de las partículas conforme pasan las iteraciones.

Figura 5. Iteraciones vs Intentos (Parábola)



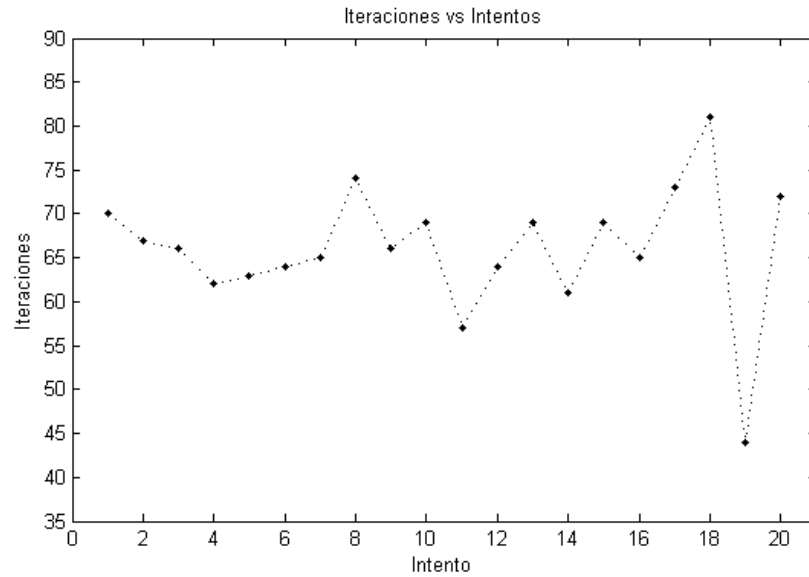
Fuente: Elaboración propia.

Figura 6. Iteración vs Mejor posición global (Parábola)



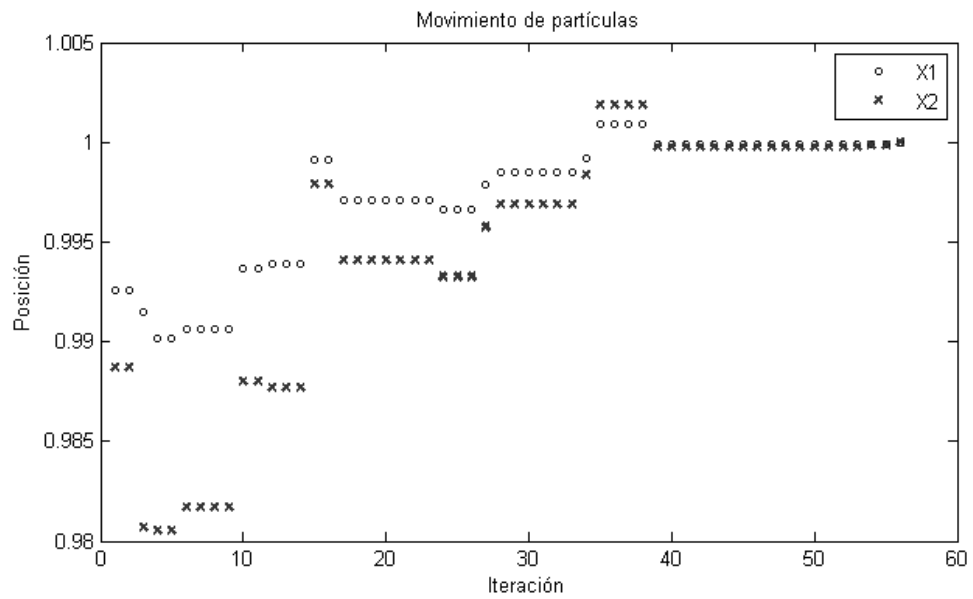
Fuente: Elaboración propia.

Figura 7. Iteraciones vs Intentos (Rosenbrock)



Fuente: Elaboración propia.

Figura 8. Iteración vs Mejor posición global (Rosenbrock)



Fuente: Elaboración propia.

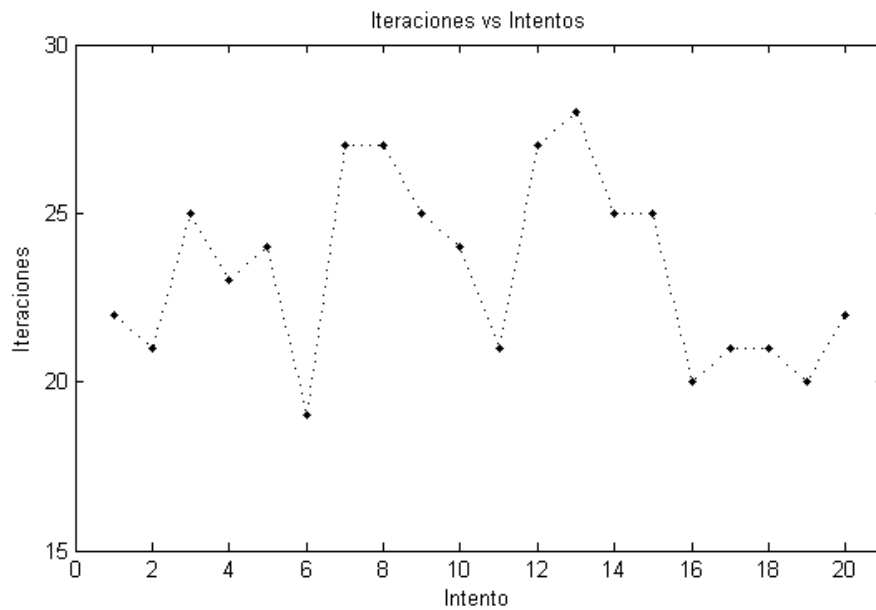
2.2.3 Función escalón penta-dimensional:

Esta función está definida por la ecuación:

$$F(X) = \sum_{i=1}^5 \llbracket X_i \rrbracket \quad (11)$$

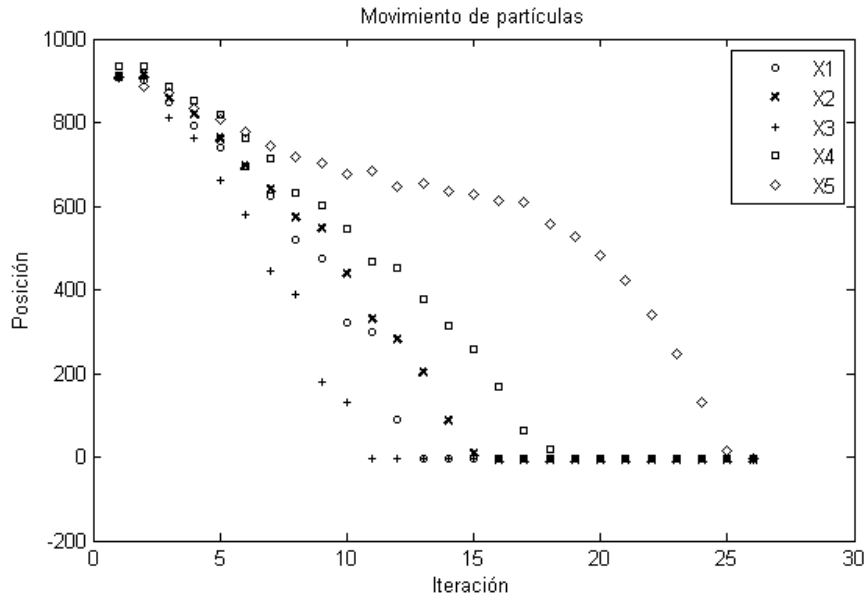
y representa un sistema de escalones de cinco dimensiones, donde se acota el problema al intervalo $X_i = [-5.12, 5.12]$. En las siguientes gráficas se muestra la solución y su tendencia hacia la respuesta. Se toma como un problema de minimización, donde la respuesta se encuentra en $F(-5.12, -5.12, -5.12, -5.12, -5.12) = -30$.

Figura 9. Iteración vs Mejor posición global 1 tomada de [8]



Fuente: Elaboración propia.

Figura 10. Iteración vs Mejor posición de 0



Fuente: Elaboración propia.

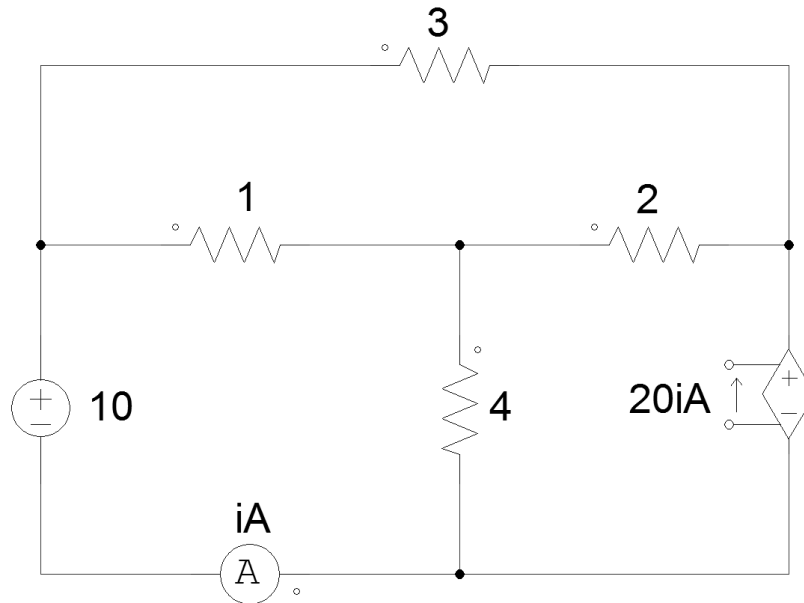
2.2.4 Solución a circuito de 3 mallas:

El circuito está configurado como se muestra en la Figura 11 y se define por las ecuaciones obtenidas a través del análisis de mallas:

$$\begin{aligned}
 F_1(X) &= 5X(1) - X(2) - 4X(3) - 10 \\
 F_2(X) &= -X(1) + 6X(2) - 2X(3) \\
 F_3(X) &= -6X(1) - 2X(2) + 6X(3)
 \end{aligned}
 \tag{12}$$

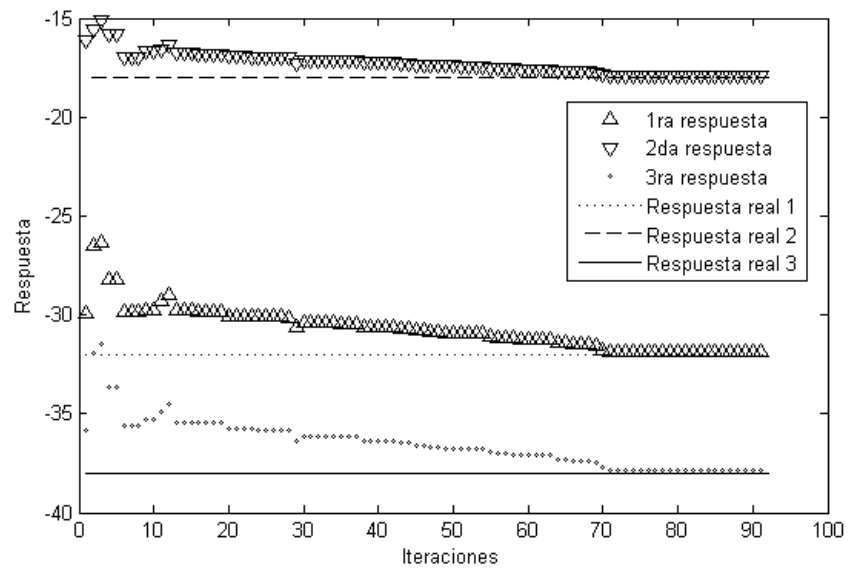
lo que plantea un sistema de tres ecuaciones con tres incógnitas, que se puede resolver por métodos convencionales de circuitos y que tiene como solución $X = [-32, -18, -38]$. Este circuito, al igual que los problemas planteados anteriormente, tiene solución únicamente real, pero se aproxima a los modelos de sistemas de potencia que se evaluarán a continuación.

Figura 11. Circuito de 3 mallas



Fuente: Apuntes de clase.

Figura 12. Iteración vs Mejor posición de 2.2.4



Fuente: Elaboración propia.

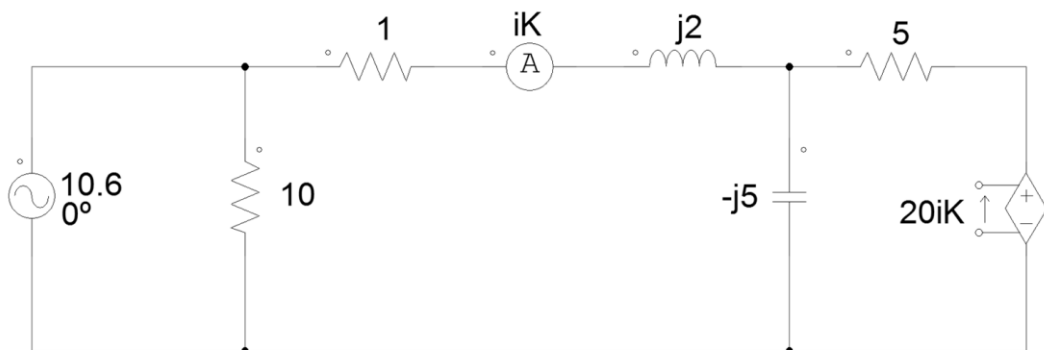
2.2.5 Solución a circuito de 2 nodos de respuesta compleja:

El circuito está configurado como se muestra en la figura y se define por las ecuaciones obtenidas a través del análisis de nodos:

$$\begin{aligned} F_1(X) &= \left(\frac{1}{10} + \frac{1}{1+j2}\right)X(1) + \left(\frac{-1}{1+j2}\right)X(2) - 10.6 \\ F_2(X) &= \left(\frac{-5}{1+j2}\right)X(1) + \left(\frac{5}{1+j2} - \frac{1}{j5} + \frac{1}{5}\right)X(2) \end{aligned} \quad (13)$$

Entonces, se plantea un sistema de dos ecuaciones con dos incógnitas, que se puede resolver por métodos convencionales de circuitos y que tiene como solución $X = [68.4 - j16.8, 68 - j26]$. Este circuito, a diferencia de los problemas planteados anteriormente, tiene solución compleja, luego la gráfica del movimiento de las partículas a través de las iteraciones se representa como parte real y parte imaginaria (ver Figura 14).

Figura 13. Circuito de 2 nodos de respuesta compleja



Fuente: Apuntes de clase.

2.3 FLUJO DE CARGAS EN SISTEMAS DE POTENCIA

El objetivo principal del análisis de flujo de cargas en un sistema de potencia es conocer cómo la potencia fluye a través de él, en estado estable. Este análisis se realiza a través del método de Newton-Raphson, el cual está implementado en la herramienta PowerFactory de la compañía DlgSILENT®. Los sistemas tipo propuestos para el análisis del funcionamiento de UPSO fueron simulados en dicha herramienta, para conocer la solución al problema de flujo de cargas y se muestran a continuación.

Figura 15. Sistema de potencia tipo de 3 barras

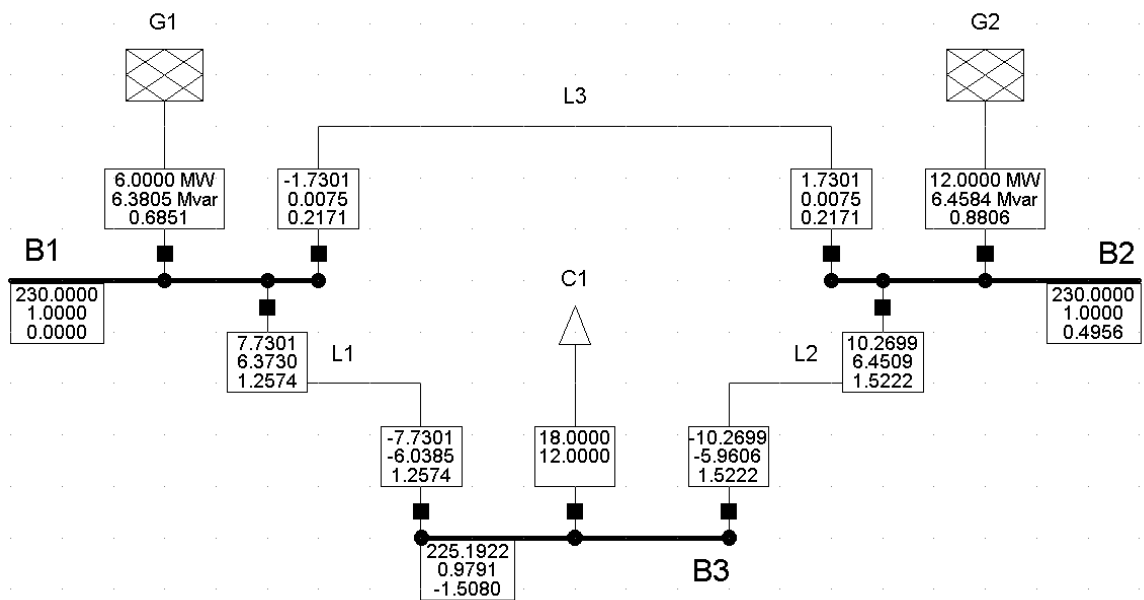


Tabla 2. Datos de simulación sistema tipo de 3 barras

Descripción	Tipo de Barra	V [pu]	δ [°]	P [MW]	Q [MVar]	P (pu)	Q (pu)
B1	Referencia	1,0000	0,0000	6,0000	6,3805	0,1000	0,1063
B2	PV	1,0000	0,4956	12,0000	6,4584	0,2000	0,1076
B3	Carga	0,9791	-1,5080	-18,0000	-12,0000	-0,3000	-0,2000

Fuente: Elaboración propia en software PowerFactory de DlgSILENT

Figura 16. Sistema de potencia tipo de 6 barras

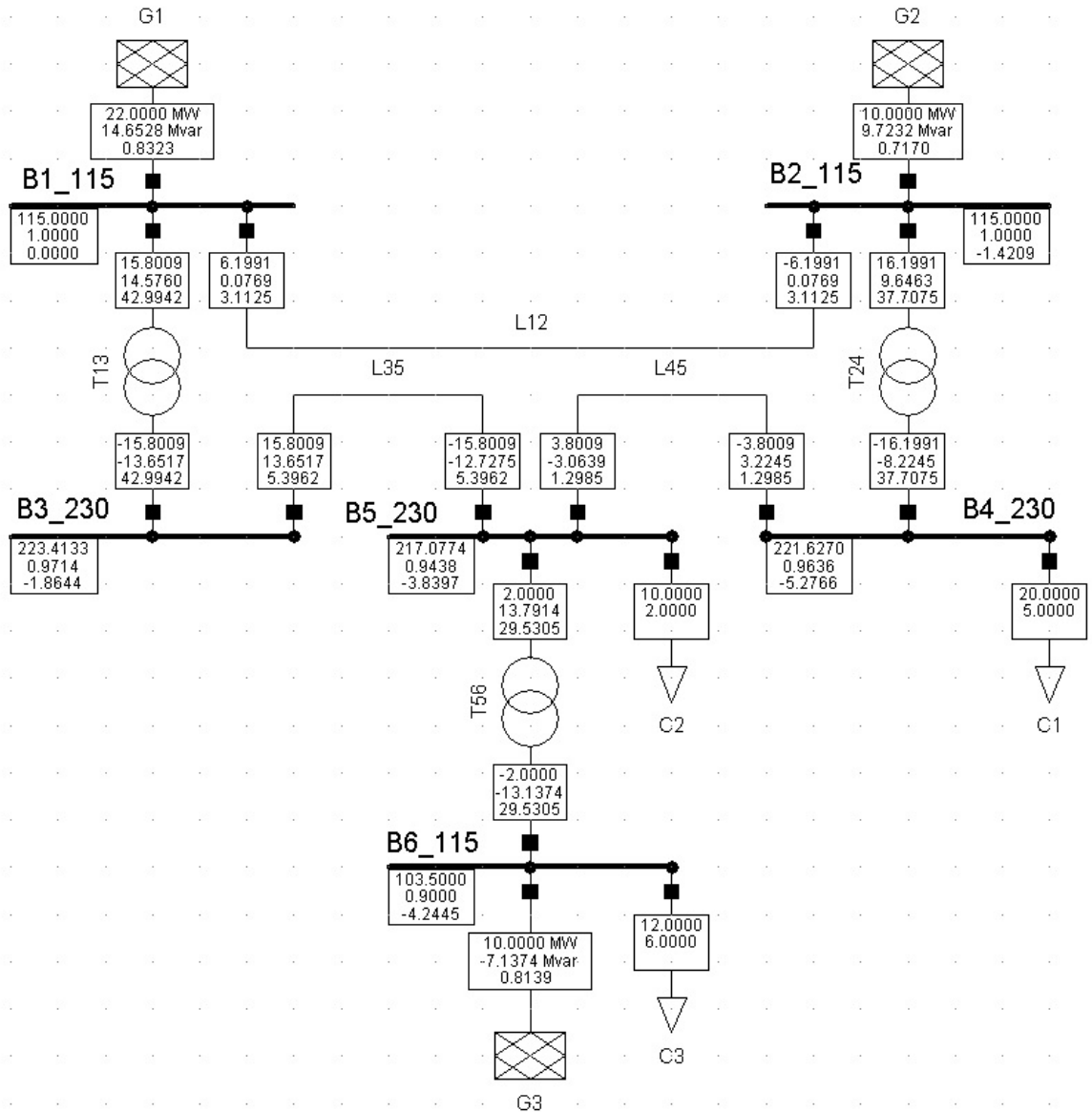


Tabla 3. Datos de simulación sistema tipo de 6 barras

Descripción	Tipo de Barra	V [pu]	δ [°]	P [MW]	Q [MVar]	P (pu)	Q (pu)
B1_115	Referencia	1,0000	0,0000	22,0000	14,6528	0,4400	0,2931
B2_115	PV	1,0000	-1,4209	10,0000	9,7232	0,2000	0,1945
B3_230	Carga	0,9714	-1,8644	0,0000	0,0000	0,0000	0,0000
B4_230	Carga	0,9636	-5,2766	-20,0000	-5,0000	-0,4000	-0,1000
B5_230	Carga	0,9438	-3,8397	-10,0000	-2,0000	-0,2000	-0,0400
B6_115	PV	0,9000	-4,2445	-2,0000	-13,1374	-0,0400	-0,2627

Fuente: Elaboración propia en software PowerFactory de DlgSILENT

Figura 17. Sistema de potencia tipo de 9 barras

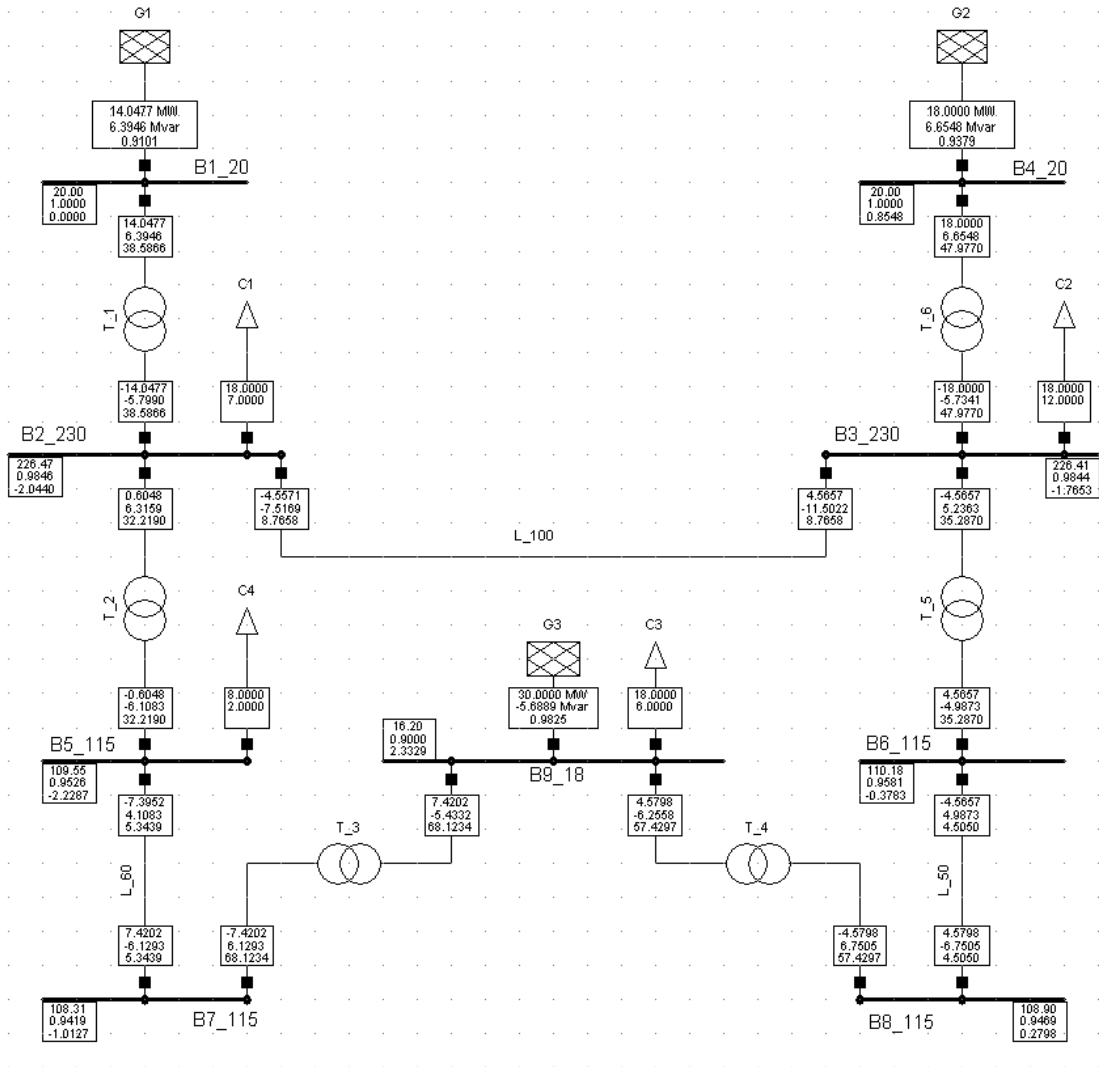


Tabla 4. Datos de simulación sistema tipo de 9 barras

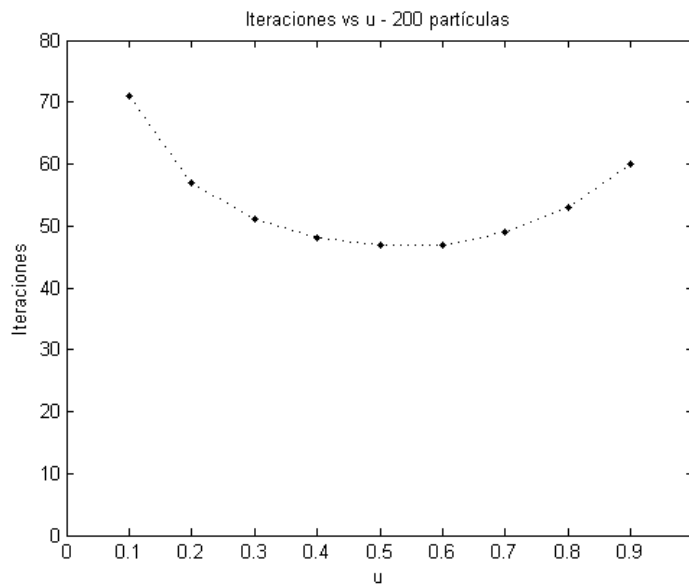
Descripción	Tipo de Barra	V [pu]	δ [°]	P [MW]	Q [MVar]	P (pu)	Q (pu)
B1_20	Referencia	1,0000	0,0000	14,0477	6,3946	0,2341	0,1066
B2_230	Carga	0,9846	-2,0440	-18,0000	-7,0000	-0,3000	-0,1167
B3_230	Carga	0,9844	-1,7653	-18,0000	-12,0000	-0,3000	-0,2000
B4_20	PV	1,0000	0,8548	18,0000	6,6546	0,3000	0,1109
B5_115	Carga	0,9526	-2,2287	-8,0000	-2,0000	-0,1333	-0,0333
B6_115	Carga	0,9581	-0,3783	0,0000	0,0000	0,0000	0,0000
B7_115	Carga	0,9419	-1,0127	0,0000	0,0000	0,0000	0,0000
B8_115	Carga	0,9469	0,2798	0,0000	0,0000	0,0000	0,0000
B9_18	PV	0,9000	2,3329	12,0000	-11,6889	0,2000	-0,1948

Fuente: Elaboración propia en software PowerFactory de DigSILENT

2.4 COEFICIENTE DE UNIFICACIÓN u

Al cambiar los valores de u se obtiene diferente rendimiento. Entonces para hacer más eficiente el algoritmo, se hacen pruebas en el sistema de 3 barras, manteniendo constante el número de partículas en $NP = 200$ y modificando el factor de unificación u , obteniendo así los resultados mostrados en la Figura 18:

Figura 18. Iteraciones vs Factor de Unificación UPSO



Fuente: Elaboración propia.

2.5 PROBLEMA TIPO DE 3 BARRAS

Figura 19. Matriz $Y_{barrera}$ problema tipo de 3 barras

$$Y_{barrera} = \begin{bmatrix} -25i/3 & 10i/3 & 5i \\ 10i/3 & -25i/3 & 5i \\ 5i & 5i & -10i \end{bmatrix}$$

Fuente: Elaboración propia

Las ecuaciones características del sistema de potencia son:

$$S_1 = V_1 * (Y_{11}V_1 + Y_{12}V_2 + Y_{13}V_3)^*$$

$$S_2 = V_2 * (Y_{21}V_1 + Y_{22}V_2 + Y_{23}V_3)^*$$

$$S_3 = V_3 * (Y_{31}V_1 + Y_{32}V_2 + Y_{33}V_3)^*$$

Adaptando estas ecuaciones a la función objetivo se tiene:

$$f_1 = V_1 * (Y_{11}V_1 + Y_{12}V_2 + Y_{13}V_3)^* - S_1$$

$$f_2 = V_2 * (Y_{21}V_1 + Y_{22}V_2 + Y_{23}V_3)^* - S_2$$

$$f_3 = V_3 * (Y_{31}V_1 + Y_{32}V_2 + Y_{33}V_3)^* - S_3$$

Reemplazando los datos del sistema:

$$f_1 = 1\angle 0 * (-25i/3 + 10i/3 i * x(3) + 5i * x(4))^* - x(1)$$

$$f_2 = x(3) * (10i/3 - 25i/3 * x(3) + 5i * x(4))^* - x(2)$$

$$f_3 = x(4) * (5i + 5i * x(3) - 10i * x(4))^* - (-0.3 - 0.2i)$$

donde:

$x(1) = S_1$: Potencia compleja barra 1.

$x(2) = S_2$: Potencia compleja barra 2, potencia activa constante en 0.2.

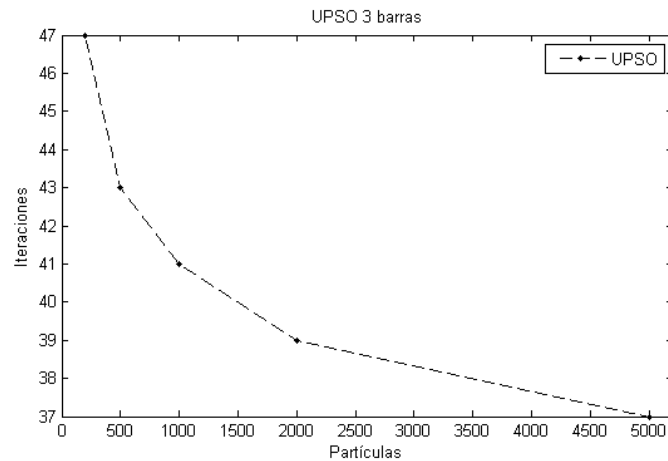
$x(3) = V_2\angle\delta_2$: Tensión en la barra 2, magnitud constante en 1.

$x(4) = V_3\angle\delta_3$: Tensión en la barra 3.

Aunque a primera vista parece un problema con infinitas soluciones, pues tiene 3 ecuaciones y 4 incógnitas, el sistema condiciona dichas incógnitas al mantener, para este caso, la potencia activa de la barra 2 en 0.2 y la magnitud de la tensión de la barra 2 en 1. Luego el problema pasa a tener una sola solución.

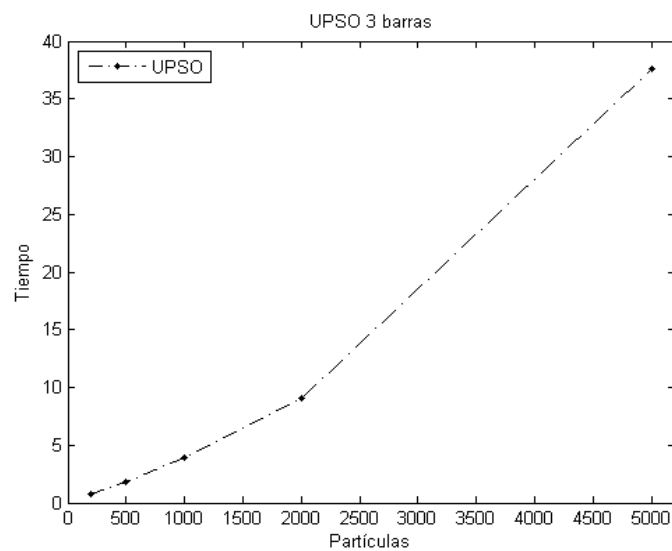
Con el fin de examinar el desempeño del algoritmo, se realizan pruebas con diferentes números de partículas y con un margen de error menor a $1 * 10^{-9}$; presentando a continuación las gráficas de iteraciones vs partículas y tiempo vs partículas:

Figura 20. Iteraciones vs partículas en UPSO sistema de 3 barras



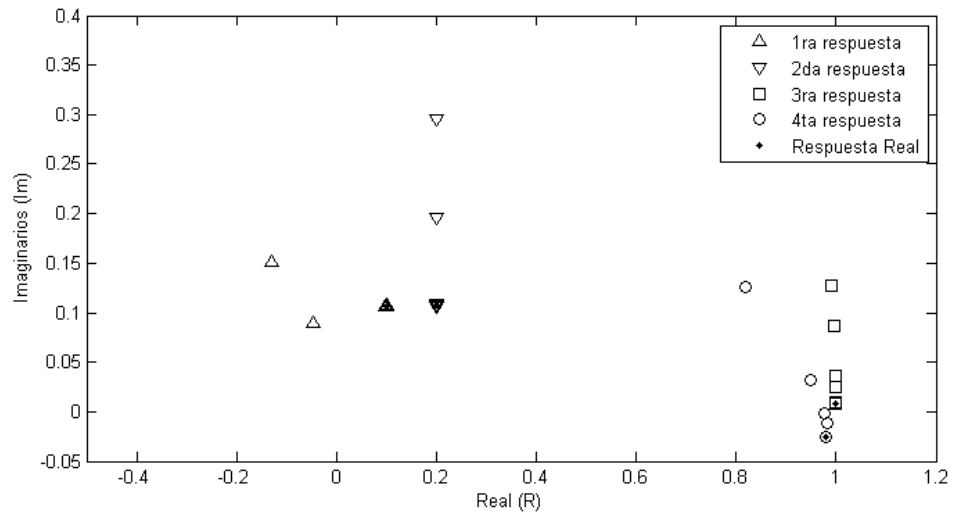
Fuente: Elaboración propia.

Figura 21. Tiempo vs partículas en UPSO sistema de 3 barras



Fuente: Elaboración propia.

Figura 22. Acercamiento a la respuesta con UPSO en sistema de 3 barras



$$x(1) = 0.099992576937988 + 0.106351978033160i$$

$$x(2) = 0.2 + 0.107642290765792i$$

$$x(3) = 1.0000 \angle 0.4958480844$$

$$x(4) = 0.9790931678 \angle -1.507896123$$

Fuente: Elaboración propia.

2.6 PROBLEMA TIPO DE 6 BARRAS

Figura 23. Matriz Y_{barra} problema tipo de 6 barras

$$Y_{barra} = \begin{bmatrix} -15i & 5i & 10i & 0 & 0 & 0 \\ 5i & -10i & 0 & 5i & 0 & 0 \\ 10i & 0 & -20i & 0 & 10i & 0 \\ 0 & 5i & 0 & -25i/3 & 10i/3 & 0 \\ 0 & 0 & 10i & 10i/3 & -20i & 20i/3 \\ 0 & 0 & 0 & 0 & 20i/3 & -20i/3 \end{bmatrix}$$

Fuente: Elaboración propia

La función objetivo está compuesta por:

$$f_1 = V_1 * (Y_{11}V_1 + Y_{12}V_2 + Y_{13}V_3 + Y_{14}V_4 + Y_{15}V_5 + Y_{16}V_6)^* - S_1$$

$$f_2 = V_2 * (Y_{21}V_1 + Y_{22}V_2 + Y_{23}V_3 + Y_{24}V_4 + Y_{25}V_5 + Y_{26}V_6)^* - S_2$$

$$f_3 = V_3 * (Y_{31}V_1 + Y_{32}V_2 + Y_{33}V_3 + Y_{34}V_4 + Y_{35}V_5 + Y_{36}V_6)^* - S_3$$

$$f_4 = V_4 * (Y_{41}V_1 + Y_{42}V_2 + Y_{43}V_3 + Y_{44}V_4 + Y_{45}V_5 + Y_{46}V_6)^* - S_4$$

$$f_5 = V_5 * (Y_{51}V_1 + Y_{52}V_2 + Y_{53}V_3 + Y_{54}V_4 + Y_{55}V_5 + Y_{56}V_6)^* - S_5$$

$$f_6 = V_6 * (Y_{61}V_1 + Y_{62}V_2 + Y_{63}V_3 + Y_{64}V_4 + Y_{65}V_5 + Y_{66}V_6)^* - S_6$$

Reemplazando los datos del sistema:

$$f_1 = 1\angle 0 * (-15i + 5i * x(4) + 10i * x(5))^* - x(1)$$

$$f_2 = x(4) * (5i - 10i * x(4) + 5i * x(6))^* - x(2)$$

$$f_3 = x(5) * (10i - 20i * x(5) - 20i * x(7))^*$$

$$f_4 = x(6) * (5i * x(4) - 25i/3 * x(6) + 10i/3 * x(7))^* - (-0.4 - 0.1i)$$

$$f_5 = x(7) * (10i * x(5) + 10i/3 * x(6) - 20i * x(7) + 20i/3 * x(8))^* - (0.2 - 0.04i)$$

$$f_6 = x(8) * (20i/3 * x(7) - 20i/3 * x(8))^* - x(3)$$

donde:

$x(1) = S_1$: Potencia compleja barra 1.

$x(2) = S_2$: Potencia compleja barra 2, potencia activa constante en 0.2.

$x(3) = S_6$: Potencia compleja barra 6, potencia activa constante en 0.2.

$x(4) = V_2\angle\delta_2$: Tensión en la barra 2, magnitud constante en 1.

$x(5) = V_3\angle\delta_3$: Tensión en la barra 3.

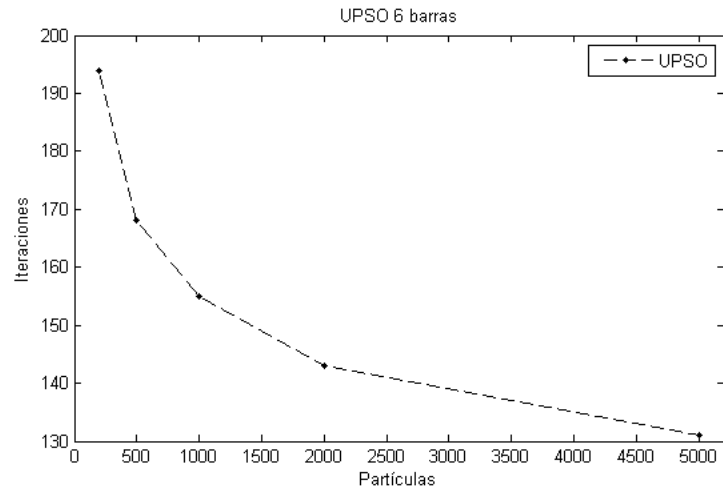
$x(6) = V_4\angle\delta_4$: Tensión en la barra 4.

$x(7) = V_5\angle\delta_5$: Tensión en la barra 5.

$x(8) = V_6\angle\delta_6$: Tensión en la barra 6, magnitud constante en 0.9.

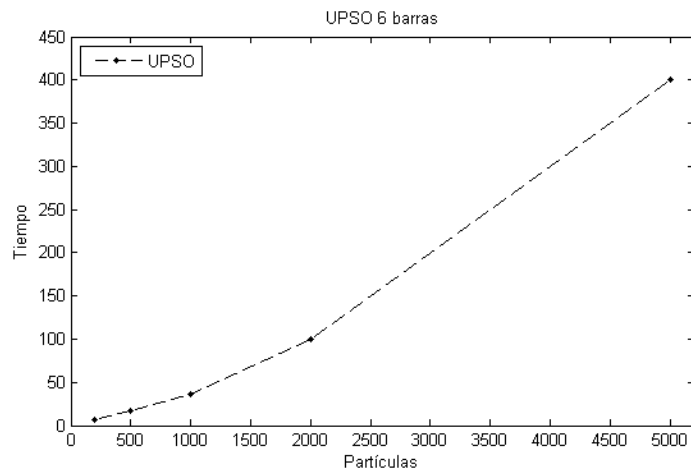
Al igual que el problema anterior, se realizaron pruebas con diferentes números de partículas, obteniendo los siguientes resultados:

Figura 24. Iteraciones vs partículas en UPSO sistema de 6 barras



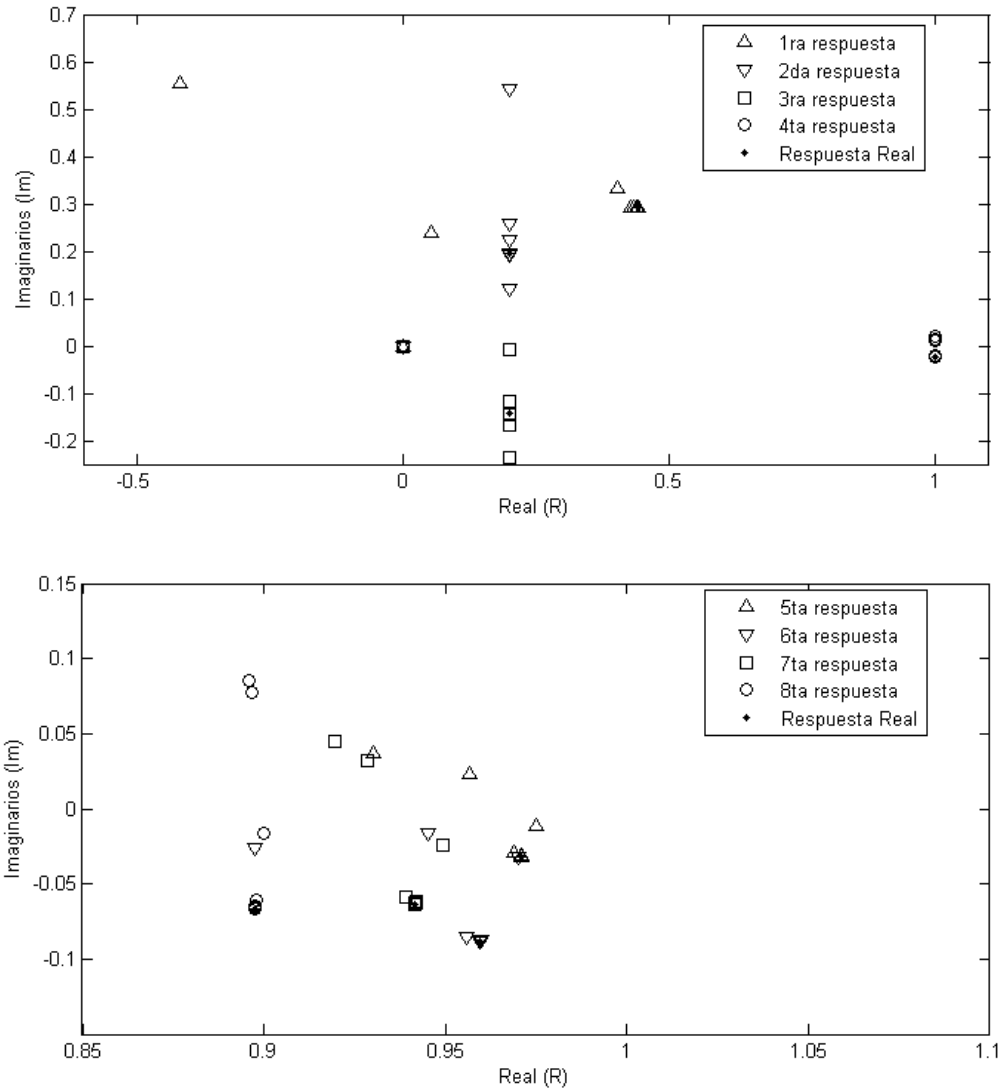
Fuente: Elaboración propia.

Figura 25. Tiempo vs partículas en UPSO sistema de 6 barras



Fuente: Elaboración propia.

Figura 26. Acercamiento a la respuesta con UPSO en sistema de 6 barras



$$x(1) = 0.440061814331108 + 0.293075959613150i$$

$$x(2) = 0.2 + 0.194471779526480i$$

$$x(3) = 0.2 - 0.142739567343866i$$

$$x(4) = 1 \angle -1.421126124^\circ$$

$$x(5) = 0.9713612957 \angle -1.86459521^\circ$$

$$x(6) = 0.9635948162 \angle -5.277020612^\circ$$

$$x(7) = 0.9438135985 \angle -3.840159615^\circ$$

$$x(8) = 0.9 \angle -4.245024086^\circ$$

Fuente: Elaboración propia.

2.7 PROBLEMA TIPO DE 9 BARRAS

Figura 27. Matriz Y_{barra} problema tipo de 9 barras

$-20i/3$	$20i/3$	0	0	0	0	0	0	0
$20i/3$	6.1976 $-26.2798i$	-6.1976 $+16.4435i$	0	$10i/3$	0	0	0	0
0	-6.1976 $+16.4435i$	6.1976 $-26.2798i$	$20i/3$	0	$10i/3$	0	0	0
0	0	$20i/3$	$-20i/3$	0	0	0	0	0
0	$10i/3$	0	0	0.8045 $-10.2267i$	0	-0.8045 $+6.9141i$	0	0
0	0	$10i/3$	0	0	0.9659 $-11.614i$	0	-0.9659 $+8.2979i$	0
0	0	0	0	-0.8045 $+6.9141i$	0	0.8045 $-9.3933i$	0	$2.5i$
0	0	0	0	0	-0.9656 $+8.2979i$	0	0.9659 $-10.787i$	$2.5i$
0	0	0	0	0	0	$2.5i$	$2.5i$	$-5i$

Fuente: Elaboración propia

Reemplazando los datos del sistema con 11 incógnitas:

$$f_1 = 1 \angle 0 * (Y_{11} + Y_{12} * x(4))^* - x(1)$$

$$f_2 = x(4) * (Y_{21} + Y_{22} * x(4) + Y_{23} * x(5) + Y_{25} * x(7))^* - (-0.3 - 0.1167i)$$

$$f_3 = x(5) * (Y_{32} * x(4) + Y_{33} * x(5) + Y_{34} * x(6) + Y_{36} * x(8))^* - (-0.3 - 0.2i)$$

$$f_4 = x(6) * (Y_{43} * x(5) + Y_{44} * x(6))^* - x(2)$$

$$f_5 = x(7) * (Y_{52}x(4) + Y_{55}x(7) + Y_{57}x(9))^* - (-0.1333 - 0.0333i)$$

$$f_6 = x(8) * (Y_{63}x(5) + Y_{66}x(8) + Y_{68}x(10))^*$$

$$f_7 = x(9) * (Y_{75}x(7) + Y_{77}x(9) + Y_{79}x(11))^*$$

$$f_8 = x(10) * (Y_{86}x(8) + Y_{88}x(10) + Y_{89}x(11))^*$$

$$f_9 = x(11) * (Y_{97}x(9) + Y_{98}x(10) + Y_{99}x(11))^* - x(3) - (-0.3 - 0.1i)$$

donde:

$x(1) = S_1$: Potencia compleja barra 1.

$x(2) = S_4$: Potencia compleja barra 4, potencia activa constante en 0.3.

$x(3) = S_9$: Potencia compleja barra 6, potencia activa constante en 0.5.

$x(4) = V_2 \angle \delta_2$: Tensión en la barra 2, magnitud constante en 1.

$x(5) = V_3 \angle \delta_3$: Tensión en la barra 3.

$x(6) = V_4 \angle \delta_4$: Tensión en la barra 4.

$x(7) = V_5 \angle \delta_5$: Tensión en la barra 5.

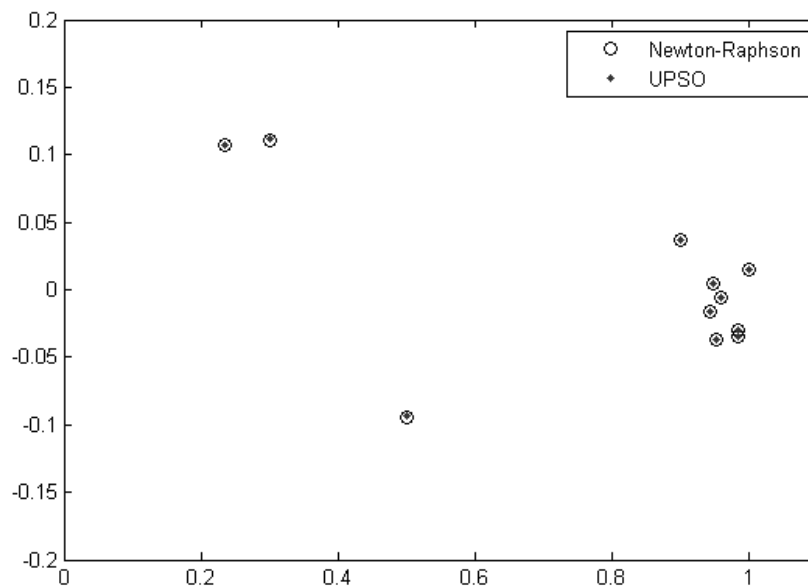
$x(8) = V_6 \angle \delta_6$: Tensión en la barra 6.

$x(9) = V_7 \angle \delta_7$: Tensión en la barra 7.

$x(10) = V_8 \angle \delta_8$: Tensión en la barra 8.

$x(11) = V_9 \angle \delta_9$: Tensión en la barra 9.

Figura 28. Respuesta Newton-Raphson vs UPSO



Fuente: Elaboración propia.

Tabla 5. Resultados UPSO problema tipo de 9 barras

$$x(1) = 0.233912196650879 + 0.107701471710026i$$

$$x(2) = 0.3 + 0.111635502319025i$$

$$x(3) = 0.5 - 0.093862550982286i$$

$$x(4) = 0.9845193646 \angle - 2.042456487^\circ$$

$$x(5) = 0.984283399 \angle - 1.76349779^\circ$$

$$x(6) = 1 \angle 0.8575307415^\circ$$

$$x(7) = 0.9522974071 \angle - 2.226316076^\circ$$

$$x(8) = 0.9579111922 \angle - 0.376402745^\circ$$

$$x(9) = 0.9415941921 \angle - 1.010080993^\circ$$

$$x(10) = 0.9467976517 \angle 0.2815145859^\circ$$

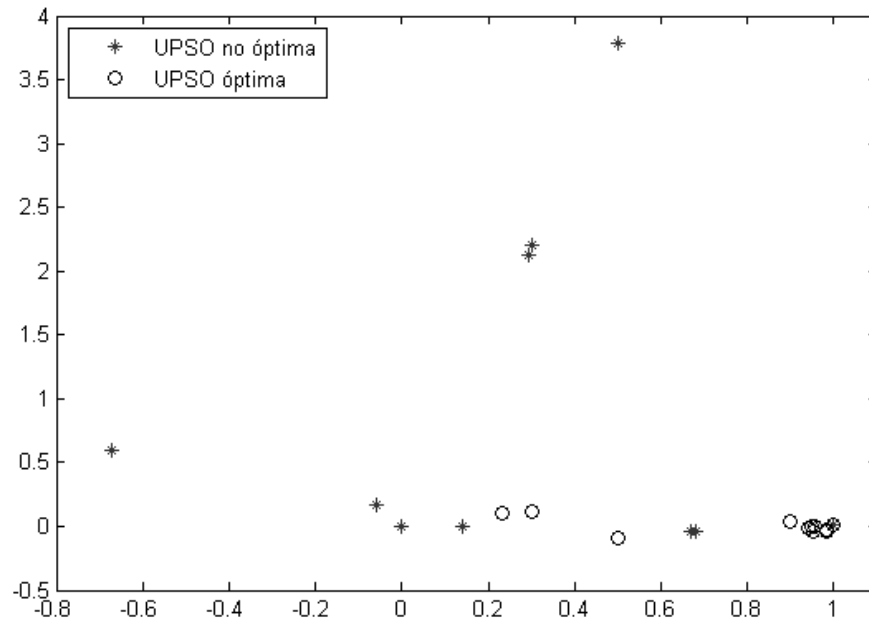
$$x(11) = 0.9 \angle 2.334788358^\circ$$

Fuente: Elaboración propia.

Como se puede observar en la Figura 28 y la Figura 29, el algoritmo tiende a respuestas diferentes. Cuando los valores de la parte real de las partículas se encuentran en el intervalo inicial de posiciones $[0,1]$, el algoritmo tiende a dos soluciones, de las cuales una satisface la solución de Newton-Raphson (DIgSILENT) y otra que cumple la condición de $F_{obj} = 0$ al ser evaluada en la función objetivo.

Al realizar 50 intentos con estos datos, se encontró que se tiende a la respuesta óptima en un 42% de los casos. Con el propósito de obtener dicha tendencia, se realizaron 50 intentos con 1000 partículas, con un factor de constricción de 0.7689, truncando el resultado a un error menor a $1 * 10^{-4}$. De esta forma, se realizan en promedio 172 iteraciones, con un tiempo promedio de 56.7049 [s] por intento; suficiente para mostrar dicha tendencia.

Figura 29. Respuestas UPSO óptima vs no óptima



Fuente: Elaboración propia.

Al cambiar el intervalo de los valores de la parte real de las partículas de $[0,1]$ a $[0.9,1]$ la frecuencia de llegada a la respuesta óptima aumentó a un 100% de los casos. De lo visto en trabajos anteriores [8], se observó que el método presenta una tendencia a la respuesta más cercana al intervalo de inicio. Después de este análisis, es correcto afirmar que se puede llegar a una de las varias respuestas del sistema de ecuaciones dependiendo de la posición inicial de las partículas. En el Anexo se encuentran las gráficas que representan el movimiento de la partícula más cercana a la respuesta, para cada dimensión de este sistema, a través de las iteraciones.

3. CONSIDERACIONES

No hay teoremas que den un valor fijo de los parámetros para problemas específicos. Por eso se hace necesario hallar de forma empírica los valores del factor de unificación (u), para encontrar una respuesta de manera más eficiente, como se ve en el análisis realizado en la sección 2.4. Además, el valor del coeficiente de constricción está dado por la ecuación $X = 2/(2 - \varphi - \sqrt{(\varphi^2 - 4\varphi)})$ como se puede observar en la referencia [10], donde $\varphi = c_1 + c_2$ y además debe cumplirse que $\varphi > 4$.

Al tomar un rango muy grande de posiciones iniciales, las partículas pueden presentar una tendencia hacia valores que no son los esperados. Luego se hace importante no sólo conocer el intervalo solución, sino acotar el intervalo inicial de forma que se pueda asegurar que el método se acercará a la respuesta esperada. Dada la particularidad del problema bajo consideraciones, se puede observar que, para el caso de 9 barras, 8 de las 11 incógnitas de la respuesta tienen valores cercanos a 0.9 en su parte real, haciéndose más sencillo acotar el intervalo inicial.

Al plantear el sistema de ecuaciones del que se deducirá la función objetivo, se puede llegar al punto de tener mayor número de incógnitas que de ecuaciones, lo cual se interpreta como un problema de respuestas infinitas. Pero, los sistemas de potencia se modelan teniendo en cuenta restricciones particulares, como lo son las magnitudes de las tensiones en ciertas barras, o la potencia activa que entregan los generadores, limitando el problema para que tenga una respuesta particular que satisface el sistema.

Una anotación importante considerada al realizar estos análisis, es que las iteraciones se realizaron en el mismo equipo, un computador de alto rendimiento, dado que equipos con diferentes capacidades pueden afectar el desempeño del algoritmo, especialmente en el tiempo de las iteraciones.

4. CONCLUSIONES - RECOMENDACIONES³

1. Se cumplieron a cabalidad los objetivos aprobados en el plan de proyecto.
2. El método de optimización por enjambre de partículas unificado (UPSO) converge a la solución del flujo de cargas de un sistema de potencia; pero se deben tener consideraciones previas para llegar a una respuesta satisfactoria, como las mencionadas en la sección 3. Esto debido a que el algoritmo tiene la capacidad de encontrar diferentes soluciones, y que el planteamiento del problema depende de las condiciones específicas. En consecuencia, se puede llegar a soluciones no deseadas; lo que es de esperarse por tratarse de un método estocástico.
3. Al utilizar un número elevado de partículas (NP), se puede incurrir en tiempos demasiado largos y alto tiempo computacional para llegar a una misma respuesta, sin presentar una mejoría notable en el número de iteraciones. Por esto, es importante tener en cuenta la relación partículas vs tiempo, que permita encontrar un balance y tener un mejor desempeño en el momento de solucionar un problema.
4. Se pueden utilizar las ecuaciones de modificación de velocidad del algoritmo UPSO para trabajar en radianes. Dado que una de las condiciones del problema es mantener la magnitud de ciertas tensiones en valores constantes, se vio la necesidad de variar la velocidad angular de las coordenadas de las partículas correspondientes a los ángulos, utilizando una variante de la fórmula general establecida en la ecuación (5). En ese caso, en lugar de x_i , se tomó θ_i que es el ángulo

³ Esto constituye un aporte intelectual consensuado entre los estudiantes Pinzón, Ardila y el profesor Rodrigo Correa. Su formulación fue la base para el desarrollo del presente trabajo de grado.

correspondiente a la partícula i -ésima. Así mismo, en lugar de p_i , p_g y p_{gi} , se toman los ángulos correspondientes. Luego, la coordenada de la partícula es examinada en la ecuación $x_i = K_i * (\cos(\theta_i) + j * \sen(\theta_i))$, donde K_i corresponde a la magnitud de dicha partícula, asegurando que se tenga un vector de componentes imaginarias con valores arbitrarios, pero de magnitud constante.

5. Al comparar los dos métodos para los ejemplos demostrativos utilizados, es posible afirmar que ambos logran soluciones similares; aunque dado el tamaño del sistema, el método Newton-Raphson los obtiene en un menor tiempo. No obstante, este comportamiento depende de la calidad de los valores iniciales.
6. Para la comparación entre los resultados obtenidos con UPSO y la respuesta de Newton-Raphson, se utilizó la herramienta comercial PowerFactory 14 de DlgSILENT en su versión Demo. Esta herramienta desarrolla flujos de cargas a través de dicho método y en forma gráfica, como se presentó en la sección 2.3. Es recomendable para este tipo de proyectos, incentivar el uso de software de tipo comercial, debido al uso extenso en la práctica profesional, beneficiando a los estudiantes que puedan trabajarlo con antelación durante la realización de su trabajo de grado.
7. Los valores utilizados para los distintos parámetros necesarios para la ecuación principal de velocidad (coeficientes cognitivo, social y de constricción) fueron tomados de la literatura. Por tanto, se recomienda como próximo paso, analizar de manera estadística diferentes valores que se pudieran tomar, con el fin de observar el desempeño que puede llegar a tener el algoritmo en un problema específico y analizar la capacidad

comercial del mismo. Además, se puede considerar la posibilidad de que el parámetro de unificación u presente un comportamiento dinámico.

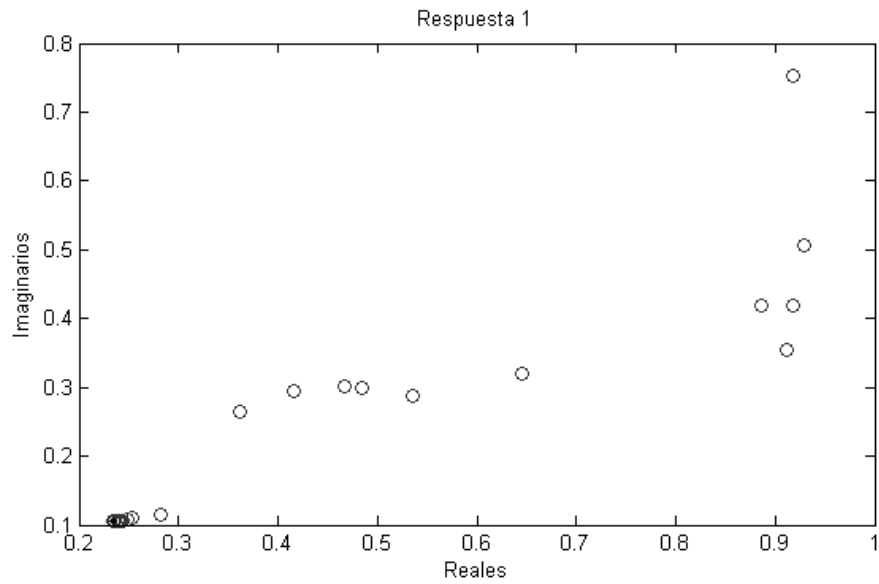
8. Como se puede observar en los resultados obtenidos, al tomar un error de referencia menor de $1 \cdot 10^{-9}$, se presenta un error considerable en los cálculos de potencias, llegando a diferencias de milésimas. Dado que en un principio se tomó esta referencia de forma arbitraria, se requiere hacer un análisis más detallado de estas respuestas, con el fin de encontrar un criterio de parada que se ajuste al problema en cuestión y dé como resultado respuestas más acordes.

BIBLIOGRAFÍA

- B. M. WEEDY, “Capítulo 5 - Flujos de carga,” in *Sistemas Eléctricos de Gran Potencia*, 2nd ed., J. Wiley, Ed. España: Editorial Reverté, 1982, pp. 182–224. [3].
- C. DUARTE and J. QUIROGA, “Algoritmo PSO para identificación de parámetros en un motor DC,” *Revista Facultad de Ingenierías Universidad de Antioquia*, vol. 55, pp. 116–124, 2010. [5].
- G. CARRILLO CAICEDO, “Capítulo 4 - Flujo de carga,” in *Sistemas de Potencia*, 1st ed., D. de P. UIS, Ed. Bucaramanga: UIS, Departamento de Electricidad y Electrónica, 1984, pp. 103–161. [2].
- I. AMAYA, J. CRUZ, and R. CORREA, “Real Roots of Nonlinear Systems of Equations Through a Metaheuristic Algorithm,” *Revista Dyna*, vol. 78, no. 170, pp. 15–23, 2011. [8].
- J. KENNEDY and R. EBERHART, “Particle Swarm Optimization,” in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, 1995, pp. 1942–1948. [6].
- K. A. DE JONG, “An Analysis of the Behavior of a Class of Genetic Adaptive Systems - A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy,” University of Michigan, 1975. [9].
- K. E. PARSOPOULOS and M. N. VRAHATIS, “Parameter selection and adaptation in Unified Particle Swarm Optimization,” *Mathematical and Computer Modelling*, vol. 46, no. 1–2, pp. 198–213, Jul. 2007. [10].
- K. E. PARSOPOULOS and M. N. VRAHATIS, “Unified Particle Swarm Optimization for Solving Constrained Engineering Optimization Problems,” pp. 582–591, 2005. [4].
- K. E. PARSOPOULOS and M. N. VRAHATIS, “Unified Particle Swarm Optimization,” pp. 590–599, 2005. [1].
- K. E. PARSOPOULOS and M. N. VRAHATIS, UPSO: A Unified Particle Swarm Optimization Scheme, *Lecture series on Computer and Computational Sciences*, vol. 1, pp. 868–873, 2004. [7].

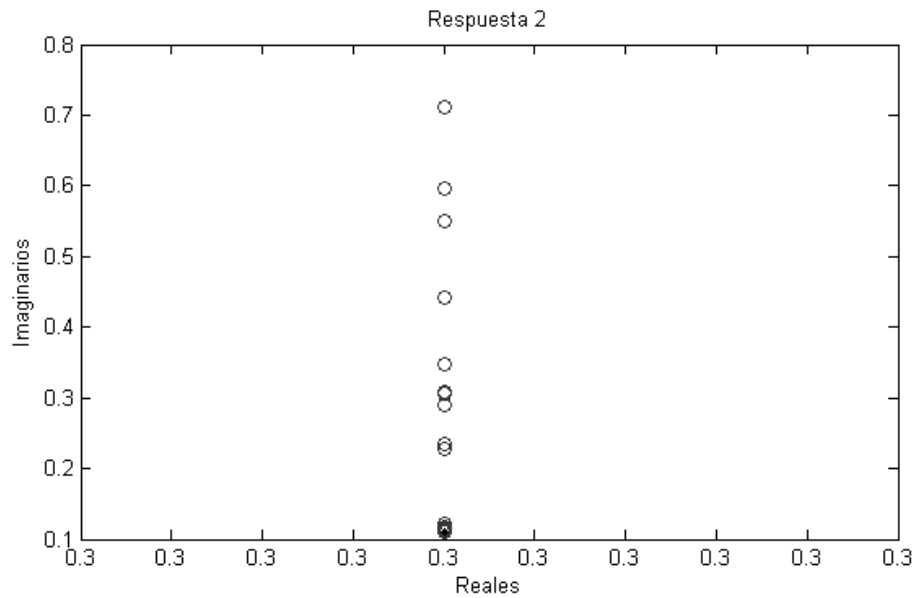
ANEXO

Figura 30. Respuesta 1 - UPSO sistema de 9 barras



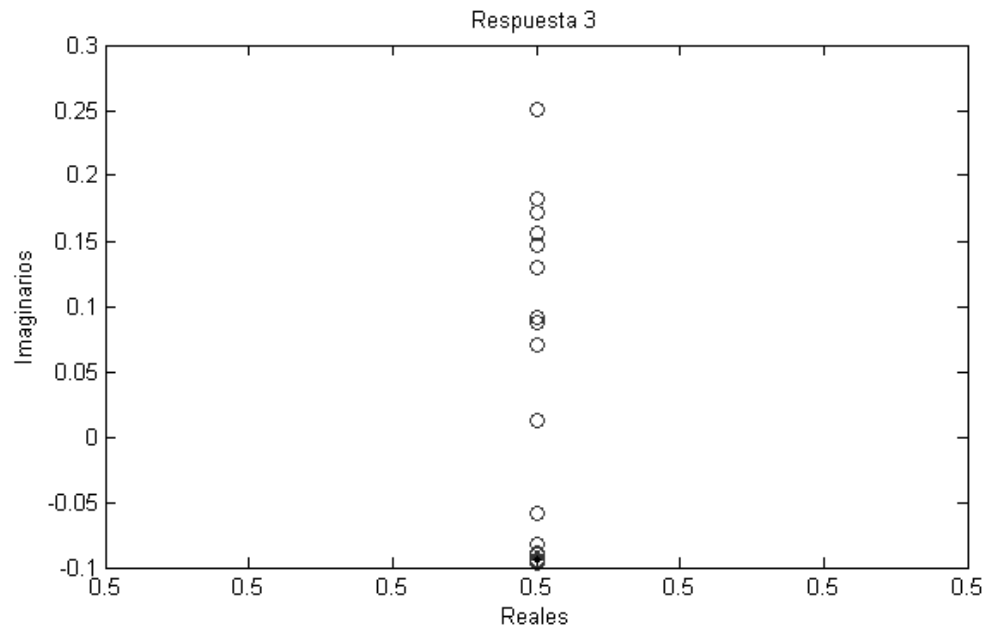
Fuente: Elaboración propia.

Figura 31. Respuesta 2 - UPSO sistema de 9 barras



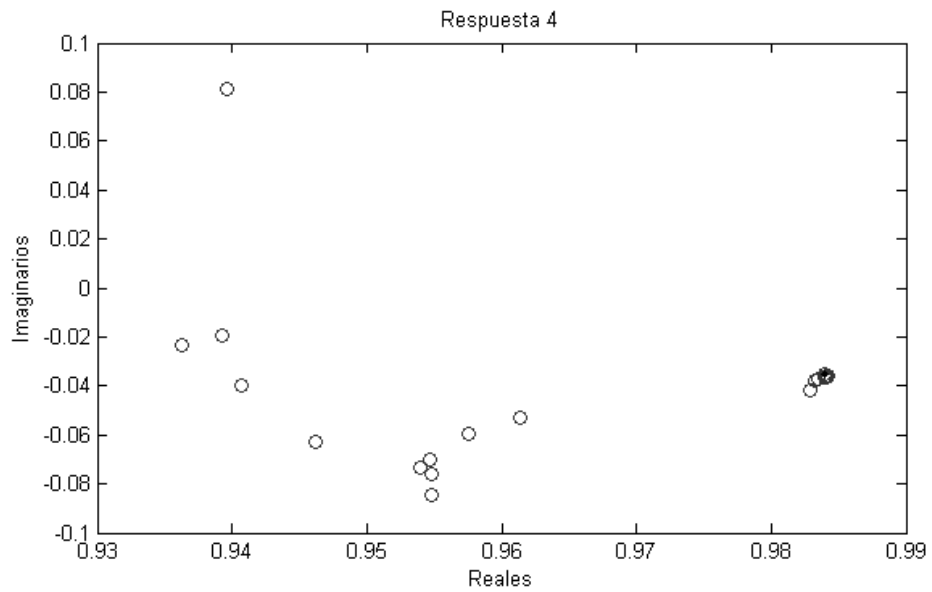
Fuente: Elaboración propia.

Figura 32. Respuesta 3 - UPSO sistema de 9 barras



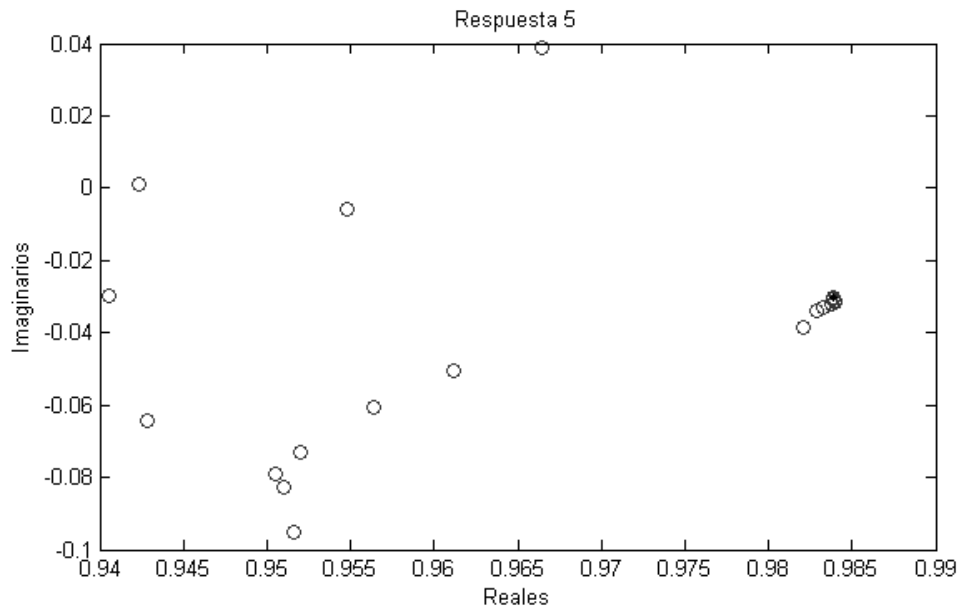
Fuente: Elaboración propia.

Figura 33. Respuesta 4 - UPSO sistema de 9 barras



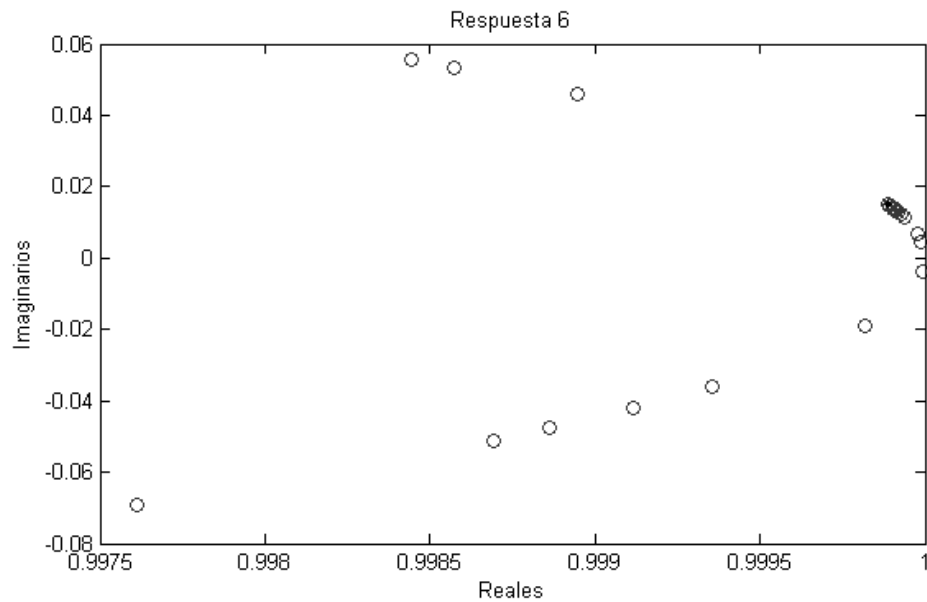
Fuente: Elaboración propia.

Figura 34. Respuesta 5 - UPSO sistema de 9 barras



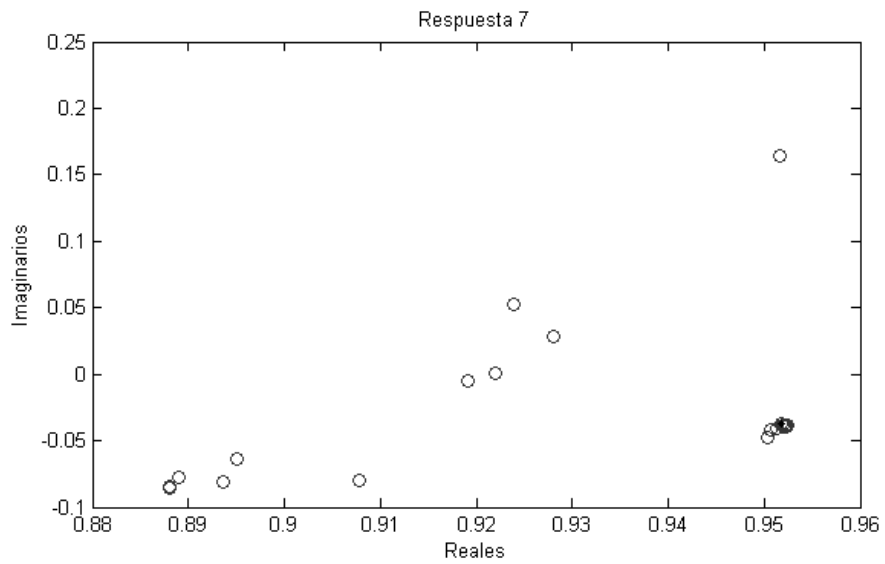
Fuente: Elaboración propia.

Figura 35. Respuesta 6 - UPSO sistema de 9 barras



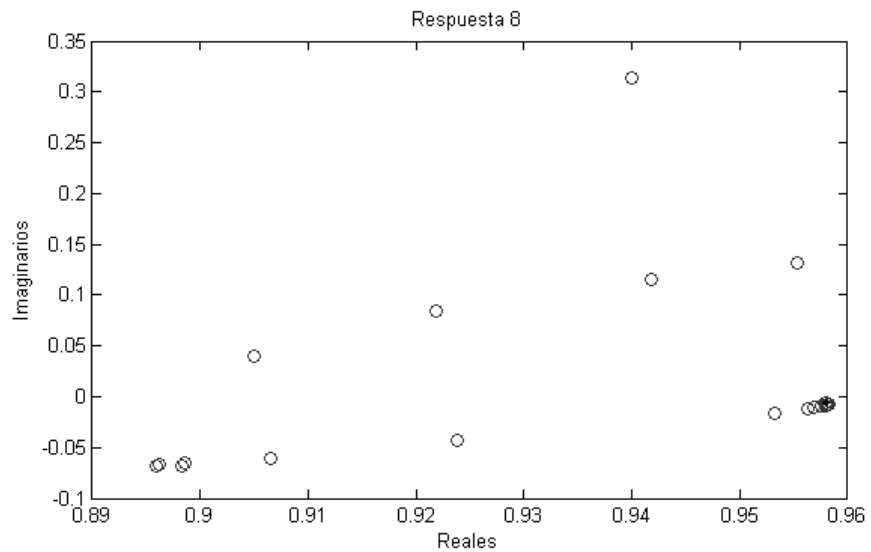
Fuente: Elaboración propia.

Figura 36. Respuesta 7 - UPSO sistema de 9 barras



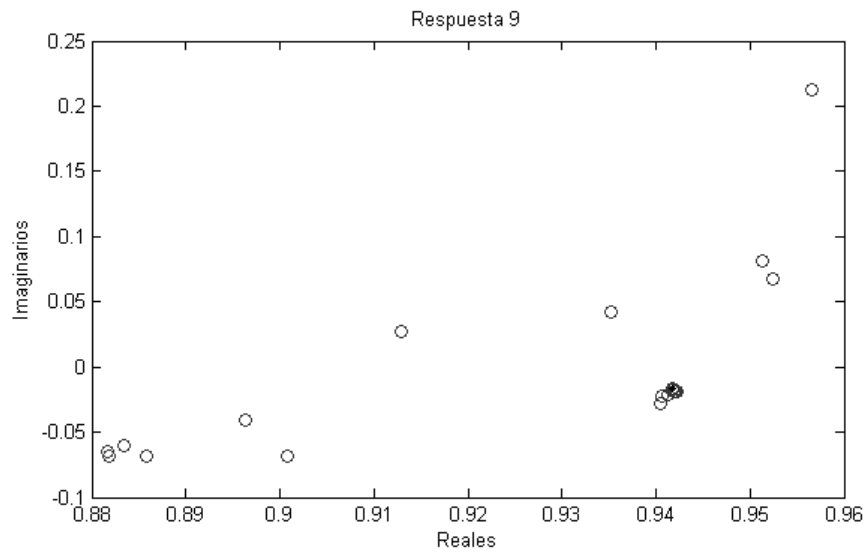
Fuente: Elaboración propia.

Figura 37. Respuesta 8 - UPSO sistema de 9 barras



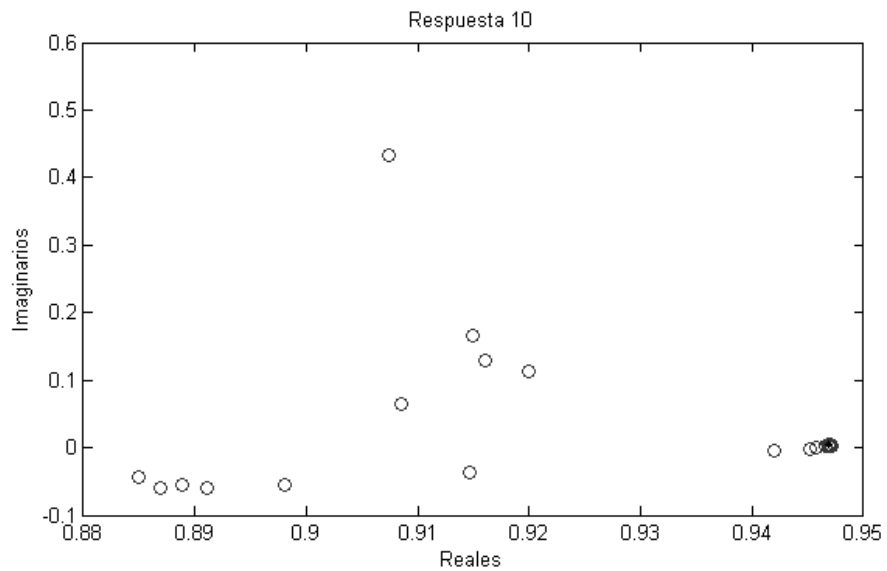
Fuente: Elaboración propia.

Figura 38. Respuesta 9 - UPSO sistema de 9 barras



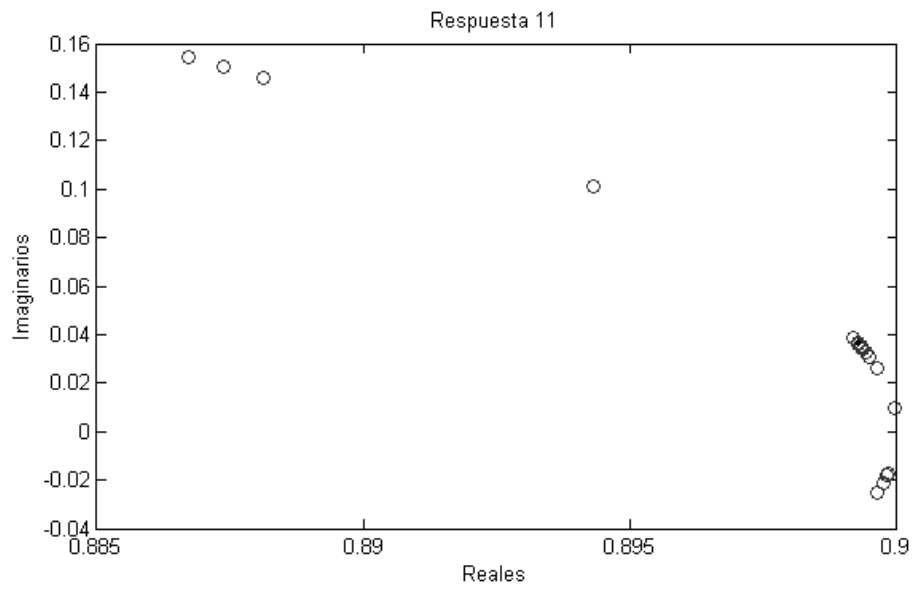
Fuente: Elaboración propia.

Figura 39. Respuesta 10 - UPSO sistema de 9 barras



Fuente: Elaboración propia.

Figura 40. Respuesta 11 - UPSO sistema de 9 barras



Fuente: Elaboración propia.