APLICACIÓN DEL ALGORITMO METAHEURÍSTICO OPTIMIZACIÓN EVOLUTIVA POR ENJAMBRE DE PARTÍCULAS (EPSO) A LA ALTERNATIVA DE ASIGNACIÓN DE ENERGÍA ELÉCTRICA: MINIMIZACIÓN DE PAGOS FINALES (PCM).

OSCAR JAVIER MUÑOZ SARMIENTO

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICO – MECÁNICAS
ESCUELA DE ESTUDIOS INDUSTRIALES Y EMPRESARIALES
BUCARAMANGA
2011

APLICACIÓN DEL ALGORITMO METAHEURÍSTICO OPTIMIZACIÓN EVOLUTIVA POR ENJAMBRE DE PARTÍCULAS (EPSO) A LA ALTERNATIVA DE ASIGNACIÓN DE ENERGÍA ELÉCTRICA: MINIMIZACIÓN DE PAGOS FINALES (PCM).

OSCAR JAVIER MUÑOZ SARMIENTO

Trabajo de Grado para optar al título de
INGENIERO INDUSTRIAL

DIRECTOR
PhD. HENRY LAMOS DÍAZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICO – MECÁNICAS
ESCUELA DE ESTUDIOS INDUSTRIALES Y EMPRESARIALES
BUCARAMANGA
2011

Al Señor Jesús

Quien en su amor me permito tener a mi lado unos magníficos padres, un gran hermano, excelentes amigos y la Mujer más maravillosa del mundo "Marcelita".

Agradecimientos.

Al Ingeniero Juan Fernando, por su constante apoyo.

Al doctor Henry Lamos, por su orientación y dedicación.

A la Escuela de Estudios Industriales y Empresariales y a cada uno de sus docentes.

.

TABLA DE CONTENIDO

INTRODUCCIÓN. 14

1. OBJETIVOS 16

1.1 OBJETIVO GENERAL. 16

1.2 OBJETIVOS ESPECÍFICOS. 16

2. MARCO TEORICO. 17

2.1 FORMULACIÓN MATEMÁTICA DEL CRITERIO DE ASIGNACIÓN BCM. 17

2.2 FORMULACIÓN MATEMÁTICA DEL CRITERIO DE ASIGNACIÓN PSMP. 19

2.3 SISTEMA DE LIQUIDACIÓN ADOPTADO MUNDIALMENTE (PAB Ó PSMP). 21

2.4 TEORÍA DE SUBASTAS 22

2.4.1 NATURALEZA DE LAS SUBASTAS 23

2.4.2 Formas de subasta 24

2.5 DESCRIPCIÓN DEL PROBLEMA 26

2.6 SITUACIÓN ACTUAL. 27

2.7 COMPLEJIDAD COMPUTACIONAL DEL PROBLEMA. 27

2.8 CONCEPTO DE HEURÍSTICA Y META-HEURÍSTICA. 28

2.8.1 Búsqueda tabú. (TS) 29

2.8.2 Estrategias evolutivas (EE). 31

2.8.3 Optimización de Enjambre de Partículas. (Particle Swarm Optimization PSO.) 33

2.8.4 Optimización Evolutiva de Enjambre De Partículas (Evolutionary Particle Swarm Optimization EPSO) 37

3. DISEÑO DE LA SOLUCIÓN....42

3.1 DESCRIPCIÓN DE LA ASIGNACIÓN PCM 42

3.1.1 Restricciones....44

3.2 ESTRATEGIAS DE SOLUCIÓN 45

3.3 DESCRIPCIÓN DETALLADA DEL ALGORITMO EPSO – EVOLUTIONARY SELF-ADAPTING PSO PARA LA PROGRAMACIÓN BINARIA. 47

3.4 APLICACIÓN DEL ALGORITMO EPSO AL TIPO DE ASIGNACIÓN DE ENERGÍA (PCM). 48

4. RESULTADOS NUMÉRICOS. 51

5. CONCLUSIONES....55

6. RECOMENDACIONES 57

7. REFERENCIAS BIBLIOGRÁFICAS: 58

8. BIBLIOGRAFÍA 60

ANEXOS....62

LISTA DE TABLAS

Tabla 1: Bloque de ofertas 20
Tabla 2 Resultado de la minimización BCM 21
Tabla 3: Resultado de la minimización PCM 21
Tabla 4: Pago incurrido mediante el esquema de liquidación PSMP asignando mediante BCM 26
Tabla 5: Permutaciones posibles de acuerdo al número de participantes y periodos de tiempo. 28
Tabla 6: Ejemplo de asignación de energía para una sola hora 46
Tabla 7: Escenarios de demanda ejemplo 1 51
Tabla 8: Bloque de ofertas ejemplo 1 52
Tabla 9 Comparación de los resultado obtenidos por los algoritmos TS, PSO y EPSO Para el ejemplo 1 53

LISTA DE ILUSTRACIONES

Ilustración 1: Subasta Tipo Oferta...23

Ilustración 2: Subasta Tipo Demanda 24

Ilustración 3: Subastas Tipo Doble 24

Ilustración 4 Concepto de la modificación de un punto de búsqueda en PSO. Fuente: Kwang Y. Lee and Mohamed A. El-Sharkawi. "MODERN HEURISTIC OPTIMIZATION TECHNIQUES".Theory And ApplicationsToPowerSystems 36

Ilustración 5 Concepto de la modificación de un punto de la búsqueda en EPSO. Fuente: Rolando M. Pringles, Vladimiro Miranda y Francisco Garcés. Expansión Óptima del Sistema de Transporte Implementando EPSO 41

Ilustración 6 Diagrama de flujo algoritmo EPSO binario, Fuente Autor..................49

Ilustración 7: Comportamiento durante las iteraciones de los algoritmos TS, PSO y EPSO Para el ejemplo 1 53

LISTA DE ANEXOS

Anexo 1. Pasos para el uso del software desarrollado en MATLAB…………….64

Anexo 2. Implementación en MATLAB del algoritmo optimización evolutiva de enjambre de partículas (EPSO) propuesto……………………………………………..73

RESUMEN

TITULO: APLICACIÓN DEL ALGORITMO METAHEURÍSTICO OPTIMIZACIÓN EVOLUTIVA POR ENJAMBRE DE PARTÍCULAS (EPSO) A LA ALTERNATIVA DE ASIGNACION DE ENERGIA ELECTRICA: MINIMIZACIÓN DE PAGOS FINALES (PCM)*.

AUTOR: OSCAR JAVIER MUÑOZ SARMIENTO**

PALABRAS CLAVES: Operador del Mercado, Generadores, Subasta, Ofertas, Esquemas de pago, Complejidad computacional, Meta-heurística.

DESCRIPCIÓN

Actualmente muchos de los mercados de energía mayorista en el mundo operan en un esquema de subasta, bajo la coordinación del “Operador del Mercado”. Este se encarga de recibir las ofertas de los participantes y realizar la asignación correspondiente para suplir la demanda. Esta subasta se realiza a partir de ciertos criterios que suelen minimizar la asignación, teniendo en cuenta lo ofertado por cada participante. Una vez terminado la asignación, sigue la liquidación a cada agente generador, para esto se utiliza un esquema de pago uniforme, ya que bajo la rama de teoría de juegos, se ha comprobado que este incentiva a los generadores a reducir sus ofertas, lo cual implica directamente una reducción en los pagos como resultado de la asignación.

Al asignar según las ofertas de los generadores y luego liquidar los pagos de una manera uniforme, evitaría obtención de menores costos en el consumidor final de la energía eléctrica. Para cubrir este problema varias investigaciones proponen un criterio de asignación de energética, que tiene en cuenta el precio marginal del sistema, lo que lo hace estar más afín al esquema de liquidación uniforme, este criterio de minimización es conocido como minimización de pagos finales, pero su implementación no ha sido factible ya que conlleva a tiempos de ejecución muy prolongados, y además su complejidad hace que los agentes generadores opongan cierta resistencia a la implementación del mismo.

Este trabajo presenta la implementación del algoritmo meta-heurístico, Optimización Evolutiva de Enjambre de Partículas , como herramienta para la solución al problema de asignación de energía eléctrica, con el fin de minimizar los tiempos de ejecución, ya que históricamente comparando los resultados obtenidos al utilizar EPSO con respecto a otros métodos meta-heurísticos como Búsqueda Tabú, Recocido Simulado, Estrategias Evolutivas, entre otros, se obtiene una mejor calidad en sus optimizaciones.

***** Proyecto de Grado. ** Facultad de Ingenierías Físico-Mecánicas. Escuela de Estudios Industriales y Empresariales. Ph.D. Henry Lamos Diaz.

SUMARY

TITLE: IMPLEMENTATION OF ALGORITHM EVOLUTIONARY PARTICLE SWARM OPTIMIZATION (EPSO) TO THE POWER ASSIGNMENT ALTERNATIVE: PAYMENT COST MINIMIZATION (PCM)*.

AUTHOR: OSCAR JAVIER MUÑOZ SARMIENTO**

KEYWORDS: Market Operator, Generator, Auction, Bids, payment schemes, computational complexity, Meta-heuristics.

DESCRIPTION

Currently the wholesale electricity markets in the world operate in an auction scheme, coordinated by the market operator; this is responsible for receiving the bids of the participants and making an appropriate assignment to supply the demand of a certain period of time. This auction is done based on certain criteria that tend to minimize the assignation, taking into account each participant bid. Once the assignment is completed, follows the liquidation of each agent generator that is determinate by using a uniform payment scheme known as “System Marginal Price” because mathematical studies under the branch of game theory, has found that when using a uniform payment scheme (pay at system marginal Price), the generators tend to reduce their bids, which directly implies a reduction of payments as a result of the assignment.

At assign according to generators bids, and then settle the payments in a uniform manner, it might avoid obtaining lower costs for the final consumer of electricity. To cover this problem several researchers propose a power allocation criterion, which takes into account the system marginal price, and makes it more consistent with the uniform payment scheme, the minimization criterion is known as (Payment Cost Minimization), but its implementation has not been feasible because it involves a very long run times, and also its complexity makes the agents oppose some resistance to its implementation.

This paper presents the implementation of meta-heuristic algorithm, Evolutionary Optimization Particle Swarm (EPSO), as a tool for solving the problem of energy assignment in order to minimize the execution time because, historically, comparing results obtained using EPSO with respect to meta-heuristics methods like Tabu Search, Simulated Annealing, Evolutionary Strategies, among others, EPSO provides a better quality in their optimizations.

***** Graduation Project ** Ability of Engineerings Physique-Mechanical. Industrial and Business Studies School. Ph.D. Henry Lamos Diaz.

INTRODUCCIÓN.

La creación de mercados de energía eléctrica (bolsas de energía o “pools”) y el desarrollo de sistemas de intercambios comerciales como se menciona en [1] y [2] han creado nuevos mecanismos de selección de las ofertas realizadas por parte de los generadores de energía (Plantas hidráulicas, térmicas, eólicas, entre otras). El criterio de minimización de costos de oferta BCM (Bid Cost minimization) y el criterio de minimización de pagos finales PCM (Payment Cost Minimization) son ampliamente usados como modelos de asignación. Al BCM y PCM utilizar ofertas de generación complejas[1], hace que sus criterios matemáticos caigan en la clase de optimización combinatoria. Por consiguiente, es necesario construir algoritmos rápidos y eficientes para hallar una buena solución del problema combinatorio.

El presente trabajo se realiza bajo la modalidad de “Trabajo de investigación” con el aval del grupo de investigación OPALO que centra sus procesos en la Modelación para resolver problemas de Ingeniería Industrial. El proyecto es una continuación del trabajo realizado por el Ingeniero electricista Juan Fernando Rodríguez [3] de la Escuela de Ingeniería Eléctrica Electrónica y de Telecomunicaciones E^3T de la Universidad Industrial de Santander, en donde demuestra que las asignaciones realizadas a través de los modelos actuales no permiten un mejor minimización del costo de energía eléctrica al consumidor, y además plantea un algoritmo matemático que pretende mejorar la calidad de la asignación energética actual.

Las herramientas heurísticas son una alternativa muy interesante para hallar soluciones buenas de problemas combinatorios para instancias grandes. Estas

[1]Son aquellas ofertas que separan los costos de arranque y parada de la oferta marginal.

[2] El operador del mercado es un ente independiente de los generadores de energía, el cual entre sus principales actividades esta en conocer el pronóstico de la demanda y la asignación de generadores para suplir la misma, en Colombia el único

herramientas incluyen la Computación Evolutiva, Recocido Simulado, Búsqueda Tabú, Enjambre de Partículas, entre otras. El desarrollo de soluciones a través de estos algoritmos heurísticos tiene ventajas importantes con respecto al tiempo de computación.

En el presente trabajo se estudia e implementa un algoritmo evolutivo que se basa en las reglas de movimiento del algoritmo Optimización de Enjambre de Partículas (Particle Swarm Optimization PSO). El algoritmo se denomina Optimización Evolutiva de Enjambre de Partículas (Evolutionary Particle Swarm Optimization EPSO). La motivación de implementar EPSO como herramienta para la solución al problema de asignación es la buena calidad de las soluciones que se obtienen respecto a otros métodos meta-heurísticos, posicionando el método como una opción importante cuando se pretende resolver problemas de optimización complejos. El trabajo se presenta de la siguiente forma: en el capítulo II se realiza una descripción del tipo de asignación que se utiliza generalmente en el mundo (BCM) y la asignación (PCM) también se presenta un enfoque de teoría de subastas y por último se presenta un descripción del algoritmo meta-heurístico EPSO. En el capítulo III se desarrolla el modelo matemático PCM (Payment Cost Minimization), se describen una estrategia para la solución, y por último se muestra la aplicación del algoritmo EPSO para la solución del problema combinatorio. En el capítulo IV se compara el método desarrollado usando instancias resueltas a través de otros algoritmos en [3]. Por ultimo en los capítulos V y VI se dan las conclusiones y recomendaciones de la presente investigación.

1. OBJETIVOS

1.1 OBJETIVO GENERAL.

Hallar la solución al problema de minimización de pagos finales (“Payment Cost Minimization”, PCM) en la transacción de energía eléctrica en bolsas aplicando el método meta-heurístico Optimización Evolutiva de Enjambre de Partículas (“Evolutionary Particle Swarm Optimization”, EPSO).

1.2 OBJETIVOS ESPECÍFICOS.

- ✓ Recopilar, seleccionar y organizar la Literatura existente concerniente al método metaheurístico Optimización Evolutiva de Enjambre de Partículas ó Evolutionary Particle Swarm Optimization EPSO y su aplicabilidad en la Programación no lineal entera mixta.
- ✓ Plantear la función objetivo y restricciones a la alternativa de asignación (Payment Cost Minimization PCM) en el esquema de liquidación (Pay at Market Clearing Price Scheme PSMP) en subastas de energía eléctrica en bolsas.
- ✓ Desarrollar un software en MATLAB aplicando el método meta-heurístico (EPSO) al tipo de asignación PCM.
- ✓ Realizar experimentos numéricos que permitan medir la eficiencia del software para la solución del problema de optimización de los pagos finales PCM.
- ✓ Elaborar un documento que contenga la base teórica del método meta-heurístico EPSO, resultados finales del software y conclusiones.

2. MARCO TEORICO.

La mayor parte de los mercados de energía eléctrica se basan en subastas de energía (Bolsas de Energía) realizadas por el operador del mercado[2], como medio para decidir cuales generadores suplirán la energía demandada por los distribuidores, dicha decisión se conoce como “asignación energética” en la actualidad existen dos criterios de asignación, “Minimización de costos de oferta (BCM)” y “Minimización de Pagos Finales (PCM)”, Además, el diseño de mercados incluye diversas reglas y procedimientos para determinar los pagos y cobros (liquidación) correspondientes a cada agente del mercado de acuerdo a su operación.

Existen dos tipo de liquidación: “pago según lo ofertado” PAB (“Pay-as- Bid”) y “pago al precio del mercado” PSMP (“Pay at System Marginal Price”). En el PAB, el valor del pago que cada generador recibe por la energía eléctrica generada es el de su oferta[3] en el momento de cierre de la subasta. Por el contrario, el PSMP se basa en que todos los agentes son liquidados a un precio uniforme independientemente del valor de sus ofertas. Este precio es conocido como el precio del mercado o “System Marginal Price”–SMP– y se define como el valor de la oferta marginal de generación (oferta más costosa de generación aceptada en el momento de cierre de la subasta).

2.1 FORMULACIÓN MATEMÁTICA DEL CRITERIO DE ASIGNACIÓN BCM.

Considerando un mercado de energía con *I* participantes o generadores (donde $i=1,2,\ldots, I$), los cuales son necesarios para suplir la demanda energética de un

[2] El operador del mercado es un ente independiente de los generadores de energía, el cual entre sus principales actividades esta en conocer el pronóstico de la demanda y la asignación de generadores para suplir la misma, en Colombia el único Operador del Mercado es XM(Expertos en Energía).
[3] Un agente generador determina su oferta de precio considerando el precio de cada MW generado y sus costos de arranque y parada.

serie de periodos (L_t) donde L es la demanda en MW y *t* corresponde a intervalo *(1 ≤ t ≤ T)* (por lo general T=24 refiriéndose a las 24 horas del día). El objetivo de la asignación es la selección de generadores $\mu_{i,t}$ (donde si $\mu_{i,t}$ = 1 indica que el generador i es seleccionado para la hora t y si $\mu_{i,t}$ = 0 lo contrario) y sus correspondientes niveles de potencia ($P_{i,t}$), para suplir la demanda de todos los periodos de tiempo (L_t).

Para ello, cada uno de los participantes presentan un solo bloque de ofertas**,** el cual contiene la oferta marginal $mo_{i,t}$ (precio de cada MW generado), sus niveles de generación de potencia máxima y mínima para todos los periodos de tiempo ($P_{i,t}^{min}$ y $P_{i,t}^{max}$), y sus costos de arranque y parada CAP_i.

El costo que se incurre según el criterio de asignación BCM a un generador i en una hora t, será su oferta marginal por la potencia asignada $mo_{i,t}$*$P_{i,t}$, además se debe tener en cuenta los costos de arranque y parada, pero estos solo son pagados únicamente si se obliga al generador encenderse nuevamente para la hora t, es decir que si en *t-1* el generador *i* estaba apagado *($\mu_{i,t-1}$ = 0)* y en t se decide que el generador i se encienda *($\mu_{i,t}$ = 1)* esto obliga a pagar el costo de arranque y parada *CAP*, matemáticamente se puede ver con la siguiente expresión $\mu_{i,t} *(CAP_i * \neg\mu_{i,t-1})$ (donde $\neg\mu_{i,t-1}$es la negación de $\mu_{i,t-1}$). En resumen la expresión matemática del pago al generador i en un tiempo t está dada por:

$$[\, mo_{i,t} * P_{i,t} + CAP_i * (\neg\mu_{t-1})] * \mu_{i,t} \quad (1)$$

Cabe resalta que $P_{i,t}$ debe estar restringido entre los limites de potencia máxima y mínima de los generadores ($P_{i,t}^{min} \leq P_{i,t} \leq P_{i,t}^{max}$).

Por lo tanto el modelo asignación BCM que tiene como objetivo la minimización de los costos según las ofertas de los generadores para así suplir la demanda cada uno de los periodos t, se formula mediante la función de Costo Tota (TP).

$$TP = \left\{ \sum_{t=1}^{T} \sum_{i=1}^{I} \left(mo_{i,t} * + CAP_i * \left(\neg \mu_{i,t-1} \right) \right) \mu_{i,t} \right\} \quad (2)$$

Sujeto a:

$$L_t = \sum_{i=1}^{I} P_{i,t} \quad \forall \ t = 1,2, \dots T$$

$$P_{i,t} = 0, \qquad si \quad \mu_{i,t} = 0 \quad \forall \ i = 1,2, \dots I$$

$$P_{i,t}^{min} \leq P_{i,t} \leq P_{i,t}^{max} \qquad si \quad \mu_{i,t} = 1 \quad \forall \ i = 1,2, \dots I$$

2.2 FORMULACIÓN MATEMÁTICA DEL CRITERIO DE ASIGNACIÓN *PSMP.*

Este criterio no varía matemáticamente en una forma significativa con respecto a la asignación BCM, su diferencia radica en que este minimiza conforme al precio marginal del mercado el cual se determina con el valor de la oferta más costosa de generación aceptada en el momento de cierre de la asignación.

Por lo tanto el costo incurrido según el criterio de asignación PCM que se realiza a un generador i en una hora t, ya no vendrá a ser su oferta marginal por la potencia asignada $mo_{i,t}$*$\mathrm{P}_{\mathrm{i,t}}$. Si no por el contrario es necesario saber cual fue la oferta más costosa aceptada para el periodo de tiempo t (π_t).

$$\pi_t = \max\left\{ \left(mo_{i,t} * \mu_{i,t} \right) \right\} \quad (3)$$

Entonces el pago que se realiza a un generador estará dado por la ecuación π_t*$\mathrm{P}_{\mathrm{i,t}}$. Por otro lado las demás condiciones como el pago de los costos de arranque y parada y las restricciones de generación y cumplimiento de la demanda se mantienen igual. Por lo tanto el modelo asignación PCM tiene como objetivo la minimización de los pagos liquidados según PSMP para así suplir la demanda en cada uno de los periodos t y este se formula mediante la función de Pago Tota (TP).

$$TP = \left\{ \sum_{t=1}^{T} \sum_{i=1}^{I} \left(\pi_t * P_{i,t} + CAP_i * \left(\neg \mu_{i,t-1} \right) \right) \mu_{i,t} \right\} \quad (4)$$

Sujeto a:

$$L_t = \sum_{i=1}^{I} P_{i,t} \quad \forall \; t = 1,2, \dots T$$

$$P_{i,t} = 0, \qquad si \quad \mu_{i,t} = 0 \qquad \forall \; i = 1,2, \dots I$$

$$P_{i,t}^{min} \leq P_{i,t} \leq P_{i,t}^{max} \qquad si \quad \mu_{i,t} = 0 \qquad \forall \; i = 1,2, \dots I$$

Para aclarar las diferencias de los conceptos expuestos se describe un sencillo ejemplo en donde se supone que para una determinada hora se requieren 100 MW, en la tabla 1 se muestran 5 generadores con sus correspondientes limites de generación, sus ofertas marginales, y sus costos de arranque y parada.

Tabla 1: Bloque de ofertas

demanda del sistema = 100 MW				
	Pmin	Pmax	$/MW	CAP
genrador 1	5	50	10	0
genrador 2	5	40	20	0
genrador 3	0	10	65	50
genrador 4	5	60	30	1800

En la tabla 2, se puede observar el resultado de la asignación BCM que tiene en cuenta la liquidación tipo PAB.

Tabla 2 Resultado de la minimización BCM

	P(MW)	mo($/MW)	P*mo	(P*mo)*CAP
genrador 1	50	10	500	500
genrador 2	40	20	800	800
genrador 3	10	65	650	700
genrador 4	0	30	0	0
			Costo total	**$ 2.000,00**

En la tabla 3, se puede observar el resultado de la asignación PCM que tiene en cuenta la liquidación tipo PSMP.

Tabla 3: Resultado de la minimización PCM

Oferta más costosa seleccionada (π) = $ 30				
	P(MW)	**mo($/MW)**	**P*π**	**(P*π)*CAP**
genrador 1	50	10	500	500
genrador 2	40	20	800	800
genrador 3	0	65	0	0
genrador 4	10	30	300	2100
			Costo total	**$ 3.400,00**

2.3 SISTEMA DE LIQUIDACIÓN ADOPTADO MUNDIALMENTE (PAB Ó PSMP).

Según el ejemplo anterior se podría determinar que el esquema de liquidación más eficiente es el PAB ya que el resultado de su asignación fue el más económico, pero esto se debe al tipo de ejemplo que se utilizo. En la realidad al utilizar el tipo de liquidación PAB influye en las ofertas marginales que envían los participantes, ya que estos al tener la certeza de que serán liquidados según su oferta marginal, los incentiva a enviar ofertas marginales altas para así obtener mejor utilidad. Pasa totalmente lo contrario al utilizar el tipo de liquidación PSMP, ya que el participante sabe que no importa la oferta marginal que envíe, este será liquidado al precio de la oferta más costosa, por lo cual en su afán de finalizar dentro de la asignación final envía una oferta marginal lo más pequeña posible, que por lo general es muy cercana a su costo marginal. Esto genera al momento de asignar, menores pagos en comparación al método de liquidación PAB. Razón por la cual se adoptó en la mayoría de los países el sistema de liquidación PSMP.

Por otro lado la dinámica de los participantes del mercado a establecer sus objetivos y decisiones estratégicas dependiendo de las decisiones de los demás participantes, se conoce como teoría de juegos[4],y sirve como base para el estudio de los comportamientos estratégicos de los participantes. En [4] utilizando el Equilibrio de Nash[5] para el concepto de la solución de los juegos, se obtienen resultados de pruebas realizadas en donde se demuestran que los pagos incurridos la minimizar liquidando PAB son mayores en comparación a la liquidación PSMP.

2.4 TEORÍA DE SUBASTAS

La teoría de las subastas es una rama de la teoría de juegos que estudia el comportamiento de los participantes en las subastas, así como diversas posibilidades o tipos de estos mecanismos.

La subasta es la negociación de un mecanismo económico definido por una serie de reglas para especificar la forma de determinar un ganador o ganadores y

[4]La Teoría de Juegos es una teoría matemática que estudia las características generales de situaciones competitivas de manera formal y abstracta. Además, otorga una importancia especial a los procesos de toma de decisiones de los adversarios

[5] Los equilibrios de Nash (definidos por John Forbes Nash) forman parte de la teoría de juegos y son muy empleados en la economía, se definen como una manera de obtener una estrategia óptima para juegos que involucren a dos o más jugadores. La idea que se persigue con este tipo de equilibrios es verificar un conjunto de estrategias, por las cuales ningún jugador se beneficie cambiando su estrategia mientras los otros no cambien la suya. Este concepto hizo su aparición por primera vez cuando Nash lo enunció en su conferencia “Non CooperativeGames” de 1950.

cuánto se les debe pagar. Una característica notable de las subastas es la presencia de información asimétrica, lo que hace que la caracterización de este mecanismo se torne necesaria, ya que los diferentes tipos de subastas pueden conducir a conclusiones diferentes.

Las subastas pueden ser definidas en cuanto a su naturaleza (oferta, demanda, o doble), la forma en cómo se presentan las ofertas (abierta o cerrada) y el método de determinación del precio de cierre (precio de primera o segunda).

2.4.1 NATURALEZA DE LAS SUBASTAS

Es determinado por el papel desempeñado por los diferentes grupos de participantes (compradores y vendedores) en la subasta. Esta característica le permite diferenciar las subastas como oferta, demanda, o doble.

Oferta: Los vendedores ofrecen un bien que el comprador tiene la intención de comprar al precio más bajo. El precio de la subasta es determinado por los vendedores. El demandante puede fijar un precio máximo, por encima del cual las mercancías no se compran, por ultimo gana el participante que hace la oferta más baja.

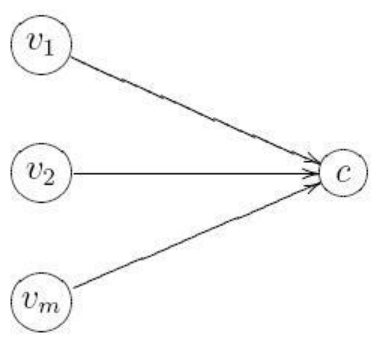


Ilustración 1: Subasta Tipo Oferta

Demanda: Donde los compradores hacen una oferta para comprar un bien o servicio que el vendedor tiene la intención de ofrecer al mejor postor. Gana el participante que hace la oferta más alta de la demanda, siempre que su oferta sea mayor que el precio mínimo estipulado por el vendedor.

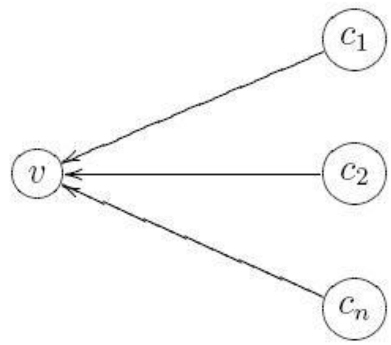


Ilustración 2: Subasta Tipo Demanda

Doble: Donde son necesarios varios vendedores y compradores al mismo tiempo. El precio de cierre está en función de las reglas de las subasta (ya sea pago al precio de oferta o pago a la oferta más costosa aceptada en la subasta). La subasta doble se caracteriza por la presencia o ausencia de agentes identificados, es decir, los participantes pueden negociar entre sí o través de un subastador.

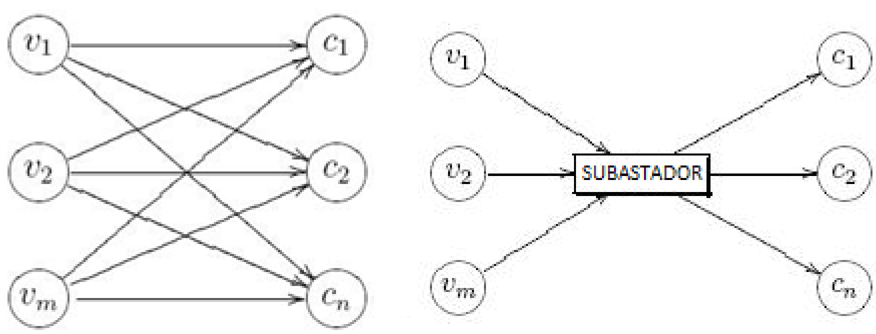


Ilustración 3: Subastas Tipo Doble

2.4.2 Formas de subasta

La forma de una subasta establece la característica que determina cómo se hacen las pujas, estas se dividen en abierta o cerrada.

Abierta: el precio de los bienes subastados se determina a través de un proceso dinámico de creación de la oferta ganadora. Este proceso dinámico se puede producir de forma ascendente o descendente.

- Ascendente: en el que el precio es constantemente elevado por el subastador a petición de los ofertantes, los cuales van abandonando a medida que el precio se vuelve demasiado alto. Esto continúa hasta que queda sólo un postor que gana la subasta al último precio. Esta subasta permite que el valor de oportunidad de cada participante se enfrente a los demás. Un fuerte argumento en favor del uso de la subasta ascendente, es su simplicidad estratégica, ya que los participantes no requieren de complejas consideraciones para determinar su estrategia. Pero hay algunos aspectos negativos relacionados con este tipo de la subasta, uno de ellos es la naturaleza de la subasta en tiempo real, produciendo los costos de transacción relativamente altos en comparación con la subasta cerrada.
- Descendente: en el que el precio comienza a un nivel suficientemente alto para disuadir a todos los licitadores y se reduce progresivamente hasta que un ofertante indica que está dispuesto a comprar al precio actual. La subasta holandesa exige una evaluación del mercado y el valor de los bienes subastados.

Cerrada: En el que los concursantes ponen su oferta en un sobre cerrado y, todos al mismo tiempo, se la entregan al subastador. Los sobres se abren y gana el concursante con la puja más alta o baja (dependiendo de la naturaleza de la subasta), el pago se puede definir mediante reglas denominadas precios de cierre.

Precio de cierre: En una subasta, el precio de cierre se puede definir como uniformes o discriminatorio.

- Uniforme: Este es el tipo más comúnmente utilizado en la subasta para la venta de la electricidad. En una subasta uniforme, todos los ganadores se pagan el mismo precio, independientemente del valor de sus ofertas. El precio pagado a todas las unidades admitidas es igual al precio de oferta más costosa aceptada al finalizar la subasta.

- Discriminatorias: Cada participante ganador es pagado al monto de su oferta.

Con lo anterior el problema de la asignación energética está clasificado en la teoría de subastas, como una subasta de naturaleza doble con la presencia de un agente subastador, que en el sector eléctrico se le conoce como el operador del mercado, y su forma de subasta es cerrada con liquidación de precio uniforme. Como ya se mencionó en la sección 2.3, este precio uniforme se liquida según PSMP que trae menores costos que PAB.

2.5 DESCRIPCIÓN DEL PROBLEMA

El problema actual radica en que aunque se adopto el sistema de liquidación tipo PSMP mundialmente, la asignación se sigue realizando mediante la función objetivo BCM (ecuación 2). Es decir se asigna teniendo en cuenta las ofertas de los generadores pero al final se hacen los pagos de una manera uniforme mediante PSMP. En la tabla 4 podemos ver el resultado de la asignación BCM teniendo en cuenta la liquidación PAB pero además se le agrega el pago real incurrido con el esquema de liquidación PSMP.

Tabla 4: Pago incurrido mediante el esquema de liquidación PSMP asignando mediante BCM

Oferta más costosa seleccionada (π) = $ 65						
	P(MW)	**mo($/MW)**	**P*mo**	**(P*mo)*CAP**	**P*π**	**(P*π)*CAP**
genrador 1	50	10	500	500	3250	3250
genrador 2	40	20	800	800	2600	2600
genrador 3	10	65	650	700	650	700
genrador 4	0	30	0	0	0	0
			Total PAB	**$ 2.000,00**	**Total PSMP**	**$ 6.550,00**

Se puede observar que aunque aparentemente el algoritmo busca un mínimo a pagar ($2.000), el pago incurrido real es mayor ($6.550) y este a su vez es mayor que el resultado obtenido utilizando el algoritmo que asigna según el esquema de liquidación PSMP ($3.400 tabla 3).

2.6 SITUACIÓN ACTUAL.

La razón por la cual en la actualidad se utiliza el modelo de asignación BCM pero se liquida según PSMP, se debe a que el modelo de asignación PCM es más complejo de resolver matemáticamente, lo cual lleva a tiempos de computación muy largos, ya que al ser π parte de la función objetivo, pero a su vez, el valor de π solo se puede conocer al final de la asignación, se encuentran con un problema de variables cruzadas lo cual hace de una estructura inseparable como se menciona en [2], lo cual no permite que se resuelva con las mismas técnicas utilizadas para la asignación tipo BCM, haciéndose necesaria la implementación de otras técnicas que permitan encontrar un solución factible y cercana a la óptima en tiempos más razonables.

2.7 COMPLEJIDAD COMPUTACIONAL[6] DEL PROBLEMA.

El problema de asignación PCM se clasifica dentro de los problemas de tipo NP-hard[7], debido a que el número de posibles soluciones teniendo en cuenta el número de participantes (I) y la cantidad de periodos de tiempo (T) es 2^{I*T}. Por lo tanto, a mayor número de participantes y periodos el problema se vuelve exponencialmente más complejo. El efecto que tiene la permutación de las

---

[6] La teoría de la complejidad computacional es la rama de la teoría de la computación que estudia, de manera teórica, la complejidad inherente a la resolución de un problema computable. Los recursos comúnmente estudiados son el tiempo (mediante una aproximación al número y tipo de pasos de ejecución de un algoritmo para resolver un problema) y el espacio (mediante una aproximación a la cantidad de memoria utilizada para resolver un problema)

[7] En teoría de la complejidad computacional, la clase de complejidad NP-hard (o NP-complejo, o NP-difícil) es el conjunto de los problemas de decisión que contiene los problemas H tales que todo problema L en NP puede ser transformado polinomialmente en H.

posibles asignaciones en relación con la cantidad de participantes y el número de periodos de tiempo se observa en la tabla 5.

Razón por la cual, en el presente trabajo se desarrolla el algoritmo meta-heurístico EPSO con el objetivo de resolver el problema de minimización de los pagos totales (TP) al problema de asignación PCM.

Tabla 5: Permutaciones posibles de acuerdo al número de participantes y periodos de tiempo.

I*T	Numero de asignaciones posibles
1	2
2	4
5	32
10	1024
15	32768
20	1048576
30	1073741824
100	1,26765E+30
200	1,60694E+60
400	2,5822E+120
600	4,1495E+180

2.8 CONCEPTO DE HEURÍSTICA Y META-HEURÍSTICA.

La heurística trata de métodos exploratorios o algoritmos para resolver problemas, las soluciones se buscan por aproximaciones sucesivas, para evaluar el progreso, hasta que se resuelva el problema.

Estos son métodos que, aunque se hace la operación en forma algorítmica, el progreso se logra mediante la evaluación puramente empírica de los resultados. Las técnicas heurísticas se usan por ejemplo en problemas cuando la complejidad del algoritmo de solución disponible es la función exponencial de algún parámetro, ya que cuando el valor de esta crece, el problema rápidamente se vuelve más complejo.

Una meta-heurística es un conjunto de conceptos que pueden ser utilizados para definir y mejorar los métodos heurísticos que se pueden aplicar a un amplio conjunto de problemas diferentes. En otras palabras, una meta-heurística puede verse como un marco general algorítmico que se puede aplicar a diferentes problemas de optimización con relativamente pocas modificaciones, para que sean adaptadas a un problema específico

Las Meta-heurísticas se puede clasificar en:

- Meta-heurísticos de búsqueda: atraviesan el espacio de búsqueda teniendo en cuenta, básicamente, la "vecindad" de la solución a la mano, que se define como el conjunto de soluciones que se pueden obtener de la aplicación de cualquier operador a la solución actual.
- Meta-heurísticas de construcción: definen meticulosamente el valor de cada componente de la solución.
- Meta-heurística de relajación: simplifican el problema (creando un problema relajado) y el uso de la solución como una guía para el problema original.
- Meta-heurísticas de evolución: hacen frente a una población de soluciones que se desarrolla principalmente a través de la interacción entre sus elementos

2.8.1 Búsqueda tabú. (TS)

Búsqueda tabú (TS) consiste en un procedimiento meta-heurístico utilizado para gestionar algoritmos heurísticos que realiza una búsqueda local, proporcionando medios para evitar quedar atrapado en soluciones óptimas locales. El nombre tabú se relaciona con el hecho de que, para evitar volver a examinar ciertas áreas del espacio de búsqueda que ya han sido explorados, el algoritmo convierte a estas áreas en tabú (o prohibidas). Esto significa que durante un cierto período de tiempo (tenencia tabú), la búsqueda no tendrá en cuenta la exanimación de las alternativas que contiene rasgos que caracterizan a la solución de los puntos pertenecientes a la superficie declarada tabú.

TS fue propuesto originalmente por Fred Glover en la década de 1980 y desde entonces ha sido aplicado con éxito a una serie de problemas complejos en ciencia e ingeniería. TS se desarrolló a partir de conceptos originalmente utilizados en inteligencia artificial. A diferencia de otros enfoques combinatorios como el algoritmo genético y recocido simulado, su origen no está relacionado con la optimización de procesos biológicos o físicos, búsqueda tabú explora el espacio de soluciones de una forma más agresiva. El algoritmos se inicializan con una configuración (o un conjunto de configuraciones, cuando la búsqueda se realiza en paralelo), y a partir de esta se define una estructura de vecindad, un nuevo movimiento se hace entonces a partir de la mejor configuración en este vecindad (es decir, en un problema de minimización, el algoritmo cambia a la configuración que presenta el menor costo). Normalmente, sólo los vecinos más prometedores son evaluados, de lo contrario el problema podría llegar a ser intratable. Por otro lado los movimientos a configuraciones con mayores costos son permitidos dando así la capacidad de moverse fuera de un óptimo local.

En general, los algoritmos TS resuelven problemas formulados de la siguiente manera:

$$Min\ f(x) \qquad (5)$$
$$Sujeto\ a.\ x \in \mathrm{X}$$

Donde x es una configuración (o variable de decisión), f, es la función objetivo, y X es el espacio de búsqueda. TS resuelve un problema del tipo de la Ecuación (5), en primer lugar aplicando una heurística de búsqueda local en el que, dada una configuración de x (una solución), el vecindario de x se define como el conjunto de todas las configuraciones x’ $\in N(x)$. Las condiciones requeridas para que x’ sea un vecino de x define la estructura de la vecindad. El algoritmo de búsqueda local encuentra él nuevo movimiento, el cual se determina mediante la selección de cual x’ presenta el mayor decremento en la función objetivo. La repetición de este procedimiento conduce finalmente a una solución óptima local. La búsqueda tabú difiere del algoritmo heurístico de búsqueda local simplemente en por lo menos en dos aspectos esenciales:

1. Son permitido movimientos que conducen a configuraciones para las cuales la función objetivo es mayor que la solución actual.
2. El vecindario de x, es decir, N (x), no es estático, sino que puede cambiar tanto en tamaño como en estructura. Un vecindario modificado N (x) se determina de diferentes maneras, por ejemplo:

- Uso de una lista tabú, que contiene los atributos de las configuraciones que están prohibidas.
- Uso de estrategias para reducir el tamaño de la vecindad con el fin de acelerar la búsqueda local.
- Redefinición de N (x) durante el proceso de optimización, lo que normalmente se hace con el fin de beneficiarse de las propiedades específicas del problema.

2.8.2 Estrategias evolutivas (EE).

Las estrategias evolutivas (Estrategias de evolución - ES), son una clase de algoritmos evolutivos utilizados principalmente para resolver problemas de optimización. Fueron desarrollados por Rechenberg, Schwefel y Bienert en la Universidad Técnica de Berlín alrededor de 1964. En un principio se trató de problemas de optimización en mecánica de fluidos, y luego comenzó a tratar la optimización de funciones de manera más general.

Los diferentes tipos de EE suelen denotarse como (a+b)-EE, donde indica que este algoritmo trabaja con “a” padres, y en cada iteración genera “b” hijos y en esencia son métodos estocásticos con pasos adaptativo, por lo que EE puede definirse como algoritmos evolutivos enfocados hacia la optimización, teniendo como características principales la utilización de una representación a través de vectores reales, una selección determinística y operadores genéticos específicos de cruce y mutación. Además, su objetivo fundamental consiste en encontrar el valor real de un vector de N dimensiones.

Las EEs pueden dividirse en dos tipos: Estrategias Evolutivas Simples y Estrategias Evolutivas Múltiples.

EEs Simples ó (1+1)-EE: Son consideradas como procedimientos estocásticos de optimización paramétrica con paso adaptativo donde se hace evolucionar un solo individuo usando únicamente la mutación como operador genético. Son relativamente sencillas, y se denominan también EEs de dos miembros. Debido a que evoluciona un solo individuo a la vez, no son consideradas estrictamente como métodos evolutivos. A pesar de ser muy sencillas, son de gran utilidad práctica y han sido utilizadas, con algunas mejoras, para resolver problemas reales en diversas áreas. En si la regla de movimiento cada generación es realizada alterando al padre agregando un ruido Gamusino de esta forma.

$x_i^{t+1} = x_i^t + N(0, \sigma)$ (6)

Donde x_i^t es la el valor asumido de la variable i en la función objetivo en la iteración t y N(0, σ) es una variable aleatoria con distribución de Gauss (media 0 y varianza σ). Esta varianza puede variar en el transcurso de las iteraciones del algoritmo (lo cual garantiza la convergencia).

Una forma sencilla de convergencia es hacer que σ varié dependiendo de cómo se está comportando el algoritmo, es decir que si los éxitos por mutación son altos se buscaría intensificar la búsqueda por cada iteración, lo cual se logra disminuyendo el valor numérico de σ, pero si por el contrario no se están teniendo buenos resultados por mutación, se pretendería diversificar la búsqueda, lo cual se logra aumentando el valor de σ.

Para optimizar la velocidad de convergencia, Rechenberg (1973) propone una regla llamada la regla "de éxito 1/5". Que dice: la razón entre mutaciones exitosas y el total de mutaciones es 1/5. Si es mayor, se incrementa la desviación estándar (dividiendo entre 0.817), si es menor, se disminuye (multiplicando por 0.817). Por último se determina un criterio de parada, ya sea un número máximo de iteraciones en total, o un número máximo de iteraciones sin mejora en la función objetivo, entre otros.

EEs Múltiples ó (μ+λ)-EE: Surgen como respuesta a las debilidades de las EEs simples, las cuales tienden a converger hacia subóptimos. En las EEs múltiples existen múltiples individuos μ (población), y se producen en cada generación varios nuevos individuos λ, usando tanto mutación como cruce. Se usa normalmente el cruce promedio, el cual genera un único descendiente de dos padres, promediando los valores de estos. En cuanto a los criterios de reemplazo, siempre se usa un esquema determinístico, pudiéndose utilizar una estrategia de inserción o de inclusión.

2.8.3 Optimización de Enjambre de Partículas. (Particle Swarm Optimization PSO.)

Eberhart y Kennedy desarrollaron (PSO), basándose en la analogía de las características del movimiento colectivo de bandadas de pájaros, cardúmenes de peces, o enjambre de abejas. Cada uno de estos enfoques, diferentes unos de otros, han traído una nueva perspectiva a la optimización, llamándola así la inteligencia de enjambre, cuyo comportamiento grupal es complicado, sin importar que las reglas de movimiento de cada individuo (agente) sean relativamente simples.

PSO es una metodología nueva en computación evolutiva, que es similar al Algoritmo Genético (GA) en donde el sistema se inicia con una población de soluciones al azar a diferencia de otros algoritmos, sin embargo, cada solución potencial (llamada partícula o agente) no convergen a un punto mediante evolución si no por el contrario mediante movimientos. Al principio PSO tenía como objetivo tratar los problemas de optimización no lineal con variables continuas, luego PSO se amplió para manejar los problemas de optimización combinatoria y con ambas variables, discretas y continuas, teniendo así la capacidad de tratar eficientemente la programación no lineal entera mixta (MINLP).

Un algoritmo PSO consiste en un proceso iterativo y estocástico que opera sobre un cúmulo de partículas. La posición de cada partícula representa una solución potencial al problema que se está resolviendo. Generalmente, una partícula i está compuesta por cuatro vectores y tres valores de fitness (valor resultado de determinada configuración en la función objetivo):

- El vector $S_i = (s_{i1}, s_{i2}, \ldots, s_{in})$, almacena la posición actual (localización) de la partícula i en el espacio de búsqueda.
- El vector $pBest_i = (p_{i1}, p_{i2}, \ldots, p_{in})$, almacena la posición de la mejor solución encontrada por la partícula i hasta el momento.
- El vector $gBest = (g_{i1}, g_{i2}, \ldots, g_{in})$, almacena la posición de la mejor solución encontrada por el enjambre hasta el momento.
- El vector de velocidad $V_i = (V_{i1}, V_{i2}, \ldots, V_{in})$, almacena el gradiente (dirección) según el cual se moverá la partícula.

- El valor de fitness_S_i almacena el valor de adecuación de la solución actual (vector xi).
- El valor de fitness de $pBest_i$ almacena el valor de adecuación de la mejor solución local encontrada hasta el momento (vector $pBest_i$).
- El valor de fitness de $gBest$ almacena el valor de adecuación de la mejor solución global encontrada hasta el momento (vector $gBest$).

Con esto cada agente intenta modificar su posición. Esta modificación o regla de movimiento (de una iteración k a una iteración k+1) puede ser representada mediante la ecuación (7) y a su vez la velocidad de cada agente puede ser modificada mediante la ecuación (8):

$s_i^{k+1} = s_i^k + v_i^{k+1}$ (7).

$v_i^{k+1} = wv_i^k + c_1 rand_1 * \left(pbest_i - s_i^k\right) + c_2 rand_2 * \left(gbest - s_i^k\right)$(8).

Donde, w es la función de ponderación (peso), $c_j$son los coeficientes de ponderación y rand es un número aleatorio entre 0 y 1, La velocidad de un agente se puede cambiar utilizando tres vectores que se muestran en la parte derecha de la ecuación (8). El primero es el vector de la velocidad actual del agente y se le conoce como inercia ya que la partícula tiende a moverse en la misma dirección. El segundo se le conoce como memoria, ya que el agente tiende a devolverse a la mejor posición que el mismo ha obtenido y el tercero se conoce como cooperación debido a que todos los agentes tienden a ir a la mejor solución que ha encontrado el enjambre, estos dos últimos vectores se utilizan para cambiar la velocidad del agente. Sin el segundo y tercer vector, el agente seguiría en la misma dirección hasta llegar al límite del espacio de búsqueda. Es decir que con los dos últimos términos el agente trata de explorar nuevas áreas.

Además, la función de ponderación w, que está atada a la inercia del agente, suele estar limitada a un cierto valor máximo y mínimo.

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} * iter \quad (9).$$

Donde w_{max} es el peso inicial, w_{min} es el peso final, $iter_{max}$ es el número máximo de iteraciones, e iter es el número de iteración actual. Como también se puede observar, la ecuación (9) se encarga de disminuir la velocidad de los agentes a medida que transcurren las iteraciones en la búsqueda de gbest, por lo tanto, al inicio del procedimiento de búsqueda, la diversificación es muy ponderada, mientras que al final del procedimiento de búsqueda la intensificación es muy ponderada.

Por otro lado, existen estudios [5] realizados en sobre la influencia de los ratios de aprendizaje (c_j) en los que se recomienda valores de $c_1 = c_2 = 2$.

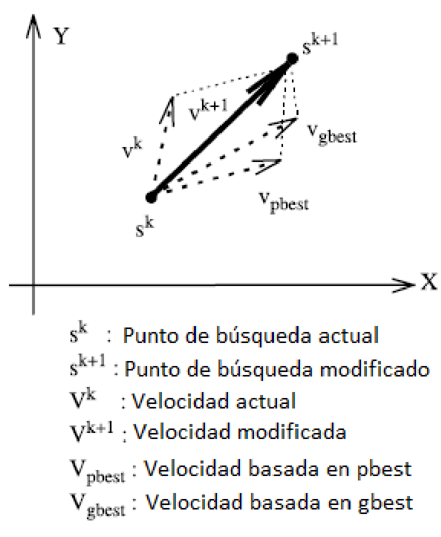


Ilustración 4 Concepto de la modificación de un punto de búsqueda en PSO. Fuente: Kwang Y. Lee and Mohamed A. El-Sharkawi. "MODERN HEURISTIC OPTIMIZATION TECHNIQUES".Theory And ApplicationsToPowerSystems

La ilustración 4 (para un ejemplo en el caso bidimensional) muestra un concepto de modificación de un punto de la búsqueda en PSO y como cada agente cambia su posición actual mediante la integración de los vectores (inercia, memoria y cooperación). El diagrama de flujo general de la PSO se describe como sigue:

Paso 1. Generación de las condiciones iniciales de cada agente.
La búsqueda inicial de puntos (s_i^0) y velocidades (v_i^0) de cada agente son por lo general de forma aleatoria dentro del rango permitido. El punto de la búsqueda actual se establece en pbest para cada agente. El mejor valor de pbest se establece en gbest, y el número del agente con el mejor valor es almacenado.

Paso 2. Modificación de cada punto de la búsqueda.
El punto de la búsqueda actual de cada agente se cambia usando las ecuaciones (7), (8) y (9).

Paso 3. Evaluación de los nuevos puntos de búsqueda de cada agente.
Se calcula el valor de la función objetivo para cada agente. Si el valor es mejor que el pbest actual del agente, el valor pbest se remplazara por el valor actual. Si el mejor valor de pbest es mejor que el gbest actual, gbest se sustituye por el mejor valor y el número del agente con el mejor valor es almacenado.

Paso 4. Comprobación de la condición de salida.
Si el número de iteración actual alcanza el máximo número de iteración predeterminado, el proceso de búsqueda termina. De lo contrario, el proceso avanza al paso 2.

2.8.4 Optimización Evolutiva de Enjambre De Partículas (Evolutionary Particle Swarm Optimization EPSO)

EPSO es un algoritmo de optimización meta-heurístico el cual se caracteriza por combinar los planteamientos de los algoritmos Estrategias Evolutivas (EE) y Optimización de Enjambre de Partículas (PSO). Por lo tanto EPSO esta inspiración en la biología de la evolución de las especies mediante una selección Darwinística y los comportamientos de distintos grupos de especies animales, donde trata de imitar el movimiento colectivo o social de bandadas de aves, cardúmenes de peces o enjambres de abejas como un conjunto de partículas que evoluciona en el espacio de búsqueda motivado por tres factores: inercia, memoria y cooperación.

El mecanismo del algoritmo EPSO se puede describir de la siguiente manera: para una iteración dada existe un conjunto de soluciones o alternativas denominadas agentes que al igual que en PSO cada uno de estos esta compuestos de los siguientes atributos.

- El vector $S_i = (s_{i1}, s_{i2}, \dots, s_{in})$, almacena la posición actual (localización) de la partícula i en el espacio de búsqueda.
- El vector $pBest_i = (p_{i1}, p_{i2}, \dots, p_{in})$, almacena la posición de la mejor solución encontrada por la partícula i hasta el momento.
- El vector $gBest = (g_{i1}, g_{i2}, \dots, g_{in})$, almacena la posición de la mejor solución encontrada por el enjambre hasta el momento.
- El vector de velocidad $V_i = (V_{i1}, V_{i2}, \dots, V_{in})$, almacena el gradiente (dirección) según el cual se moverá la partícula.
- El valor de fitness_S_i almacena el valor de adecuación de la solución actual (vector xi).
- El valor de fitness de $pBest_i$ almacena el valor de adecuación de la mejor solución local encontrada hasta el momento (vector $pBest_i$).
- El valor de fitness de $gBest$ almacena el valor de adecuación de la mejor solución global encontrada hasta el momento (vector $gBest$).

Cada partícula está definida por una posición en el espacio de búsqueda S_i y una velocidad V_i. En un momento dado, hay al menos una partícula que tiene la mejor posición en el espacio de búsqueda. La población de las partículas reconoce tal posición ($gBest$), entonces las partículas tienden a moverse en esa dirección, además cada partícula es atraída a su mejor posición anterior ($pBest_i$). Las partículas se reproducen y evolucionan a lo largo de un número de generaciones según los siguientes pasos.

REPLICACIÓN: cada partícula es replicada un número de r veces, dando lugar a nuevas partículas iguales.

MUTACIÓN: los parámetros estratégicos (w) que afectan el movimiento de las partículas son mutados.

REPRODUCCIÓN: de cada partícula se genera un sucesor según la regla de movimiento de la partícula.

EVALUACIÓN: cada sucesor será evaluado con una función objetivo.

SELECCIÓN: por un torneo estocástico u otro proceso de selección, las mejores partículas sobreviven para formar una nueva generación.

La regla de movimiento de las partículas (de una iteración k a la iteración k+1) es la siguiente:

$s_i^{k+1} = s_i^k + v_i^{k+1}$ (10).

$v_i^{k+1} = w_{i0}^* v_i^k + w_{i1}^* \left(pbest_i - s_i^k\right) + w_{i2}^* (gbest^* - s_i^k)$ (11).

En la ecuación (11) el símbolo * significa que esos parámetros presentaran una evolución producto del proceso de mutación. En cuanto a los pesos (w) la regla de mutación es:

$w_{ik}^* = w_{ik} + \tau N(0.1)$

Donde τ es el parámetro de aprendizaje, fijado externamente, el cual controla la amplitud de las mutaciones, N(0,1) es una variable aleatoria con distribución de Gauss (media 0 y varianza 1) y w_{ik} varía a medida que progresa el algoritmo de la siguiente manera:

$$w_{ik} = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} * iter \quad (12).$$

Además la mejor solución global es mutada de la siguiente manera.

$$gbest^* = gbest + w_{i3}^* N(0.1). \quad (13).$$

Donde w_{i3}^* también es un parámetro de aprendizaje y este suele llamársele también el cuarto parámetro estratégico. Este controla la amplitud del vecindario de gbest donde es más probable encontrar la mejor solución global o al menos una mejor solución que el gbest actual. El peso w_{i3}^* también es mutado acorde a la regla de mutación general descrita anteriormente.

La estructura de la ecuaciones (10) y (11) son muy parecidas a la regala de movimiento de PSO (ecuaciones (7) y (8)), pero con respecto a EPSO no sólo se ve en el comportamiento evolutivo de las partículas, sino también en los pesos que afectan al movimiento de estas a medida que se avanza en el espacio de búsqueda, es decir, los parámetros de ponderación c_i en PSO que se definían como $c_1 = c_2 = 2,$ en EPSO son remplazados por w_{ik}^* mutación de w_{ik} que se caracteriza no por ser constantes sino por su carácter auto-adaptivos, permitiendo ajustar automáticamente sus parámetros o comportamientos en respuesta a la manera en que progresa la solución del problema. EPSO contiene dos mecanismos (evolutivo y auto-adaptivo) actuando en secuencia, cada uno con su propia probabilidad de producir no solo mejores individuos, sino también un promedio grupal mejor. La evolución permite que en cada recombinación se induzca un movimiento en dirección al óptimo, entonces, la selección que actúa sobre una generación que es en promedio mejor que la precedente, produce una nueva generación que será mejor que la primera generación de partículas. Que sea auto-adaptivo suma otro interés al método, esta evita en gran medida la necesidad de un ajuste fino de los parámetros iniciales del algoritmo, porque se espera que el procedimiento aprenda (en el sentido evolucionario) las características del espacio de búsqueda y corrija (autoajuste) los pesos en orden a generar una adecuada tasa de progreso hacia el óptimo. Esta característica da robustez al modelo. Esto significa, que independientemente de los valores iniciales, el algoritmo converge al óptimo o un resultado próximo.

En resumen EPSO se mueve de manera similar que PSO mediante la suma de los vectores de inercia, memoria y cooperación pero como lo muestra la ilustración 5, cada uno de estos vectores en EPSO es ponderado de una forma evolutiva y auto-adaptativa.

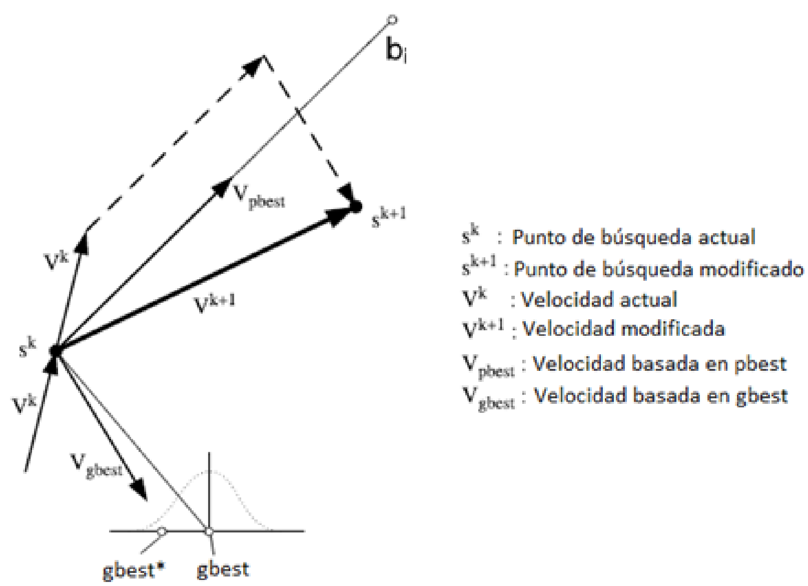


Ilustración 5 Concepto de la modificación de un punto de la búsqueda en EPSO. Fuente: Rolando M. Pringles, Vladimiro Miranda y Francisco Garcés. Expansión Óptima del Sistema de Transporte Implementando EPSO

3. DISEÑO DE LA SOLUCIÓN.

3.1 DESCRIPCIÓN DE LA ASIGNACIÓN PCM

El problema de Minimización de los costos finales (Payment Cost Minimization-PCM) consiste en determinar los generadores y los periodos de tiempo que suplen la demanda energética en un día determinado.

considerando un mercado de energía con *I* participantes o generadores (donde *i=1,2,…,I)*, los cuales son necesarios para suplir la demanda energética de un serie de periodos (L_t) donde L es la demanda en MW y *t* corresponde al intervalo *(1 ≤ t ≤ T)* (por lo general T=24 refiriéndose a las 24 horas del día), el objetivo de la asignación es la selección de generadores $\mu_{i,t}$ (donde si $\mu_{i,t} = 1$ indica que el generador i es seleccionado para la hora t y si $\mu_{i,t} = 0$ lo contrario) y sus correspondientes niveles de potencia ($P_{i,t}$), para suplir la demanda de todos los periodos de tiempo (L_t).

Para ello, cada uno de los participantes presentan un solo bloque de ofertas**,** el cual contiene la oferta marginal $mo_{i,t}$ (precio de cada MW generado), sus niveles de generación de potencia máxima y mínima para todos los periodos de tiempo ($P_{i,t}^{min}$ y $P_{i,t}^{max}$), y sus costos de arranque y parada CAP_i.

Para determinar el costo que se incurre a un generador i en una hora t, y teniendo en cuenta como se aclaro en la sección (3.3) que el criterio de asignación PCM se basa en la oferta marginal del mercado, se hace necesario saber cuál fue la oferta más costosa aceptada para el periodo de tiempo t (π_t).

$$\pi_t = \max\{(mo_{i,t} * \mu_{i,t})\} \ (14).$$

Además se debe tener en cuenta los costos de arranque y parada, pero estos solo son pagados únicamente si se obliga al generador encenderse nuevamente para la hora t, es decir que si en *t-1* el generador *i* estaba apagado *($\mu_{i,t-1}$ = 0)* y en t se decide que el generador i se encienda *($\mu_{i,t}$= 1)* esto obliga a pagar el costo de arranque y parada *CAP*, matemáticamente se puede ver con la siguiente expresión $\mu_{i,t} *(CAP_i * \neg\mu_{i,t-1})$ (donde $\neg\mu_{i,t-1}$es la negación de $\mu_{i,t-1}$). En resumen la expresión matemática del pago al generador i en un tiempo t está dada por:

$$[\,\pi_t * P_{i,t} + CAP_i * (\neg\mu_{t-1})] * \ \mu_{i,t}\,(15).$$

Por lo tanto el modelo asignación PCM tiene como objetivo la minimización de los pagos liquidados según PSMP para así suplir la demanda de cada uno de los periodos t y se formula mediante la función de Pago Tota (TP).

$$TP = \left\{ \sum_{t=1}^{T} \sum_{i=1}^{I} \left(\pi_t * P_{i,t} + CAP_i * \left(\neg\mu_{i,t-1}\right)\right) \mu_{i,t} \right\} \quad (16)$$

Donde:

- $\boldsymbol{TP}$ **=** Pago Total que incurriría el sistema
- $\boldsymbol{\pi_t}$**=** Precio de bolsa o “System Marginal Price”-SMP en el período t resultado de la subasta.
- $\boldsymbol{P^*_{i,t}}$ **=** potencia asignada por el operador del mercado, como resultado de la subasta de energía a la unidad de generación i en el periodo t
- $\boldsymbol{CAP_i}$ **=** Costo de Arranque y parada del generador i
- $\boldsymbol{\mu_{i,t}}$ **=** Valor de la función de encendido-apagado de la unidad i en el período t determinado por el operador del mercado [0,1]
- $\boldsymbol{\neg\mu_{i,t-1}}$**=** Negación del valor de la función de encendido-apagado de la unidad i en el período t-1 determinado por el operador del mercado [0,1]

3.1.1 Restricciones

Las restricciones a las que pueda estar sujeto el problema de minimización, dependen del grado de aproximación y detalle con el que se quiera modelar el mecanismo de la subasta. De acuerdo con los alcances de esta investigación, se estudiará un sistema en el que se consideran condiciones como la competencia perfecta, una demanda completamente inelástica y conocida, condiciones de trasmisión ideales (sin perdidas y sin restricciones de flujo por las líneas), y costos de arranque constantes y completamente compensados. Para este caso las restricciones asociadas a la función objetivo son las que se explican a continuación.).

3.1.1.1 Balance de potencia.

La demanda en cualquier sistema eléctrico de potencia no es la misma para cualquier hora, sino que va variando con respecto a las necesidades de los usuarios. Tal característica hace necesario hacer el balance oferta-demanda en los despachos de unidades de cada intervalo de tiempo. Al fin de cuentas, lo que resulta es una restricción para la función objetivo que puede modelarse matemáticamente con la siguiente ecuación:

$$L_t = \sum_{i=1}^{I} P_{i,t} \qquad \forall\ t = 1,2, \dots T \quad (17)$$

Donde L_t es la demanda requerida en el instante de tiempo t

3.1.1.2 Límites de generación máxima y mínima.

Por efectos de fabricación, características intrínsecas y clase de tecnología, cada unidad de generación cuenta con unos límites mínimos y máximos fuera de los cuales su operación no sería económicamente viable o seria físicamente imposible, lo que se convierte en la siguiente restricción para la función objetivo:

$$\begin{matrix} P_{i,t} = 0, & si \quad \mu_{i,t} = 0 \\ P_{i,t}^{min} \leq P_{i,t} \leq P_{i,t}^{max} & si \quad \mu_{i,t} = 0 \end{matrix} \quad (18)$$

Por lo tanto el modelo asignación PCM tiene como objetivo la minimización de los pagos liquidados según PSMP para así suplir la demanda en cada uno de los periodos t y este se formula mediante la función de Pago Tota (TP).

$$TP = \left\{ \sum_{t=1}^{T} \sum_{i=1}^{I} \left(\pi_t * P_{i,t} + CAP_i * \left(\neg \mu_{i,t-1} \right) \right) \mu_{i,t} \right\} \quad (19)$$

Sujeto a:

$$L_t = \sum_{i=1}^{I} P_{i,t} \quad \forall \ t = 1,2, \dots T$$

$$P_{i,t} = 0, \qquad si \quad \mu_{i,t} = 0 \qquad \forall \ i = 1,2, \dots I$$

$$P_{i,t}^{min} \leq P_{i,t} \leq P_{i,t}^{max} \qquad si \quad \mu_{i,t} = 1 \qquad \forall \ i = 1,2, \dots I$$

3.2 ESTRATEGIAS DE SOLUCIÓN

En el presente trabajo se pretende simplificar las iteraciones del meta-heurístico haciendo una suposición que está atada a una característica intrínseca del proceso de asignación, la cual consiste en que todos los generadores escogidos para suministrar la energía requerida para cierto instante de tiempo t siempre emitirán su potencia máxima, excepto un solo generador el cual emitirá lo restante para suplir los requerimientos.

Se describe un ejemplo que permite aclarar los conceptos expuestos; suponiendo que para una determinada hora se requieren 100 MW y se tienen 5 generadores con sus correspondientes límites de generación

Tabla 6: Ejemplo de asignación de energía para una sola hora

Demanda = 100 MW		
	P min (MW)	P max (MW)
GENERADOR 1	5	40
GENERADOR 2	3	35
GENERADOR 3	5	55
GENERADOR 4	15	20
GENERADOR 5	2	20

Una asignación factible sin importar si es la óptima, podría ser la potencia máxima de los generadores 3, 4 y 5, los cuales aportaría en total 95 MW, y en cuanto a los 5 MW faltantes para cumplir los requerimientos de la demanda se escoge cualquiera de los otros dos generadores faltantes.

Ahora bien, ya que siempre se utiliza la potencia máxima de los generadores excepto uno, se podría eliminar las variables de decisión de potencia $\mathrm{P_{i,t}}$ de la función objetivo y remplazarla por $\mathrm{P_{i,t}^{max}}$, la cual es una constante intrínseca de cada generador. Al efectuar este cambio también se estará asegurando el cumplimiento de la restricción que controla la potencia asignada $(\mathrm{P_{i,t}})$, para que no exceda los limites de generación (es decir $P_{i,t}^{min} \leq P_{i,t} \leq P_{i,t}^{max}$), esto debido a que con esta suposición la potencia asignada siempre será la máxima de cada generador si $\mu_{i,t} = 1$ o en su defecto cero si $\mu_{i,t} = 0$. Una vez el meta-heurístico termina una iteración, ajusta automáticamente las potencias para así cumplir la demanda de marea exacta y evalúa la solución actual teniendo en cuenta todas las restricciones del criterio de asignación PCM.

Como se puede observar las únicas variables de decisión son $\mu_{i,t}$, por lo que el problema que se tenía anteriormente de programación entera mixta (ecuación 19), ahora solo se tiene un problema de programación binaria.

3.3 DESCRIPCIÓN DETALLADA DEL ALGORITMO EPSO –EVOLUTIONARY SELF-ADAPTING PSO PARA LA PROGRAMACIÓN BINARIA.

En un espacio binario, una partícula puede parecer moverse cerca o lejos al cambiar varios números de sus coordenadas binarias o bits, además la velocidad que tiene esta partícula puede ser descrita por el numero de bits cambiados por iteración, es por esto que una partícula con cero bits cambiados no se mueve, mientras que esta se mueve lo más lejos posible al cambiar por completo todas sus coordenadas binarias.

La solución a este dilema es la definición de la trayectoria, velocidad y movimiento. Los cuales se pueden definir en términos de las probabilidades de cambio para que una variable cambie de un estado al otro, es por esto que en el algoritmo binario una partícula se mueve en cada una de sus coordenadas en un espacio restringido de [0 1] dependiendo de su velocidad, donde cada V representa la probabilidad de que una de las variables de decisión de la partícula tome determinado valor. En otras palabras, si para una partícula la velocidad de una de sus variable de decisión es V=0.20, entonces hay una probabilidad del veinte por ciento de que esta variable tome el valor de 1, y un ochenta por ciento de probabilidad de tomar el valor de 0.

Pero la velocidad en el algoritmo EPSO no está representada como una probabilidad. En la sección 3.7.4, se muestra la velocidad se rige bajo la siguiente ecuación.

$$v_i^{k+1} = w_{i0}^* v_i^k + w_{i1}^*\left(pbest_i - s_i^k\right) + w_{i2}^*(gbest^* - s_i^k) \qquad (21)$$

Es por esto que se utiliza la función sigmoide para convertir la expresión de la velocidad en una probabilidad.

$$K_{(v_i^k)} = 1/(1 + e^{-(v_i^{k-1}})\quad (22)$$

Y por último la regla de movimiento de la partícula de una iteración k a una iteración k+1 está dada por la siguiente ecuación:

$$s_i^{k+1} = \begin{matrix} 1\ si\ \ rand() \leq \mathrm{K}_{(v_i^k)} \\ 0\ \ si\ \ rand() > K_{(v_i^k)} \end{matrix}\quad (22)$$

En resumen, el diagrama de flujo de EPSO binario es dado por la ilustración 6.

3.4 APLICACIÓN DEL ALGORITMO EPSO AL TIPO DE ASIGNACIÓN DE ENERGÍA (PCM).

La motivación de emplear EPSO es la buena calidad de las soluciones que se obtienen respecto a otros métodos Meta-heurísticos. En [6] los autores muestran la superioridad de EPSO frente al algoritmo clásico PSO en varios problemas de optimización de funciones de prueba (test functions). En [7] muestran una aplicación de EPSO a sistemas de potencia, más precisamente a un problema de minimización de pérdidas y control de tensión en redes de distribución. En el trabajo comparan la eficiencia de EPSO con respecto al algoritmo de Recocido Simulado y concluyen que EPSO encuentra una mejor solución en menos iteraciones y muestra mayor robustez en los resultados. En [8] los autores comparan a EPSO con tres variantes de algoritmos genéticos en el contexto de una simulación de mercado multienergético con una plataforma de agentes inteligentes, y definen un conjunto de agentes y se estudia uno en particular (distribuidores), donde observan que en todos los escenarios simulados, los agentes distribuidores equipados con el algoritmo EPSO se comportan de mejor manera que los agentes que poseen cualquiera de las variantes de algoritmos genético.

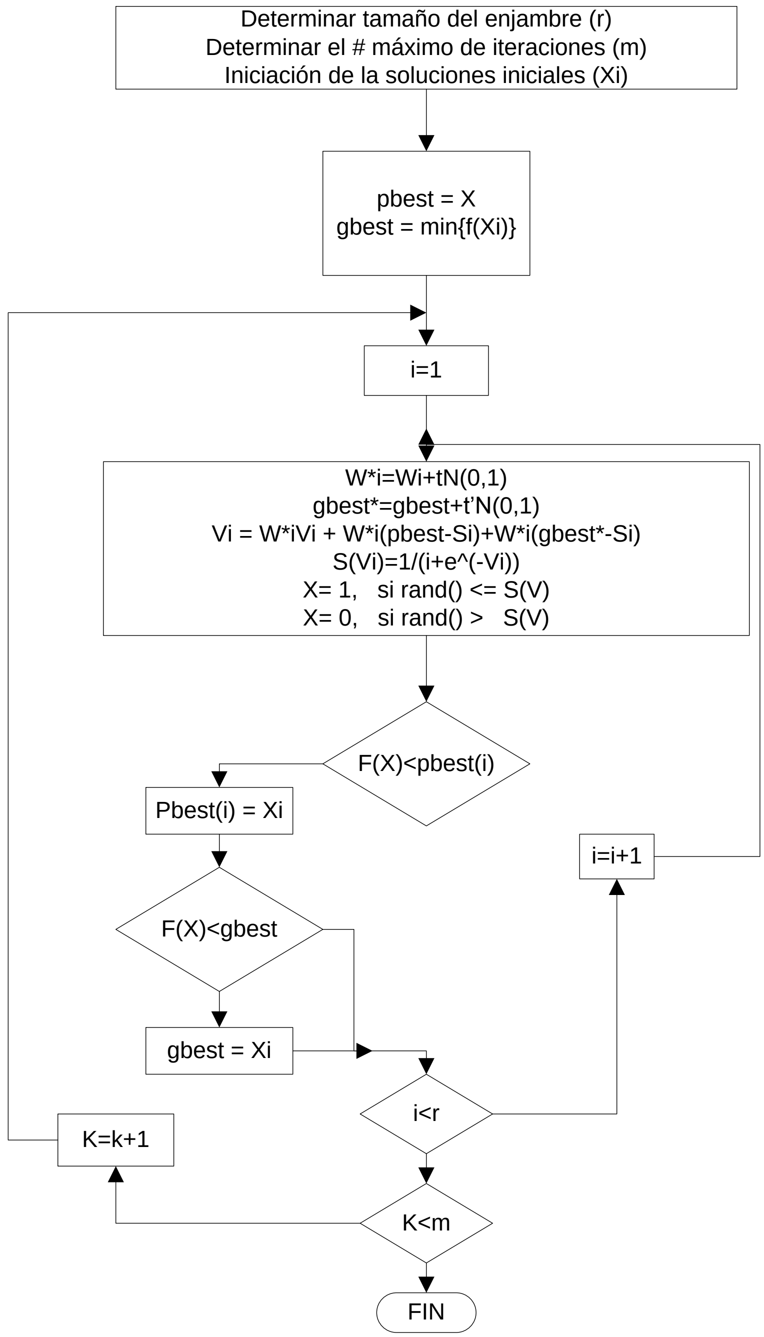


Ilustración 6 Diagrama de flujo algoritmo EPSO binario, Fuente Autor

En [9] el autor presenta un problema de despacho clásico el cual no considera la red de transmisión. Compara los resultados de EPSO con los de un método de programación no lineal clásico, y con los resultados de dos modelos de programación evolutiva (uno adaptivo y otro no adaptivo). En todos los casos muestra que EPSO arroja mejores resultados y con menores desviaciones, es decir, presenta mayor robustez.

4. RESULTADOS NUMÉRICOS.

En esta sección se muestran un ejemplo que permite realizar una comparación numérica un tanto más detallada entre los resultados numéricos obtenidos en [3]. Para su cálculo, fueron implementados los algoritmos Búsqueda Tabu (TS), Optimización de Enjambre de Partículas (PSO) y por ultimo Optimización Evolutiva de Enjambre de Partículas (EPSO) en la función objetivo descrita en la ecuación (20) de la sección 4.2, en el sistema de programación MATLAB en un computador personal Intel centrino-Core 2duo– 1.83 GHz.

En ambos ejemplos se consideran condiciones de competencia perfecta, una demanda de energía inelástica y conocida a priori, se desprecian los servicios complementarios, las restricciones del sistema de transmisión y las pérdidas, y se asume que se utiliza un esquema de liquidación uniforme PSMP.

Ejemplo: Este ejemplo considera un sistema con 25 ofertas de diferentes participantes y 24 escenarios de demanda. Cada situación de demanda se presenta en la tabla 7. De igual forma el precio de las ofertas de costo marginal, las restricciones de generación máxima y mínima de cada unidad, así como los costos de arranque-parada para cada unidad, se muestran en la tabla 8

Tabla 7: Escenarios de demanda ejemplo 1

HORA	DEMANDA DEL SISTEMA	HORA	DEMANDA DEL SITEMA	HORA	DEMANDA DEL SITEMA
1	30	13	455	9	55
2	34	14	350	10	57
3	35	15	300	11	58
4	37	16	200	12	59
5	40	17	350	21	190
6	42	18	320	22	180
7	45	19	240	23	170
8	47	20	200	24	160

Tabla 8: Bloque de ofertas ejemplo 1

Hora 1-24				
Oferta	Min MW	Max MW	$/MW	CAP
1	100	455	30	1200
2	60	350	34	1150
3	50	300	35	1100
4	30	200	37	1000
5	40	350	40	350
6	40	320	42	350
7	30	240	45	300
8	30	200	47	300
9	20	190	55	180
10	20	180	57	175
11	20	170	58	160
12	20	160	59	250
13	20	150	60	200
14	20	140	62	150
15	20	130	63	180
16	20	125	65	140
17	20	120	66	140
18	20	115	68	140
19	20	110	70	160
20	20	120	75	250
21	20	115	78	250
22	20	110	80	300
23	20	100	90	50
24	15	90	93	50
25	10	80	95	40

Como se hizo mención anteriormente, este ejemplo se corrió con tres diferentes algoritmos (TS, PSO y EPSO), en la tabla 9 se pueden observar el costo total incurrido después de 450 iteraciones en los tres algoritmos, al igual que en la ilustración 7 se puede observar el comportamiento de cada uno durante todas la iteraciones, con respecto al costo total incurrido en la asignación.

Tabla 9 Comparación de los resultado obtenidos por los algoritmos TS, PSO y EPSO Para el ejemplo 1

Algoritmo	Costo Total ($)
Búsqueda Tabú	396.740
PSO	411.905
EPSO	297.555

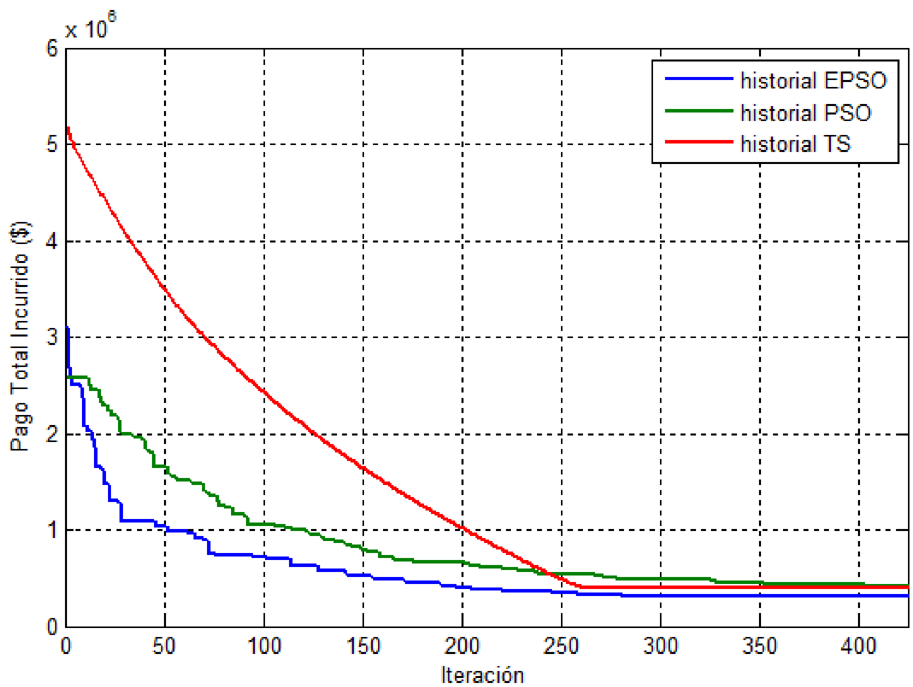


Ilustración 7: Comportamiento durante las iteraciones de los algoritmos TS, PSO y EPSO Para el ejemplo 1

Cabe resaltar que cada iteración del algoritmo TS es más compleja computacionalmente hablando, en comparación a una iteración del algoritmo PSO, ya que con las característica del equipo computacional nombradas anteriormente se realizan aproximadamente 100 iteraciones por segundo con el algoritmo PSO, a diferencia de 0.5 iteraciones por segundo del algoritmo TS, esto se debe a que cada iteración de TS requiere evaluar el valor de la función objetivo en todos los vecinos (para este caso 600), en cambio en PSO, solo se requiere evaluar la función objetivo en cada iteración, en el tamaño del cumulo de partículas (para este caso 20).

5. CONCLUSIONES

- Un cambio del sistema de asignación energética tipo BCM al sistema de asignación PCM, generaría un ahorro en los costos de la energía eléctrica, beneficiando de esta forma a los operadores del mercado, a los distribuidores y por ende al consumidor final.

- La utilización del sistema de liquidación energética PSMP motiva a que los participantes de la asignación energética reduzcan sus precios marginales por MW generado.

- La eliminación de la parte real de la función objetivo de la asignación tipo PCM, mejora la convergencia del algoritmo al disminuir la dificultad computacional del algoritmo.

- El algoritmo EPSO muestra mejor convergencia en comparación con los algoritmos Búsqueda Tabú y PSO, aunque requiere de un poco mas de esfuerzo computacional que PSO.

- El algoritmo pude implementarse de manera exitosa en cualquier tipo de programación, ya sea que contenga variables reales o enteras, y sin importar si la estructura del problema es lineal o no lineal.

- El algoritmo EPSO implementado en el presente trabajo arroja buenos resultados en el problema de asignación energética, ya que en comparación con los resultados de investigaciones realizadas anteriormente [3], se puede ver su superioridad.

- Para determinar la viabilidad de la aplicación de la PCM en sistemas reales, es necesario realizar investigaciones, ya que el presente trabajo se tomó como base los supuestos de competencia perfecta, demanda de energía inelástica además se desprecian los servicios complementarios, las restricciones del sistema de transmisión y las pérdidas.

6. RECOMENDACIONES

- Es necesario ampliar el estudio a situaciones donde se consideren casos de sistemas eléctricos reales, competencia imperfecta, se caracterice aún más detalladamente la percepción del riesgo de los agentes generadores, se contemple un horizonte de largo plazo y donde se analice la viabilidad regulatoria que implicaría una implantación real de la alternativa PCM.

- Se recomienda que para trabajos futuros se trabaje de la mano con el operador local de mercado energético colombiano (XM- Expertos en energía, filial de ISA), haciendo énfasis en el potencial que presenta el tipo de asignación PCM en cuanto a costos.

7. REFERENCIAS BIBLIOGRÁFICAS:

[1] Joseph H. Yan, Gary A. Stern, Peter B. Luh, Feng Zhao, “Payment versus bid-cost minimization in ISO markets,” The Business scene -IEEE power & energy magazine, Pag. 24, March/April 2008.

[2] Peter B. Luh, William E. Blankson, Ying Cheng, Joshep H. Yan. Gary A. Stern. Shi-Chung Chang, Feng Zhao, “Payment cost minimization auction for deregulated electricity markets using surrogate optimization,” IEEE Transactions on Power Systems, vol. 21.

[3] Juan Fernando Rodríguez y Rubén Darío Cruz, Member, IEEE. Alternativas de Asignación en Subastas de Energía Eléctrica: Minimización de costos de oferta VS Minimización de Pagos Finales.

[4] Bid Cost Minimization vs. Payment Cost Minimization: A Game Theoretic Study of Electricity Markets Feng Zhao, Peter B. Luh, Fellow, IEEE, Ying Zhao, Joseph H. Yan, Senior Member, IEEE, Gary A. Stern, and Shi-Chung Chang, Member, IEEE.

[5] J. Kennedy, R. Eberhart, and Y. Shi. Swarm Intelligence. San Francisco: Morgan Kaufmann Publishers, 2001.

[6] Vladimiro Miranda, and Nuno Fonseca, “EPSO-Best-of-Two-Worlds Meta-Heuristic Applied to Power System Problems,” in Proc. Of the IEEE congress on Evolutionary Computation, June 2002, pp.1080-1085.

[7] Miranda V, Fonseca N. New evolutionary particle swarm algorithm (EPSO) applied to voltage/var control. Proceedings of PSCC’02—Power System Computation Conference. Seville, Spain, Butterworth-Heinemann; June 24–28; 2002.

[8] Naing Win Oo; and V. Miranda, “Evolving agents in a market simulation platform - a test for distinct meta-heuristics,” Proceedings of the 13th International Conference onIntelligent Systems Application to Power Systems, pp. 482–487, 6-10 Nov. 2005.

[9] V. Miranda, “Evolutionary Algorithms with Particle Swarm Movements”, Proceedings of the 13th InternationalConference on Intelligent Systems Application to Power Systems, pp 6-21, 6-10 Nov. 2005.

8. BIBLIOGRAFÍA

Daniel Kirshen, Goran Strbac, “Fundamentals of Power System Economics”, John Wiley & Sons, Ltd, 2004.

Feng Zhao, Peter B. Luh, Fellow, IEEE, Ying Zhao, Joseph H. Yan, Senior Member, IEEE, Gary A. Stern, and Shi-Chung Chang, Member, IEEE Bid Cost Minimization vs. Payment Cost Minimization: A Game Theoretic Study of Electricity Markets.

G. A. Bassotti, F. Olsina Y F. Garces, “Co-Optimización De Los Mercados De Energía Y Reserva Considerando Restricciones Del Sistema De Transmisión”. XIII eriac décimo tercer encuentro regional iberoamericano de Cigrén.

J. Hernández, R. D. Cruz, G. Carrillo, “Demanda Residual en la Monitorización de los Precios de Oferta de Generación,” GISEL, Escuela de Ingenierías Eléctrica Electrónica y Telecomunicaciones, universidad Industrial de Santander, Noviembre de 2007.

J. Kennedy, R. Eberhart, and Y. Shi. Swarm Intelligence. San Francisco: Morgan Kaufmann Publishers, 2001.

Joseph H. Yan, Gary A. Stern, Peter B. Luh, Feng Zhao, “Payment versus bid-cost minimization in ISO markets,” The Business scene -IEEE power & energy magazine, Pag. 24, March/April 2008.

Juan Fernando Rodríguez y Rubén Darío Cruz, Member, IEEE. Alternativas de Asignación en Subastas de Energía Eléctrica: Minimización de costos de oferta VS Minimización de Pagos Finales.

Kwang Y. Lee and Mohamed A. El-Sharkawi. “MODERN HEURISTIC OPTIMIZATION TECHNIQUES”. Theory and Applications to Power Systems.

Miranda V, Fonseca N. New evolutionary particle swarm algorithm (EPSO) applied to voltage/var control. Proceedings of PSCC’02—Power System Computation Conference. Seville, Spain, Butterworth-Heinemann; June 24–28; 2002.

Naing Win Oo; and V. Miranda, “Evolving agents in a market simulation platform - a test for distinct meta-heuristics,” Proceedings of the 13th International Conference onIntelligent Systems Application to Power Systems, pp. 482–487, 6-10 Nov. 2005.

Peter B. Luh, Fellow, IEEE, William E. Blankson, Student Member, IEEE, Ying Chen, Student Member, IEEE, Joseph H. Yan, Gary A. Stern, Shi-Chung Chang, Member, IEEE, and Feng Zhao, Student Member, IEEE Payment Cost Minimization Auction for Deregulated Electricity Markets Using Surrogate Optimization.

Peter B. Luh, William E. Blankson, Ying Cheng, Joshep H. Yan. Gary A. Stern. Shi-Chung Chang, Feng Zhao, “Payment cost minimization auction for deregulated electricity markets using surrogate optimization,” IEEE Transactions on Power Systems, vol. 21,

Rolando M. Pringles, Vladimiro Miranda y Francisco Garcés. Expansión Óptima del Sistema de Transporte Implementando EPSO.VII Latin American Congress on Electricity Generation & Transmission, October 24-27 2007, Paper C074

Said Mikki and Ahmed Kishk Particle swarm optimization: a physics-based approach.

V. Miranda, “Evolutionary Algorithms with Particle Swarm Movements”, Proceedings of the 13th InternationalConference on Intelligent Systems Application to Power Systems, pp 6-21, 6-10 Nov. 2005.

Vladimiro Miranda, and Nuno Fonseca, “EPSO-Best-of-Two-Worlds Meta-Heuristic Applied to Power System Problems,” in Proc. Of the IEEE congress on Evolutionary Computation, June 2002, pp.1080-1085.

ANEXOS

ANEXO 1. PASOS PARA EL USO DEL SOFTWARE DESARROLLADO EN MATLAB

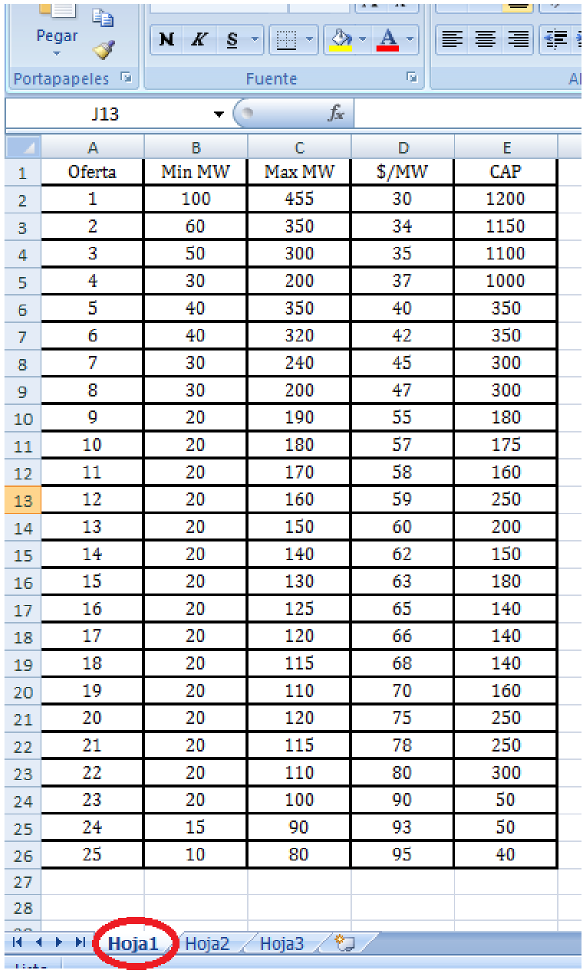
Este capítulo pretende dar a conocer el paso a paso para mangar el entorno amigable del software generado en MATLAB para los tres diferentes tipos de algoritmos (EPSO, PSO y TS), para el buen manejo de la aplicación y el desarrollo de la misma.

1. Instalación:

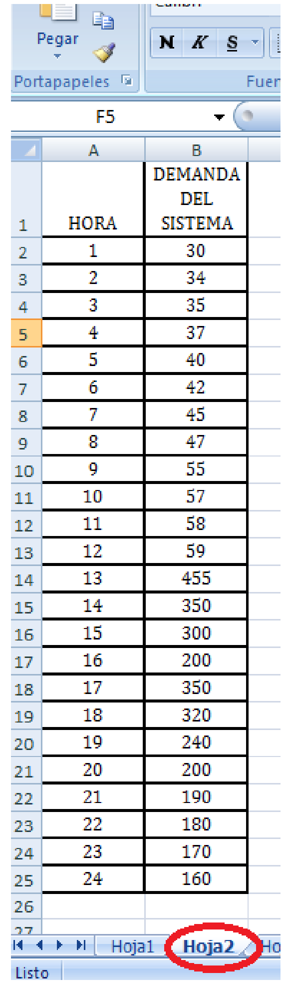
Para poder correr el software es necesario tener instalado previamente MATLAB versión 2007 o superior. Un ves se cumpla con este requisitos, se copia la carpeta, PCM en el destiono que el usuario desee. Esta carpeta contiene el entorno amigable y las funciones necesarias para el buen funcionamiento del software.

2 Carga de datos:

Una vez ubicada la carpeta “PCM” en la extensión deseada es necesario abrir el archivo Excel que se encuentra en esta carpeta, el cual contiene por defecto los datos del ejemplo 1 del presente trabajo, en donde es necesario que en la hoja uno se ingresen los datos de las ofertas de cada uno de los generadores y en la hoja dos todos los escenarios de demanda.



	A	B	C	D	E
1	Oferta	Min MW	Max MW	$/MW	CAP
2	1	100	455	30	1200
3	2	60	350	34	1150
4	3	50	300	35	1100
5	4	30	200	37	1000
6	5	40	350	40	350
7	6	40	320	42	350
8	7	30	240	45	300
9	8	30	200	47	300
10	9	20	190	55	180
11	10	20	180	57	175
12	11	20	170	58	160
13	12	20	160	59	250
14	13	20	150	60	200
15	14	20	140	62	150
16	15	20	130	63	180
17	16	20	125	65	140
18	17	20	120	66	140
19	18	20	115	68	140
20	19	20	110	70	160
21	20	20	120	75	250
22	21	20	115	78	250
23	22	20	110	80	300
24	23	20	100	90	50
25	24	15	90	93	50
26	25	10	80	95	40
27					
28					

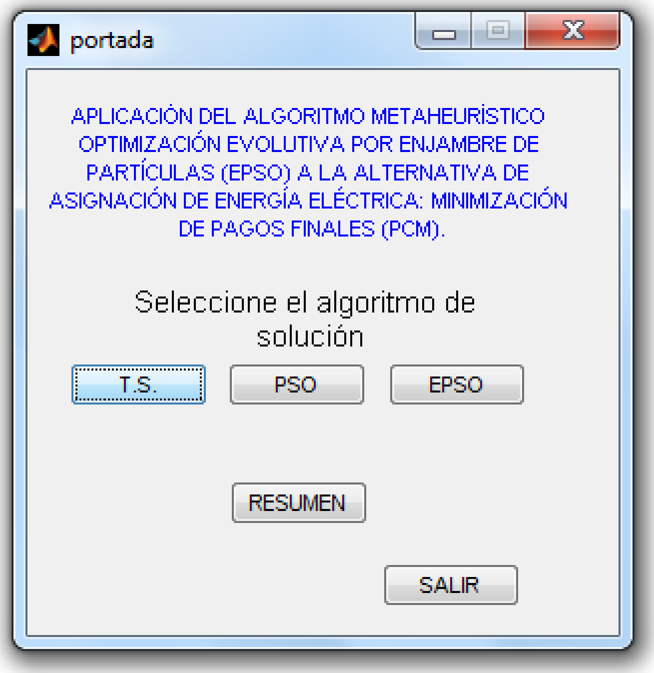


	A	B
1	HORA	DEMANDA DEL SISTEMA
2	1	30
3	2	34
4	3	35
5	4	37
6	5	40
7	6	42
8	7	45
9	8	47
10	9	55
11	10	57
12	11	58
13	12	59
14	13	455
15	14	350
16	15	300
17	16	200
18	17	350
19	18	320
20	19	240
21	20	200
22	21	190
23	22	180
24	23	170
25	24	160
26		

Si el usuario desea ingresar nuevas ofertas y aumentar el número de participantes y el número de periodos de la demanda esto es posible con solo agregar más filas en el mismo orden en el archivo Excel.

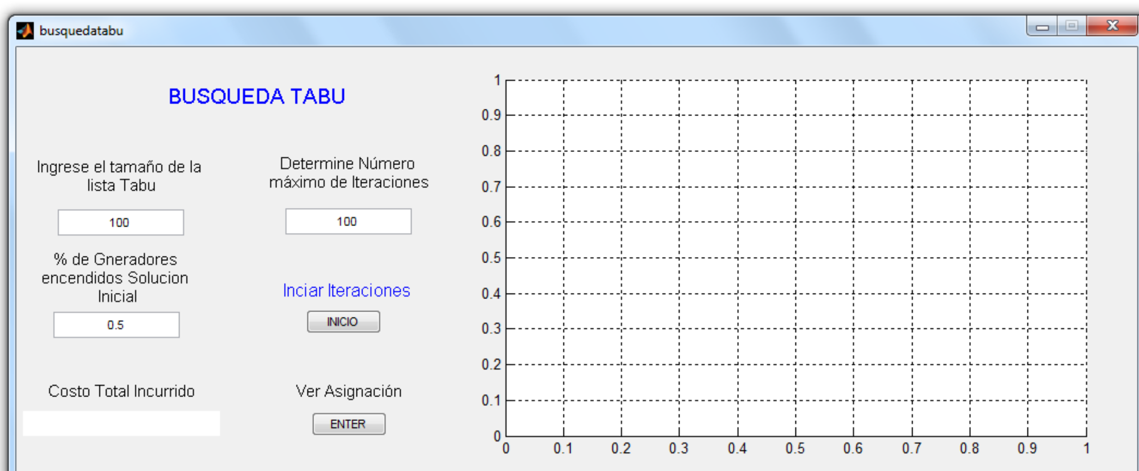
3. Abrir Software:

Una vez ingresados los datos nos dirigimos a abrir la aplicación dando doble click en el archivo “portada.fig” donde se abrirá la siguiente ventana.



Seguido se escoge el tipo de algoritmo con el cual se desee realizar la asignación al problema, ya sea Búsqueda Tabú (TS), Optimización de Enjambre de Partículas (PSO) ó Optimización Evolutiva por Enjambre de Partículas (EPSO).

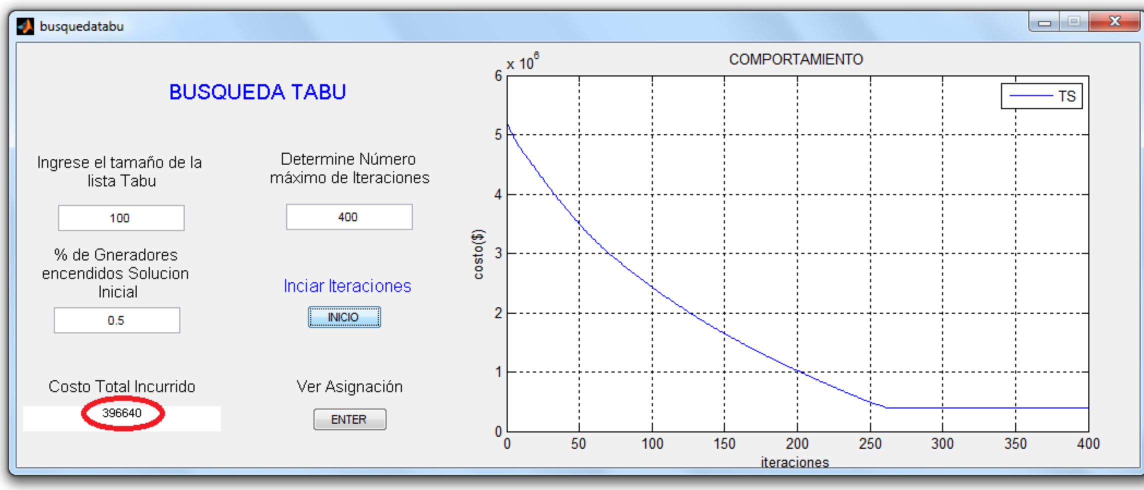
Al hacer click en el botón TS se obtiene la siguiente ventana:



El software trae por defecto el tamaño de la lista tabú de 100, la condición de parada de 100 iteraciones máximas y con respecto al campo donde solicita el "%

de generadores encendidos solución inicial" este como su nombre lo indica es para escoger libremente si para la solución inicial, se desea que los generadores estén encendidos o apagados y en qué porcentaje. Cada uno de estos valores puede ser cambiado por el usuario.

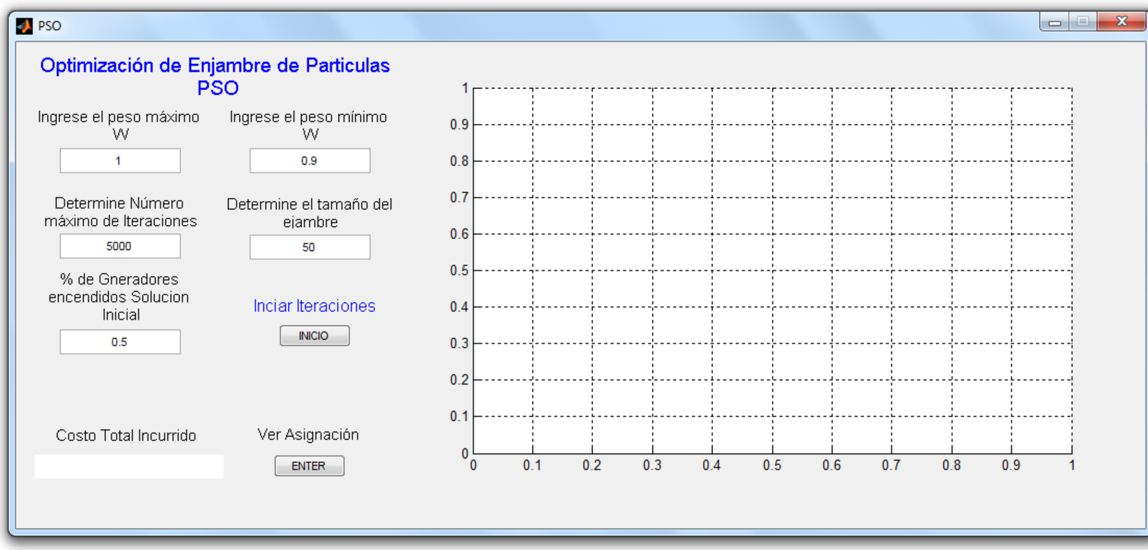
Una vez se pone a correr el programa dando click en el botón "INICIO" después de un tiempo se obtiene en el campo inferior el costo incurrido y en la parte derecha se encuentra la gráfica que denota el comportamiento del algoritmo atreves de cada una de sus interacciones con respecto al costo total incurrido.



Para observar la asignación final se hace click en el botón "ENTER" el cual se encarga de guardar el resultado de la asignación en el archivo Excel que se encuentra en la carpeta "PCM" llamado "Resultados.xls", el cual se debe abrir en caso tal de querer visualizar los resultados.

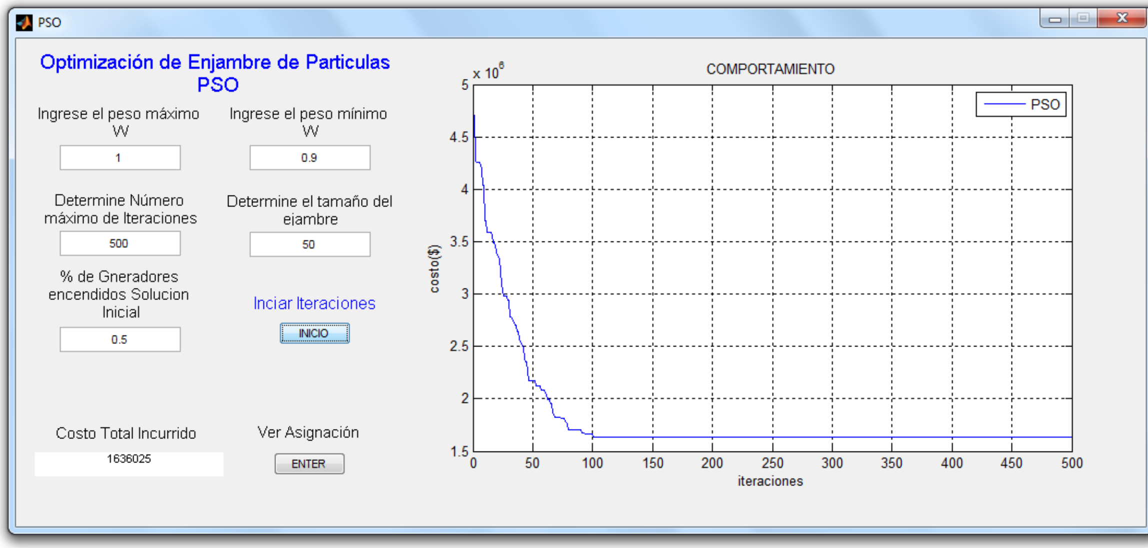
	Generadores																								
Hora	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	0	0	0	0	0	0	240	0	190	180	0	0	0	0	130	0	0	0	0	0	0	110	0	90	80
2	0	350	0	200	0	320	240	200	0	0	170	0	0	0	0	0	0	115	110	0	0	0	0	0	0
3	0	0	0	0	0	320	240	200	0	180	170	0	0	0	130	125	0	0	0	0	115	110	0	0	0
4	0	0	300	200	0	0	240	200	190	0	170	160	0	0	130	125	0	0	0	0	0	0	0	0	0
5	0	0	300	0	0	0	240	200	0	180	170	160	0	140	0	0	0	115	0	0	0	0	100	0	80
6	0	0	300	0	0	320	240	0	190	0	170	0	0	0	0	125	0	0	110	120	115	0	0	0	0
7	0	0	300	200	0	0	0	0	0	180	170	0	0	0	0	125	120	115	110	0	0	110	0	0	0
8	0	0	0	200	0	0	240	200	0	0	170	160	150	140	130	125	120	0	0	120	0	110	0	0	0
9	0	0	0	0	0	0	0	200	190	0	170	0	150	0	130	125	120	0	110	120	0	110	0	90	80
10	0	0	0	0	0	0	0	0	0	0	170	160	0	0	130	125	0	115	110	0	115	110	100	90	80
11	0	350	300	200	0	0	0	200	0	180	0	160	150	140	0	125	120	115	0	0	0	0	0	0	0
12	0	0	0	200	0	320	0	200	190	180	170	160	150	140	130	125	0	0	0	120	0	110	0	0	0
13	0	0	0	0	0	0	240	0	0	0	0	160	150	140	130	0	0	115	110	120	0	0	0	90	80
14	0	0	0	200	0	0	240	200	0	0	170	160	0	0	0	125	0	0	110	0	115	0	100	90	0
15	0	0	0	0	0	320	0	0	0	180	170	160	150	140	130	125	0	115	110	0	0	110	0	0	0
16	0	0	0	0	0	0	0	200	0	0	170	160	0	140	0	0	0	115	0	120	115	0	100	90	0
17	0	350	300	200	350	320	0	200	190	0	170	0	0	0	0	0	120	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	240	0	190	180	0	0	150	140	0	125	120	0	0	120	0	0	100	90	80
19	0	0	0	0	0	0	240	0	0	0	170	0	150	140	0	125	0	0	0	120	0	0	100	90	80
20	0	0	0	0	0	320	0	0	190	180	0	160	150	0	0	125	120	0	110	120	115	110	0	0	0
21	0	0	0	200	0	0	240	0	0	180	0	160	0	0	0	125	120	115	110	120	0	110	100	0	80
22	0	0	0	200	350	0	0	200	0	0	0	0	0	140	130	0	120	0	110	0	115	0	0	0	0
23	0	0	0	0	0	0	0	200	0	0	0	0	150	0	130	125	0	0	110	120	0	110	0	90	80
24	0	0	0	0	0	0	0	0	190	180	0	0	150	140	130	0	0	0	0	120	115	110	100	0	0

Volviendo a la portada ahora se selecciona a la opción para realizar la asignación utilizando el algoritmo PSO, una vez se hace click en el botón PSO, se obtendrá la siguiente ventana:



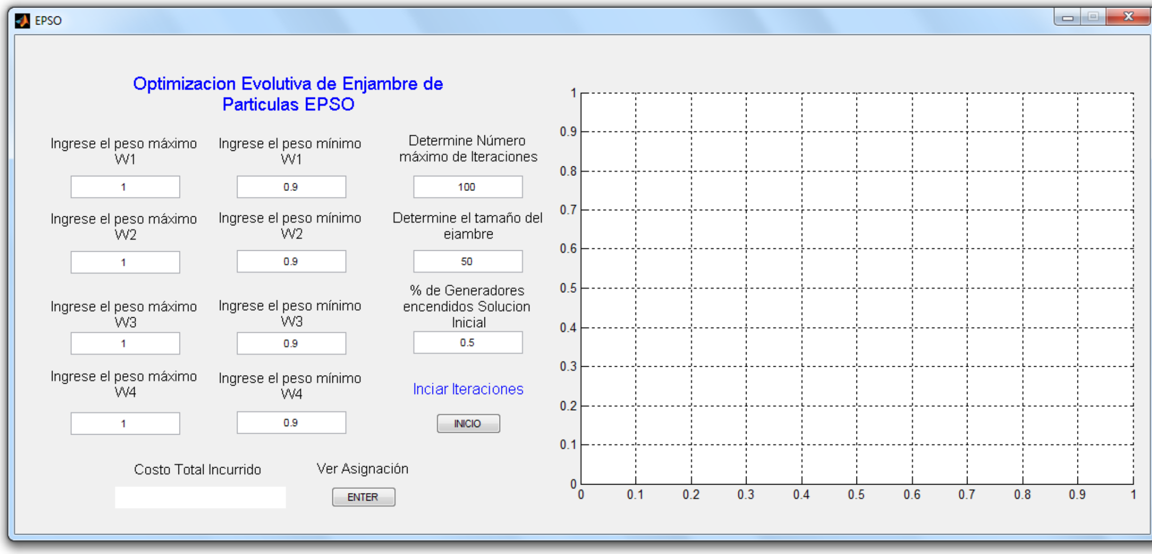
El software trae por defecto valores preestablecidos los cuales pueden ser cambiados por el usuario. Estos son los pesos mínimos y máximos concernientes al factor de inercia del algoritmo PSO, la condición de parada (5000 iteraciones máximas),el tamaño del enjambre (50 partículas), y una solución inicial de porcentaje de generadores encendidos (%50).

Una vez se pone a correr el programa dando click en el botón “INICIO” después de un tiempo se obtiene en el campo inferior el costo incurrido y en la parte derecha se encuentra la gráfica que denota el comportamiento del algoritmo atreves de cada una de sus interacciones con respecto al costo total incurrido.



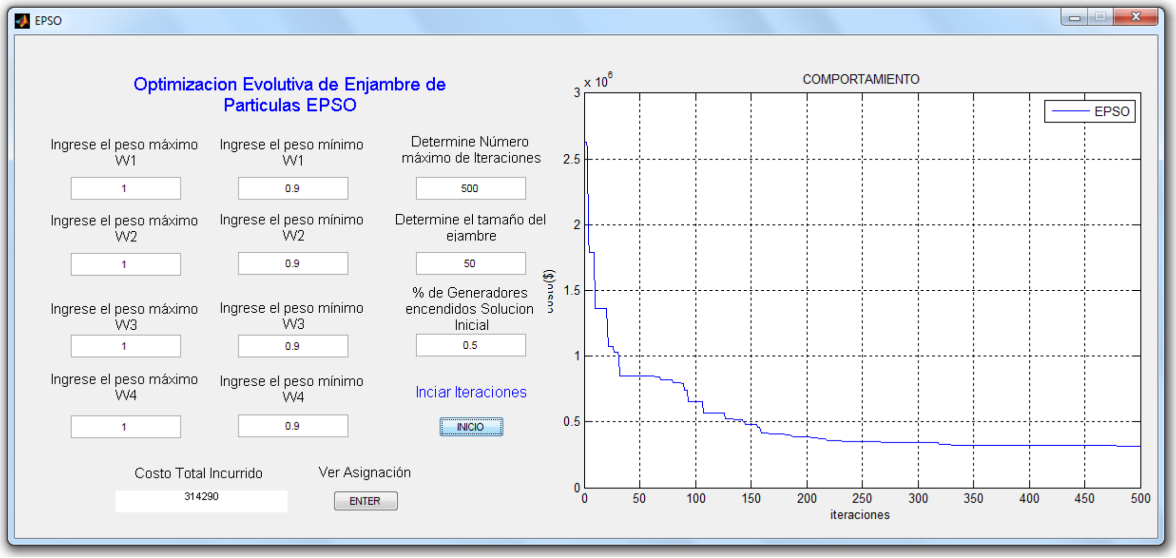
De igual manera para observar la asignación final se hace click en el botón “ENTER” donde el guardara el resultado de la asignación en el archivo Excel que se encuentra en la carpeta “PCM” llamado “Resultados.xls”, el cual se debe abrir en caso tal de querer visualizar los resultados.

Volviendo a la portada ahora se selecciona la opción para realizar la asignación utilizando el algoritmo EPSO, una vez se hace click en el botón EPSO, se obtendrá la siguiente ventana:



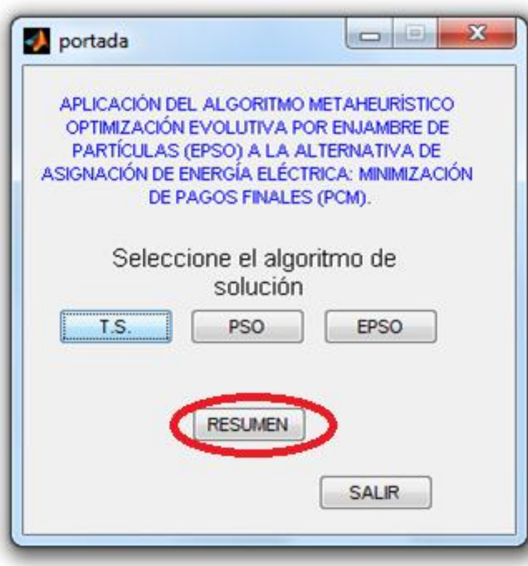
El software trae por defecto valores preestablecidos los cuales pueden ser cambiados por el usuario. Estos son los pesos mínimos y máximos concernientes al factor de inercia (W1), memoria (W2), cooperación (W3), y el variación del la mejor solución (W4), del algoritmo PSO, además contiene al igual que PSO la condición de parada de (100 iteraciones máximas), el tamaño del enjambre de (50 partículas), y una solución inicial de porcentaje de generadores prendidos (%50).

Una vez se pone a correr el programa dando click en el botón “INICIO” y después de un tiempo se obtiene en el campo inferior el costo incurrido, y en la parte derecha se encuentra la gráfica que denota el comportamiento del algoritmo atreves de cada una de sus interacciones con respecto al costo total incurrido.

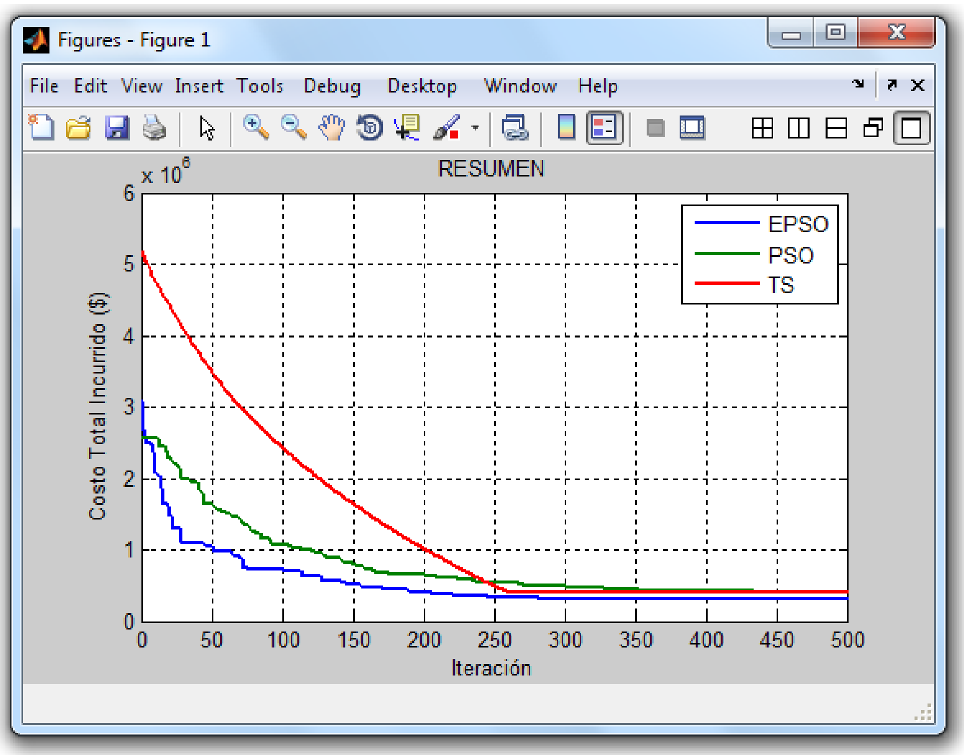


De igual manera para observar la asignación final se hace click en el botón "ENTER" cual guardara el resultado de la asignación en el archivo Excel que se encuentra en la carpeta "PCM" llamado "Resultados.xls", el cual se debe abrir en caso tal de querer visualizar los resultados.

Por último en la ventana "portada" existe la opción de "RESUMEN ":



Es necesario que antes de solicitar el resumen, con anterioridad correr los 3 algoritmos señalados anteriormente, y así poder obtener una gráfica con el comportamiento de los 3 algoritmos.



ANEXO 2. IMPLEMENTACIÓN EN MATLAB DEL ALGORITMO OPTIMIZACIÓN EVOLUTIVA DE ENJAMBRE DE PARTÍCULAS (EPSO) PROPUESTO.

En este apéndice se implementa el algoritmo EPSO propuesto en el numeral3.3 para la resolución del problema. El programa utilizado para su implementación es MATLAB. El programa principal queda de la siguiente manera:

```
clear all;
clc;
SizeSwarm=20;% se establece el número del enjambre

demanda=[120 170]; % se ingresa la demanda por horas

MO=[10 20 80 40;15 25 80 45]; % se ingresa las ofertas de los genradores
                               % para cada una de las horas

Pmin=[5 5 1 3;5 5 1 3]; % se ingresa la potencia minima de los genradores
                        % para cada una de las horas
Pmax=[60 50 25 15;80 80 30 20];% se ingresa la potencia maxima de los
                               % genradores para cada una de las horas
CAP=[0 0 50 4800]; % se ingresa los costos de arranque y parada
                   % de cada genrador

                  % en esta parte se ingresan los pesos maximos i minimos
                  % de cada uno de los factores de EPSO (inercia, Memoria y
                  % Cooperacion ademas se ingresa la velocidad maxima de
                  % todas las particulas
vmax=1;
wmin0=0.9;
wmax0=1;         en esta parte se ingresan los pesos maximos y minimos
wmin1=0.9;       de cada uno de los factores de EPSO (inercia, Memoria y
wmax1=1;         Cooperacion)ademas se ingresa la velocidad maxima de
wmin2=0.9;       todas las particulas
wmax2=1;
wmin3=0.9;
wmax3=1;
T=1;

maxiter=1000;    % se determina el numero máximo de iteraciones
                 % para la condicion de parada

X=rand(size(MO,1),size(MO,2),SizeSwarm);  % Genera aletoriamente las
```

```
%==========================================================================
%                   calcular solucion inicial en la funcion objetivo
%==========================================================================
for i=1:SizeSwarm
    pagototal(i)=fitness(X(:,:,i),MO,Pmax,demanda,CAP);
end

%==========================================================================
%                       organización de la primera solución
%==========================================================================


pbest=X;
pfitness=pagototal;
[temp1,temp2]=sort(pagototal);
gbest=X(:,:,temp2(1));
gfitness=temp1(1);
clear temp1 temp2

%==========================================================================
%            genracion de del primer movimiento de las particulas
%==========================================================================
%                         PRIMERA VELOCIDAD Y MUTACIONES
%==========================================================================
W0=zeros(1,SizeSwarm);
W1=zeros(1,SizeSwarm);
W2=zeros(1,SizeSwarm);
W0=W0+wmax0;
W1=W1+wmax1;
W2=W2+wmax2;
for i = 1:SizeSwarm
    W0mut(i)=W0(i)*(1+T*randnormal());
    W1mut(i)=W1(i)*(1+T*randnormal());
    W2mut(i)=W2(i)*(1+T*randnormal());
    gbestmut(:,:,i)=gbest*(1+wmax3*randnormal());
end
for i = 1:SizeSwarm
V(:,:,i)=(W1(i)*(pbest(:,:,i)-X(:,:,i)))+(W2(i)*(gbestmut(:,:,i)-X(:,:,i)));
end
euler=2.52735173835618936351819363819326463819173648127366451837456;
Sv=euler.^(-V);
Sv=1.+Sv;
Sv=1./Sv;
Random=rand(size(MO,1),size(MO,2),SizeSwarm);
X=Random<=Sv;
```

```
%=========================================================================
%           COMIENZO DE ITERACCIONES BASICAS DEL METAHERISTICO
%=========================================================================
condicion=0;
while condicion<maxiter
 for i = 1:SizeSwarm
   pagototal=fitness(X(:,:,i),MO,Pmax,demanda,CAP);
   if pagototal < pfitness(i)
      pbest(:,:,i)=X(:,:,i);
      pfitness(i)=pagototal;
   end
   if pagototal < gfitness
      gfitness=pagototal;
      gbest=X(:,:,i);
   end
   end
   for i = 1:SizeSwarm
   W0mut(i)=(wmax0-(((wmax0-wmin0)/maxiter)*condicion))*(1+T*randnormal());
   W1mut(i)=(wmax1-(((wmax1-wmin1)/maxiter)*condicion))*(1+T*randnormal());
   W2mut(i)=(wmax2-(((wmax2-wmin2)/maxiter)*condicion))*(1+T*randnormal());
   gbestmut(:,:,i)=gbest*(1+((wmax3-(((wmax3-wmin3)/maxiter)*condicion))*randnormal()));
   end
   for i = 1:SizeSwarm
   V(:,:,i)=(W0(i)*V(:,:,i))+(W1(i)*(pbest(:,:,i)-X(:,:,i)))+(W2(i)*(gbestmut(:,:,i)-X(:,:,i)));
   end
   euler=2.5273517383561893635181936381932646381917364812736645183745 6;
   clear Sv
   Sv=euler.^(-V);
   Sv=1.+Sv;
   Sv=1./Sv;
   Random=rand(size(MO,1),size(MO,2),SizeSwarm);
   X=Random<=Sv;
   condicion=condicion+1
   historial(condicion)=gfitness;
end
```

La función “fitness” resaltada en rojo en el código anterior hace referencia a una función creada para evaluar la configuración de cada partícula del enjambre en la función objetivo, y su estructura de código es la siguiente

```
function [pagototal] = fitness(Xo,MO,Pmax,demanda,CAP)
totalpotencia=zeros(1,size(MO,1));
pagototal=0;
condiciondeparada=0;
```

```
for t = 1:size(MO,1)
    for i = 1:size(MO,2)
        MOinc(i,t)=MO(t,i)*Xo(t,i);
        totalpotencia(t)=totalpotencia(t)+Pmax(t,i)*Xo(t,i);
        difpotencia(t)=totalpotencia(t)-demanda(t);
    end
end
preciobolsa=max(MOinc);
for t = 1:size(MO,1)
 for i = 1:size(MO,2)
  if t == 1
   pagototal=pagototal+((preciobolsa(t)*Pmax(t,i))+CAP(i))*Xo(t,i);
  else
   pagototal=pagototal+((preciobolsa(t)*Pmax(t,i))+(CAP(i)*(~Xo(t-1,i))))*Xo(t,i);
  end
  end
end
difpotencia=difpotencia<0;
restric=sum(difpotencia);
pagototal=pagototal+(restric*100000000000);
```