

Optimización binaria por cúmulo de partículas con memoria (MBPSO) para resolver un problema de espectro compartido

Esteban Martínez¹, Ángel G. Andrade¹, Anabel Martínez-Vargas², Guillermo Galaviz¹

¹ UABC, Facultad de Ingeniería, Grupo de Tecnologías para Ambientes Inteligentes, Mexicali, Baja California, México

² Universidad Politécnica de Pachuca, Maestría en Tecnologías de la Información y Comunicaciones, Zempoala, Hidalgo, México

{esteban.martinez98, aandrade, ggalaviz}@uabc.edu.mx, anabel.martinez@upp.edu.mx

Resumen. Compartir espectro es una de las soluciones que permitirá que Redes Heterogéneas (HetNet), formada por diversas tecnologías de acceso inalámbricos, dispongan de recurso espectral adicional. Con esto se promueve la coexistencia entre los diferentes sistemas de radio, incrementando con ello su eficiencia espectral, pero también la interferencia en el sistema. En este trabajo se implementa una estrategia que controla la interferencia y asignación de canal en una HetNet, con el fin de maximizar la tasa de datos y la cantidad de usuarios que comparten espectro concurrentemente. Para resolver este problema se utilizó la técnica de Optimización Binaria por Cúmulo de Partículas con Memoria (MBPSO). A diferencia de la técnica optimización por Cúmulo de Partículas Socio-Cognitiva (SCPSO), MBPSO evita la convergencia prematura en óptimos locales y la perdida de diversidad. Los resultados muestran que al utilizar MBPSO se mejoran las soluciones sobre la capacidad del sistema que cuando se utiliza SCPSO. Además se compara el desempeño de MBPSO con las técnicas PSO Modificado (ModBPSO) y Optimización por Cúmulo de Partículas con Modulación Angular (AMPSCO), ya que estas últimas también cuentan con la habilidad de explorar y explotar diversos espacios de soluciones.

Palabras clave. Compartición de espectro, eficiencia espectral, redes heterogéneas, MBPSO.

Memory Binary Particle Swarm Optimization (MBPSO) Applied to a Spectrum Sharing Problem

Abstract. Spectrum sharing is one of the solutions for Heterogeneous Networks (HetNets) for achieving additional spectral resource. The aim is to promote the

coexistence of different radio systems in the same spectral portion increasing the spectral efficiency of the HetNet, but at the same time the interference is increased. In this paper, we tackle with the problem of spectrum sharing in a HetNet composed of a macrocell and several femtocells. We propose a strategy, in which macrocell and femtocells can share simultaneously the available bandwidth while avoiding intra-tier interference. Our approach is formulated as a binary optimization problem. The fitness is evaluated considering techniques of binary optimization with memory to overcome the problem of premature convergence or loss of diversity that Socio-Cognitive Particle Swarm Optimization (SCPSO) presents. The results show that by using the Memory Binary Particle Swarm Optimization (MBPSO) algorithm, the system's capacity is improved in comparison with solutions obtained using SCPSO. Also the performance of MBPSO is compared with Angle Modulated PSO (AMPSCO) and Modified BPSO (ModBPSO) algorithms.

Keywords. Spectrum sharing, spectral efficiency, heterogeneous networks, MBPSO.

1. Introducción

Hoy en día existe una diversidad de dispositivos que ofrecen servicios de comunicación inalámbrica, ya sea voz, video o datos; que permiten comunicarnos en cualquier momento y lugar. Actualmente existen a nivel mundial más de un trillón de dispositivos móviles que cuentan con algún tipo de interfaz inalámbrica, por lo que el consumo de datos de banda ancha está aumentando vertiginosamente [1]. Para atender

las demandas de ancho de banda y Calidad de servicio (QoS), las nuevas generaciones de redes móviles e inalámbricas requerirán del trabajo conjunto de múltiples redes y tecnologías de acceso de radio (RATs: Radio Access Technologies), conocidas como Redes Heterogéneas (HetNets: Heterogeneous Networks). Una HetNet se compone principalmente por una combinación de celdas de gran tamaño (macro-celdas) y celdas pequeñas (micro-celdas, pico-celdas y femto-celdas). Las redes y aplicaciones inalámbricas están reguladas mediante una política de asignación de espectro fija. El Estado regula, administra y asigna la utilización de las diferentes bandas de frecuencia a distintas empresas, usuarios y/o servicios. Con esta administración fija del espectro se le garantiza a cada tecnología inalámbrica protección contra la interferencia pero por otro lado, provoca que sea más difícil encontrar bandas libres para el despliegue de HetNets, generando la idea de una escasez de espectro.

Por lo anterior, se ha considerado la necesidad de introducir reformas, no sólo para mejorar su utilización, sino también para proveer de nuevo espectro a las nuevas aplicaciones de banda ancha móvil. Una solución que permitirá que los sistemas de comunicación inalámbrica que no dispongan de recurso espectral propio, puedan acceder al recurso espectral disponible de otras redes, es la de compartir bandas de frecuencia. El esquema de espectro compartido promueve la coexistencia entre diferentes sistemas de radio en la misma porción espectral sin importar la banda de frecuencia de operación que el Estado les asignó originalmente [2]. Entre mayor sea la cantidad de usuarios que utilicen un canal concurrentemente, mayor será la eficiencia en el uso del espectro, aunque también mayor será la interferencia a la que estarán expuestos los usuarios de la red que participan en la compartición de espectro [3].

Se considera que la interferencia es dañina si es lo suficientemente elevada como para causar una interrupción en el servicio a cualquiera de los usuarios de la red. Para controlarla, se propone controlar el acceso de usuarios al sistema mediante una asignación efectiva de canales de transmisión. Si se cumple la restricción de admisión impuestas por el algoritmo, entonces se

permite que el usuario utilice los recursos de la red, en caso contrario, el usuario será bloqueado.

El problema de espectro compartido se considera como un problema combinatorio de gran complejidad (NP-difícil), debido al gran consumo en recursos computacionales necesarios para encontrar una solución global óptima, los cuales crecen de forma exponencial con respecto al tamaño del problema [3-5]. En este caso, la complejidad del problema se incrementa conforme se incrementa el número de usuarios que le solicitan a otra red un canal de comunicación compartido. En la literatura se reportan algunas propuestas que resuelven el problema de espectro compartido considerando la interferencia como su principal restricción [4-10].

En [4], los autores maximizan la cantidad de datos transmitidos (throughput) en una red heterogénea de espectro compartido utilizando una técnica basada en la programación geométrica secuencial.

Utilizando el Método Montecarlo, en [5] se generan más un millón de escenarios distintos a partir de diferentes combinaciones para determinar la ubicación espacial de los usuarios secundarios y la tasa de datos trasmisitos durante la detección de canales disponibles en bandas reguladas de televisión.

En [6], los autores proponen una estrategia para que los usuarios exploten bandas de frecuencia en una de dos modalidades; (i) concurrentemente con usuarios de diferentes sistemas o, (ii) esperar a que una banda se libere para explotarla. La decisión para conmutar entre una modalidad y otra, la toman con base a no sobrepasar un nivel de interferencia permitido en la red.

En [7], los autores aplican Teoría de Juegos para maximizar la tasa de datos de una HetNet. Definen un compromiso que consiste en que la macrocelda venda una porción de su ancho de banda a solicitud de la femtocelda, permitiéndole transmitir simultáneamente en su misma banda de frecuencia. El costo del ancho de banda es directamente proporcional a la cantidad de espectro solicitado por la femtocelda, pero se reduce proporcionalmente con la cantidad de usuarios macrocelulares que la femtocelda pueda atender con esa misma cantidad de espectro. Entre mayor sea la cantidad de espectro solicitado,

mayor será la tasa de datos que logre transmitir la femtocelda, al igual que el costo y la interferencia generada hacia usuarios macrocelulares. Aunque estos trabajos resuelven para un gran número de usuarios secundarios que pueden compartir un canal, requieren de una excesiva complejidad computacional para encontrar el óptimo global.

En contraste con los trabajos anteriores, en [8-10], los autores utilizan técnicas de inteligencia computacional inspirados en procesos naturales para encontrar soluciones "buenas" al problema de espectro compartido, en un tiempo y consumo de recursos aceptables. Tal es el caso de Algoritmos Genéticos (GA) [8], Optimización por Colonia de Hormigas (ACO) [9] y Optimización por Cumulo de Partículas (PSO) [10]. La ventaja de aplicar estos algoritmos es que son sencillos de implementar y requieren poca información sobre el espacio de búsqueda de posibles soluciones.

En un trabajo previo [12], utilizamos la técnica de optimización por Cúmulo de Partículas Socio-Cognitiva (SCPSO) [19] para maximizar la cantidad de datos transmitidos (throughput) en una HetNet y la cantidad de usuarios que concurrentemente utilizan el mismo canal. Sin embargo, debido a un posible estancamiento o convergencia prematura en óptimos locales que presenta la técnica SCPSO como se reporta en [13-18], las soluciones obtenidas en [12] no se consideran como suficientemente "buenas". Para resolver lo anterior, en la literatura se reportan algunas propuestas que permiten que el enjambre, en la técnica PSO, se mueva hacia distintas regiones, para lograr mayor diversidad del cúmulo y evitar que se concentren en una región reducida del espacio de soluciones. Ejemplos de ellas son; Optimización Binaria por Cúmulo de Partículas Modificado (ModBPSO) [14], Optimización por Cúmulo de Partículas con Modulación Angular (AMPSO) [15] y Optimización Binaria por Cúmulo de Partículas con Memoria (MBPSO) [16].

En este trabajo se propone resolver el problema de espectro compartido en una HetNet aplicando la técnica de optimización MBPSO. Una HetNet está formada por una macro-celda (conocida como la red primaria) y por un conjunto de celdas más pequeñas o femto-celdas (denominada la red secundaria) que operan en la misma banda de frecuencia que la red primaria tiene asignada. El problema de optimización consiste en maximizar

la cantidad de datos transmitidos (throughput) en la red y, esta depende de la cantidad de usuarios femtocelulares o secundarios (SUs) que utilicen el mismo canal primario que los usuarios de la macrocelda o primarios (PUs). La asignación de un canal primario a un SU depende del nivel de interferencia que éstos generan a los PUs y, a aquellos SUs que utilizan el mismo canal primario. Los resultados obtenidos con la técnica MBPSO se comparan con los obtenidos con SCPSO, considerada como una adaptación binaria del algoritmo PSO para espacios continuos. Además de presentar una análisis comparativo con las técnicas AMPSO y ModBPSO.

El resto del artículo está organizado de la siguiente manera, en la sección 2 se describe la metaheurística MBPSO, la estrategia de control de admisión y asignación de canal en una HetNet y la integración de la técnica MBPSO se describe en la sección 3. En la sección 4 se describe el escenario de evaluación y se analizan los resultados, las conclusiones de este trabajo se presentan en la sección 5.

2. Optimización binaria por cúmulo de partículas con memoria (MBPSO)

2.1. PSO

La Optimización por Cúmulo de Partículas (PSO) es una técnica heurística desarrollada por Kennedy y Eberhart [11], basada en la simulación del comportamiento social observado en las aves dentro de una parvada. En PSO, cada individuo representa una partícula que viaja a través de un espacio de soluciones del problema. Cada partícula tiene asignada una posición x_{id} , que representa una posible solución al problema a resolver, y un valor de velocidad v_{id} , que regula el movimiento en la posición de la partícula. La calidad de la posición de cada partícula (aptitud) se relaciona con una o varias funciones objetivo que representan el problema a resolver. Para promover la exploración amplia del espacio de búsqueda, las posiciones y velocidades iniciales asignadas a cada partícula se generan de forma aleatoria. Conforme avanza el algoritmo, la velocidad y la posición cambian en función de la interacción social basada en la tendencia social de

Tabla 1.- Valores del estado de bit de velocidad de la partícula

Estado actual	Estado objetivo	Resultado
1	0	0
2	0	1
3	1	0
4	1	1
		Sin cambio

cada individuo a emular el éxito de otros individuos en la población. El cambio en la posición de cada partícula depende de su propio conocimiento y medio ambiente, ya que en dicho cambio se considera la mejor posición visitada por la partícula y la mejor posición visitada por algún individuo del enjambre (*pbest*). A medida que el algoritmo avanza, las partículas se concentran en zonas con soluciones de buena calidad del espacio de búsqueda (*gbest*). Al finalizar, el algoritmo devuelve la mejor solución visitada por algún individuo del enjambre.

A pesar de que PSO y algunas de sus variantes, como SCPSO, se consideran técnicas robustas para solucionar complejos y diversos problemas de optimización, son deficientes al tratar de satisfacer un balance entre la *exploración* y *explotación*. La *exploración* se refiere a la capacidad del algoritmo de moverse dentro de un espacio de búsqueda amplio, de tal forma que no se limite a buscar en una sola región, sino que explore otras. En cambio, la *explotación* es la habilidad que ofrece la técnica para evaluar localmente una región y guiar a las particulas a la búsqueda del mínimo local. Incrementar solo la exploración de la técnica resulta en una búsqueda aleatoria en el dominio del problema, y mejorar únicamente la explotación resultará en una convergencia prematura a posibles soluciones.

2.2. MBPSO

PSO convencional se diseñó para resolver problemas de espacio continuo, mientras que SCPSO se aplica en la resolución de problemas de optimización que requieren una representación binaria de sus soluciones. En esta versión se mantiene la diversidad en el cúmulo introduciendo

la distancia entre *gbest* y *pbest* como una nueva ecuación de actualización de velocidad. La convergencia del algoritmo es influenciada por el número de partículas en el cúmulo y la dimensión del problema, por lo que para problemas de alta complejidad la técnica no converge a una solución confiable y en algunos casos requiere de más iteraciones para converger.

La técnica MBPSO [16] propone algunas modificaciones al algoritmo SCPSO, las cuales evitan que la información se propague con rapidez entre las partículas y se agrupen alrededor del primer óptimo local que encuentran. En MBPSO, el valor de la posición de la partícula x_{id} , se actualiza con base a su posición previa. En SCPSO, v_{id} representa la probabilidad de que un bit cambie a 1, mientras que en MBPSO, representa la probabilidad de que un bit permanezca sin cambio dependiendo de uno de los cuatro estados esperados (ver tabla 1), la cual conoce a partir de *pbest* y *gbest*. Cuando el bit del *estado actual* es el mismo que el del *estado objetivo*, el bit memoriza el valor del bit del estado actual y se mantiene con el mismo valor, de lo contrario se incrementará la probabilidad de que el bit cambie de estado. Los valores de 0 y 1 que puede tomar el bit se obtienen a partir de la siguiente expresión:

$$\text{sig}(v_{id}^{k+1}) = \frac{1}{1 + e^{-v_{id}^{k+1}}}. \quad (1)$$

En este sentido, la representación de velocidad y posición de la partícula se determinan a partir de las siguientes expresiones:

$$v_{id}^{k+1} = w v_{id}^k + c_1 \times r_1 \times \text{Equal}(p_{id}, x_{id}^k) \\ + c_2 \times r_2 \times \text{Equal}(p_{gd}, x_{id}^k), \quad (2)$$

$$x_{id}^{k+1} = \begin{cases} x_{id}^k & \text{if } (\text{rand}() < \text{sig}(v_{id}^{k+1})) \\ \bar{x}_{id}^k & \text{otro} \end{cases}, \quad (3)$$

donde

$$\text{Equal}(a, b) = \begin{cases} 1 & \text{si } (a == b) \\ -1 & \text{otro} \end{cases}, \quad (4)$$

a, b toman el valor correspondiente a p_{id} , X_{id} , o p_{gd} , X_{id} según corresponda, w, c_1, c_2 son el peso de inercia y las constantes cognitiva y social respectivamente; r_1, r_2 son números aleatorios uniformemente distribuidos entre $[0,1]$, p_{id} es la mejor solución encontrada (p_{best}) por la partícula i y p_{gd} es la mejor solución encontrada (g_{best}) por la partícula g . La función $rand()$ genera un número entero aleatorio con distribución uniforme entre 0 y N .

Finalmente, en SCPSO se consideran valores grandes de peso de inercia (ω) para mejorar la búsqueda global de soluciones o, valores pequeños para mejorar la búsqueda local. En MBPSO se propone una estrategia lineal decreciente para determinar el peso de inercia, la cual está definida por la siguiente expresión:

$$\omega = \omega_{\max} - \frac{(\omega_{\max} - \omega_{\min})}{Iter_{\max}} \times Iter_{current}, \quad (5)$$

donde ω_{\max} y ω_{\min} son el límite superior e inferior de ω respectivamente, $Iter_{\max}$ es el número máximo de iteraciones que se le permiten a la mejor solución (valor de aptitud) permanecer sin cambio, mientras que $Iter_{current}$ representa el número de iteraciones consecutivas durante las que la mejor solución no mejora.

3. Procedimiento para la asignación de canal en la HetNet

En una red heterogénea distintas tecnologías inalámbricas coexisten con el fin de incrementar la capacidad del sistema en aquellas zonas de mayor demanda de recursos, distribuyendo el tráfico entre las celdas pequeñas para que no se generen cuellos de botella en la macro-celda (ver Fig.1).

Al operar en un esquema de espectro compartido, la HetNet requiere de una estrategia para controlar el acceso a los *SUs* y asignarles un canal de comunicación de alguna de las bandas de frecuencia disponibles en el sistema.

Para garantizar la transmisión exitosa entre los usuarios que comparten simultáneamente un canal, es necesario mantener el nivel de interferencia del sistema por debajo de un valor de umbral predeterminado.

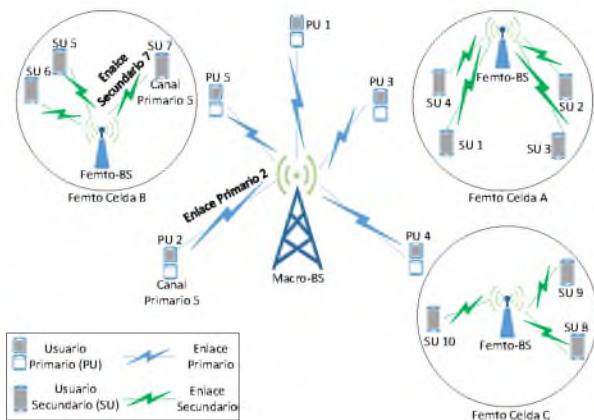


Fig. 1. Red Heterogénea, formada por varias femtoceldas desplegadas sobre una macrocelda

La Relación Señal-Interferencia (*SIR*) es la métrica que evalúa la relación entre la potencia de transmisión del usuario y la potencia que éste percibe de otros transmisores que transmiten en el mismo canal (interferencia). La selección de aquellos *SUs* que se les puede asignar un canal primario depende de la interferencia que éstos generen a los *PUs* y a aquellos *SUs* que utilicen el mismo canal primario. Para determinar el nivel de interferencia que experimenta cualquiera de los enlaces en el sistema, las expresiones (6) y (7), permiten determinar el valor de *SIR* que observa el receptor secundario y primario respectivamente:

$$SIR_v = \frac{P_v/ldp(v)^n}{\sum_{k \in \Phi} P_k/dps(k, v)^n}, \quad (6)$$

$$SIR_u = \frac{P_u/lds(u)^n}{\sum_{k \in \Phi} P_k/dss(k, u)^n + P_v/dps(v, u)^n}. \quad (7)$$

Para la expresión (6), P_v es la potencia del v -ésimo transmisor primario, $ldp(v)$ es la distancia entre el v -ésimo transmisor-receptor primario y n es el exponente de pérdidas por propagación que puede tomar valores entre 2 y 4. P_k es la potencia de transmisión del k -ésimo transmisor secundario, $dps(k, v)$ es la distancia entre el v -ésimo receptor primario y el k -ésimo transmisor secundario y Φ es el conjunto de *SUs* activos que comparten un canal primario con el v -ésimo *PU*, esto representa

la interferencia que perciben los *PUs* debido a las transmisiones de los *SUs*. Por otra parte, para la ecuación (7), P_u es la potencia del u -ésimo transmisor secundario, $lds(u)$ es la distancia entre el u -ésimo transmisor-receptor secundario. Mientras que, $dss(k,u)$ es la distancia entre el u -ésimo receptor secundario y el k -ésimo transmisor secundario, $dps(v,u)$ es la distancia entre el u -ésimo receptor secundario y v -ésimo el transmisor primario.

La capacidad de la HetNet se define por la máxima tasa de datos que la red transporta y, ésta depende de la cantidad de *SUs* con permiso para transmitir en un canal primario. La función objetivo (8) permite contabilizar la suma total de datos que se genera en la HetNet, mientras que las expresiones (9) y (10) calculan respectivamente la tasa de datos de aquellos *SUs* y *PUs* que cumplen con la restricción de no generar interferencia dañina entre ellos, garantizando con ello una QoS en ambas redes. En este caso x_u es una variable binaria que indica si a un *SU* se le asignó un canal para considerarlo en el cálculo de la tasa de datos.

La función objetivo está sujeta a que se garantice una QoS tanto para los *SUs* (11) como para los *PUs* (12). En este sentido, α y β representan el nivel de interferencia máximo que pueden tolerar las redes secundaria y primaria respectivamente, con el fin de mantener un nivel de QoS a los usuarios de la HetNet.

Con base a la Recomendación UIT-T G.174 de la Unión Internacional de Telecomunicaciones, en la que se establecen los objetivos de calidad de transmisión para los sistemas digitales terrestres móviles que utilizan terminales portátiles para acceder a la red telefónica, mantener un valor de α y β igual o superior a los 4 dB significa que los usuarios de la HetNet que utilicen un servicio que requiera condiciones de tráfico bajo (por ejemplo; servicio de mensajería) no experimentarán una interrupción de su servicio durante el tiempo que lo utilicen.

Por otro lado, para mantener sin interrupción a aquellos servicios o aplicaciones que demandan tráfico alto (por ejemplo, transmisión/recepción de video en tiempo real) la HetNet debe mantener un nivel de SIR igual o superior a los 14 dB (esto es, $\alpha=\beta>14$ dB).

Comunicación del *PU* que está utilizando el mismo canal primario. Para evitar que las pérdidas

por propagación degraden la comunicación entre *SU-femtocelda* y *PU-macroBS* y, que el algoritmo de optimización no converja a alguna solución, se limitaron las distancias a 1000 m y 30 m respectivamente.

$$\text{Max} \sum_{u=1}^{SU} c'_u x_u + \sum_{v=1}^{PU} c''_v, \quad (8)$$

$$c''_v = B \log_2(1 + SIR_v), \quad (9)$$

$$c'_u = B \log_2(1 + SIR_u), \quad (10)$$

$$SIR_u \geq \alpha, \quad (11)$$

$$SIR_v \geq \beta. \quad (12)$$

El algoritmo 1 muestra la implementación de MBPSO para resolver el problema de espectro compartido. Cada partícula que se utiliza en el algoritmo representa una posible solución al problema, esto es, el conjunto de *SUs* que maximizan la capacidad de la HetNet. Cada partícula del cúmulo cuenta con tres vectores *D-dimensionales* asociados (donde $d=1,2,\dots,D$ representa el número de *SUs* desplegados en la HetNet), de velocidad V , de posición X y el que almacena las mejores soluciones P . Cada componente de la partícula x_{id} , puede tomar un valor de entre {0,1}, mientras que la velocidad, v_{id} , puede tomar un valor de entre $[-V_{max,max}]$, donde V_{max} es la velocidad máxima. Durante la *Fase de inicialización*, se genera aleatoriamente el espacio de búsqueda inicial de posibles soluciones.

Se proponen tres nuevos vectores; X' , que contiene el canal asignado de cada *SU*, P' , que contiene la mejor asignación de canal encontrada hasta el momento para cada partícula y el vector **Estado del espectro**, que indica el canal primario (índice) asignado a un *PU* (contenido). Este vector se inicializa aleatoriamente y su tamaño depende de la cantidad de *PUs* que se consideren en el problema.

La aptitud de la partícula se refiere a la cantidad de datos total que el sistema tiene capacidad de transmitir en un instante de tiempo, este valor depende principalmente del ancho de banda del canal, B , y, de la interferencia presente en la HetNet generada por los *PUs* y *SUs*. La

interferencia en el sistema se mide en términos del S/I , si el nivel de S/I de alguno de los SUs o PUs que comparten algún canal está por debajo de α y β respectivamente, entonces se considera una solución *no válida* y la aptitud se penaliza con un valor 0. En caso contrario, se calcula la tasa de datos para cada red utilizando (9 y 10).

El vector P_i guarda la mejor solución encontrada por la partícula hasta el momento, así como en P'_i se almacena la mejor asignación de canal. Por último, se generan nuevas soluciones candidatas con base a las ecuaciones (1 y 5). En esta fase, se genera un nuevo valor en la posición x_{ij} de dimensión d del vector X_i y un nuevo valor de velocidad v_{ij} correspondiente a la misma dimensión d del vector V_i , para la partícula j .

La mejor solución encontrada por el algoritmo consiste en; (i) la cantidad de SUs seleccionados que pueden utilizar un canal primario concurrentemente con los PUs , (ii) su ubicación espacial, (iii) el canal primario que están compartiendo PUs y SUs y, (iv) la tasa de datos máxima que ofrece el sistema $f(P_{gt})$.

Parámetros de entrada: SU considerados en la HetNet, PU considerados en la HetNet, umbral de calidad requerido para la red primaria y secundaria (α , β respectivamente), tamaño del cúmulo (S partículas), número de iteraciones (T_{max}), V_{max} , límite superior e inferior de ω (ω_{max} , ω_{min}), vector de posición (X), vector de mejores soluciones (P), vector de velocidad (V), vector que contiene el canal asignado de cada SU (X'), vector que contiene la mejor asignación de canal encontrada hasta el momento para cada partícula (P).

Algoritmo 1. Pseudocódigo para resolver el problema de espectro compartido aplicando la técnica MBPSO.

```

1: Desplegar aleatoriamente PUs y SUs
2: Inicializar aleatoriamente X, X', V y vector
   Estado del espectro.
   Hacer P=X, P'=X'
   Repite
3: Para i = 1 a S
4:   Hacer mapeo de X'_i y Estado del espectro
      para calcular niveles de SIR en SUs y PUs
5:   Hacer mapeo de P'_i y Estado del espectro
      para calcular niveles de SIR en SUs y PUs.
6:   Calcular aptitud f(X'_i)
7:   Calcular aptitud f(P'_i)
8:   Si f(X'_i) > f(P'_i) entonces
9:     Para j = 1 a número de dimensiones
      p_{ij} = x_{ij}
      p'_{ij} = x'_ij

```

```

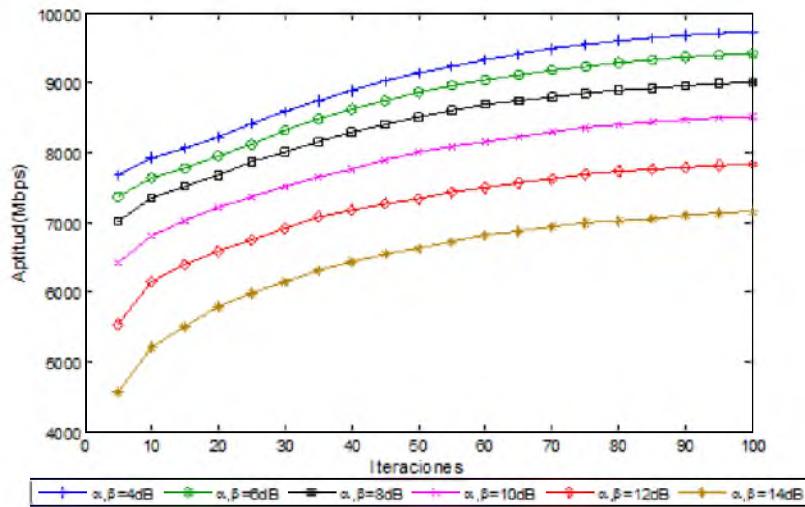
Siguiente j
Fin Si
gt = i-1
Para j = 1 a S
12:   Hacer mapeo de P'_j y Estado del espectro
      para calcular niveles de SIR en SUs y PUs.
13:   Hacer mapeo de P'_{gt} y Estado del espectro
      para calcular niveles de SIR en SUs y PUs.
14:   Calcular aptitud f(P'_{gt})
15:   Si f(P'_j) > f(P'_{gt}) entonces gt = j
16: Siguiente j
17: Para j = 1 a número de dimensiones
18:   v_{ij} = wv_{ij} + c_1r_1 x Equal(p_{ij},x_{ij} ) + c_2r_2 x
      Equal(p_{ij} - x_{ij} )
19:   Calcular sig(v_{ij})
20:   Si rand()<sig(v_{ij}) entonces x_{ij} = x_j
21:   Si x_{ij}=1 entonces asignar aleatoriamente un
      nuevo canal primario a x'_{ij}
22: Siguiente j
23: Siguiente i
24: Hasta que se cumpla Tmax
25: Regresar parámetros de salida: Tasa Máxima de
      datos lograda en la Hetnet (f(Pgt)), conjunto de SUs
      seleccionados (Pgt), vector de Estado de espectro
      en el que se informa con que canal primario quedó
      asignado cada SU seleccionado.

```

4.2. Evaluación del problema de espectro compartido utilizando MBPSO

En esta subsección se analiza el desempeño de MBPSO para distintos parámetros de $iter_{max}$ y V_{max} , para esto se utiliza como caso de estudio el problema de espectro compartido descrito en la sección 4.1. Para evaluar la capacidad de la HetNet se realizaron 500 experimentos, por cada experimento se realizan 30 corridas independientes y cada una de ellas consiste de un nuevo despliegue (aleatorio) de SUs y PUs [12]. En cada corrida se evalúan un conjunto de posibles combinaciones de asignación de un canal primario a un SU , la cantidad de combinaciones depende de la cantidad de partículas e iteraciones. Los valores de α y β establecen la QoS (medido en términos de S/I) en las redes primaria y secundaria respectivamente. Con base a [16] y [19], en la Tabla 2 se muestran los parámetros utilizados durante los experimentos.

El algoritmo de evaluación y las técnicas de optimización se implementaron en Lenguaje C, ejecutándose en una estación de trabajo Dell Precision Tower 7910, la cual incluye un

**Fig 2.** Convergencia de soluciones al problema de espectro compartido, PU=6, SU=20, Itermax=20, Vmax=6**Tabla 2.-** Parámetros utilizados durante la evaluación

Para MBPSO [16]	Valor
Tamaño del cúmulo (S)	40
Número Máximo de iteraciones (T_{max})	100
Constantes cognitiva y social	2
ω_{max}	1.4
ω_{min}	0.1
$Iter_{max}$	5,10,15,20,25
V_{max}	4,6,8

Para despliegue de la HetNet [19]	Valor
Número de usuarios secundarios (SU)	20,50,100
Número de usuarios primarios (PU)	6,12,30
Canales primarios	6,12,30
α, β (en dB)	4,6,8,10,12,14
Ancho de banda del canal (B en MHz)	20

Procesador intel® Xeon® E5-2609 v3 (15MB Caché, 1.90 GHz) y sistema operativo Windows® 7 Professional, 64-bit, 32 GB DDR4 a 2133 MHz.

En la Fig. 2 se muestra la convergencia de los resultados encontrados para cada valor de SIR . Cada curva corresponde al promedio de la aptitud de la tasa de datos para un determinado valor de SIR .

En estas curvas se observa que al incrementar el requerimiento de QoS, la tasa de datos promedio disminuye.

Esto sucede debido a que al incrementarse el valor de SIR , la interferencia tolerable en el sistema es menor, lo que dificulta al algoritmo asignar un canal a todos los SUs , por lo que la tasa de datos disminuye.

Tabla 3. Soluciones encontradas para distintos valores de $Iter_{max}$, $SU=20$, $PU=6$, $V_{max}=6$

$\alpha=\beta$ dB	$Iter_{max}$	Aptitud (Mbps)	Enlaces Secundarios seleccionados	iteraciones
4	5	11514,79	20	24
	10	11621.15	20	36
	15	11732.38	20	57
	20	11817.28	20	83
	25	11732.38	20	79
	5	11995.20	20	43
6	10	11219.68	20	69
	15	11533.32	20	90
	20	12057.84	20	88
	25	11533.32	20	1
	5	11465.85	20	50
	10	11345.10	20	47
8	15	11400.56	20	65
	20	11748.95	20	58
	25	11400.56	20	83
	5	10993.44	20	36
	10	10822.56	20	39
	15	11140.75	20	87
10	20	11513.11	20	75
	25	11337.41	20	82
	5	10239.63	20	34
	10	11158.14	20	68
	15	10841.29	20	83
	20	11779.87	20	77
12	25	11468.96	20	81
	5	10300.16	20	46
	10	10778.87	20	64
	15	10825.01	20	46
	20	11053.89	20	45
	25	10862.54	20	72

En las Tablas 3 y 4 se presenta la evaluación del impacto de los parámetros $Iter_{max}$ y V_{max} sobre la técnica MBPSO, respectivamente.

En la Tabla 3 se muestran los resultados de la mejor solución de la tasa de datos obtenida en cada experimento. Se puede observar que al

incrementar el valor de $Iter_{max}$, el valor de aptitud se incrementa, sin embargo para valores de $Iter_{max}=25$, no se observa una mejora significativa, incluso para valores de $SIR=12,14$ dB, el valor de aptitud se reduce, por lo que parece que este parámetro presenta un límite en

Tabla 4. Soluciones encontradas para distintos valores de V_{max} , $SU=20$, $PU=6$, $Iter_{max}=20$

$\alpha=\beta$ dB	V_{max}	Aptitud (Mbps)	Enlaces Secundarios seleccionados	iteraciones
4	4	11544.65	20	58
	6	11717.28	20	83
	8	11852.34	20	52
6	4	11609.82	20	95
	6	11757.84	20	88
	8	11610.90	20	1
8	4	11426.29	20	97
	6	11748.95	20	58
	8	11288.92	20	77
10	4	11147.94	19	90
	6	11337.41	20	75
	8	11289.28	20	76
12	4	11024.31	20	52
	6	11779.87	19	77
	8	10956.57	20	62
14	4	10744.46	19	76
	6	11053.89	18	45
	8	11343.05	19	94

$Iter_{max}=20$ a partir del cual no hay una mejora significativa en la aptitud. Los resultados de la evaluación de aptitud para distintos valores de V_{max} se presentan en la Tabla 4. Los resultados muestran que las mejores soluciones de la tasa de datos para distintos umbrales de SIR se obtiene para valores de $V_{max}=6$.

4.3. Análisis comparativo

Para realizar un análisis comparativo también se implementaron las técnicas MBPSO, SCPSO, AMPSO, ModBPSO para resolver el problema de espectro compartido. A continuación se describe brevemente su operación.

4.3.1. Descripción de ModBPSO y AMPSO

v_{id} cercanos a 0, indica la convergencia del algoritmo y los bits de la partícula se modificarán a pesar de que se realizó una menor exploración.

Además, considerando que SCPSO carece de la habilidad de recordar soluciones de buena calidad ya encontradas, le es difícil converger a una solución óptima debido al incremento de aleatoriedad que se produce conforme las iteraciones avanzan, quedando atrapado en óptimos locales. Al igual que MBPSO, ModBPSO [14] memoriza las mejores soluciones encontradas para que la aleatoriedad de la actualización de posición de la partícula no le afecte, incrementando con ello su velocidad de convergencia.

$$T(v_{id}^{k+1}) = \begin{cases} 1 - \frac{2}{1 + \exp(-v_{id}^{k+1})}, & v_{id}^{k+1} \leq 0 \\ \frac{2}{1 + \exp(-v_{id}^{k+1})} - 1, & v_{id}^{k+1} > 0 \end{cases}, \quad (13)$$

Sin embargo, la modificación principal consiste en utilizar una nueva función de transferencia, definida en (13), que influye significativamente en

evitar que el algoritmo converga prematuramente en óptimos locales. Además en (14), se definen las reglas para actualizar la posición de la partícula;

$$x_{id}^{k+1} = \begin{cases} 0, & \text{if } \text{rand}() \leq T(v_{id}^{k+1}) \text{ and } v_{id}^{k+1} \leq 0 \\ 1, & \text{if } \text{rand}() \leq T(v_{id}^{k+1}) \text{ and } v_{id}^{k+1} > 0. \\ x_{id}^{k+1}, & \text{if } \text{rand}() > T(v_{id}^{k+1}) \end{cases} \quad (14)$$

Para incrementar la diversidad del cúmulo y reducir la probabilidad de que el algoritmo quede atrapado en un óptimo local, se incluye una operación de mutación. Con esto, en la k -ésima iteración, cada posición binaria de la i -ésima partícula cambiará a su valor opuesto en función de una probabilidad de mutación r_{mu} definida en (15):

$$x_{id}^k = \begin{cases} 1 - x_{id}^k, & \text{if } \text{rand}() < r_{mu}, \\ x_{id}^k, & \text{otro} \end{cases} \quad (15)$$

donde $\text{rand}()$ es un número aleatorio distribuido uniformemente entre $[0,1]$. r_{mu} es la probabilidad de que un bit en la partícula cambie de 1 a 0 o de 0 a 1.

Por otro lado, AMPSO [15] no incluye la característica de memorizar la mejor solución encontrada para reducir la convergencia en óptimos locales, sino que aplica una técnica de modulación angular para mapear el problema de optimización binario a espacios de búsqueda continuos y viceversa. Además de incrementar la diversidad del cúmulo, AMPSO mejora la deficiencia que presenta SCPSO para resolver problemas de alta dimensionalidad.

Para problemas de optimización en espacio discreto, en los que la partícula se representa por una cadena de bits, la dimensionalidad se refiere al número de bits que representan una solución del problema a resolver. Problemas de gran dimensión, no solamente incrementan el espacio de búsqueda de posibles soluciones sino que también incrementan la complejidad computacional. Para generar la cadena de bits, los autores en [15] utilizan las expresiones en (16):

$$g(x) = \sin(2\pi(x-a) \times b \times \cos(A)) + e, \quad (16)$$

donde

$$A = 2\pi \times c(x - a), \quad (17)$$

x representa un solo valor (o muestra) de un conjunto de intervalos espaciados uniformemente, este valor depende de la cantidad de bits utilizados en una partícula (esto es, la dimensión de la partícula), a es el desplazamiento horizontal de la función, b y c representan la frecuencia máxima de la función seno y coseno respectivamente y, e representa el desplazamiento vertical de la función. En AMPSO, la técnica PSO convencional se aplica entonces para optimizar la tupla formada por (a,b,c,d) en (16) y (17), en lugar de evolucionar toda la cadena de bits que forma una partícula de n -dimensiones. Una vez que se ejecutaron las iteraciones con PSO, los valores obtenidos de (a,b,c,d) se sustituyen de nuevo en (16) para generar una nueva función de tiempo continuo. En cada intervalo de la función resultante se toman muestras x que permitirán generar un bit (0 o 1) por cada intervalo, el conjunto de bits que se forman por todos los intervalos de la función representa el vector solución (partícula) del problema. A esta nueva partícula se le evalúa su adaptabilidad o fitness de la misma forma como se hace con PSO convencional, la velocidad v_{id} y posición x_{id} , se actualizan mediante (18):

$$\begin{aligned} v_{id}^{k+1} &= wv_{id}^k + (c_1 r_1 (p_{id} - x_{id}^k)) + c_2 r_2 (p_{gd} - x_{id}^k), \\ x_{id}^{k+1} &= x_{id}^k + v_{id}^{k+1}. \end{aligned} \quad (18)$$

4.3.2 Resultados con MBPSO, ModBPSO y AMPSO

Los parámetros de operación utilizados por las técnicas MBPSO, AMPSO y ModBPSO se muestran en la Tabla 5 y, para una comparación justa, los parámetros de evaluación de la HetNet considerados durante la simulación son los mismos que se muestran en la Tabla 2 [12]. La Fig. 3 muestra la convergencia de las mejores soluciones encontradas con cada una de las técnicas de optimización para un escenario de evaluación con $SU=100$, $PU=30$ y $\alpha=\beta=8$ dB. Se puede observar en la Fig.3 que AMPSO y MBPSO presentan una menor velocidad de convergencia

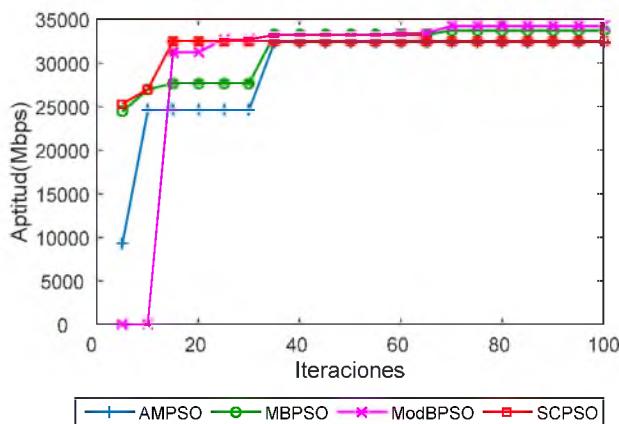


Fig. 3. Convergencia de la aptitud de la HetNet para $SU=20$, $PUs=6$, $\alpha=\beta=8$ dB

en comparación con SCPSO y ModBPSO. Sin embargo, no existe diferencia significativa en las soluciones obtenidas.

Para evaluar la capacidad de la HetNet en términos de la tasa de datos y enlaces secundarios seleccionados, se consideraron dos escenarios, (i) de interferencia baja, representada por pocos usuarios en la red, esto es, 20 femtoceldas y un usuario femtocelular o *SU* en cada una ($SU=20$), y una red macrocelular con 6 usuarios macrocelulares o primarios ($PU=6$), (ii) de interferencia alta, cuando se tienen $SU=100$ y $PU=30$ usuarios en la HetNet. En la Tabla 6 se presenta un comparativo del promedio de la tasa de datos, para distintos valores de umbral S/R , determinada a partir de un total de 500 experimentos.

Los resultados de la Tabla 6 describen la capacidad de la red para el escenario en el que existe poca interferencia, ya que solo se despliegan 20 femtoceldas, las cuales pueden compartir uno de los 6 canales primarios con que cuenta la red macrocelular. De los resultados, se puede observar que con las técnicas MBPSO, AMPSO y ModBPSO se obtienen mejores resultados de la tasa de datos que la HetNet puede transmitir con respecto a cuando se utiliza SCPSO. Considerando el caso de $S/R=10$ dB, la ganancia en la tasa de datos obtenida con MBPSO es del 19.62%, 0.8% y 8.6% con respecto a la obtenida con SCPSO, AMPSO y ModBPSO respectivamente. Además, para todos los casos de S/R evaluados, a diferencia de ModBPSO y

SCPSO, los algoritmos MBPSO y AMPSO encuentran una solución en el que los 20 *SUs* logran compartir un canal primario con alguno de los *PUs* o usuarios macrocelulares desplegados en la HetNet. También se puede observar que, como se reporta en la literatura, SCPSO converge prematuramente en óptimos locales, ya que las soluciones que presenta las obtiene en etapas tempranas de ejecutar la evaluación.

En la Tabla 7 se muestran los resultados de capacidad de la HetNet, para el caso en el que se produce interferencia alta, representada por el despliegue de 100 femtoceldas, con un usuario femtocelular o *SU* en cada una y, en la que cada *SU*, puede compartir uno de los 30 canales primarios con que cuenta la red macrocelular. Otra característica que se presenta con este escenario, es que el problema se vuelve de alta dimensionalidad, en comparación con el caso anterior (resultados de la Tabla 6), ya que cada bit de una partícula identifica un enlace secundario, esto es una conexión entre estación base femtocelular y su correspondiente *SU*. En la Tabla 7 se observa que con la técnica MBPSO, AMPSO y ModBPSO se obtienen mejores soluciones para requerimientos bajos de QoS ($S/R=6,8,10$ dB) en el sistema en comparación con los obtenidos con la técnica SCPSO.

Sin embargo, para valores de $S/R=12,14$ dB, en los que se incrementa el requerimiento de QoS de las redes primaria y secundaria, las técnicas SCPSO y ModBPSO no encuentran una solución (no converge). Mientras que con MBPSO y AMPSO, se obtiene una solución sobre la capacidad del sistema, así como también, encuentran una configuración en la que se asignan uno de los 30 canales primarios a un usuario femtocelular, por ejemplo, para niveles de $S/R=12$ dB, con MBPSO se logra una asignación al 44% de los *SU's* desplegados en la HetNet y con AMPSO del 53%, para un $S/R = 14$ dB es del 38% y 43% respectivamente.

Se debe notar que por su característica de resolver problemas de alta dimensionalidad, AMPSO resuelve el problema de espectro compartido con mejores resultados que los encontrados por MBPSO para niveles altos de QoS (S/R de 12 y 14 dB).

Tabla 5. Parámetros utilizados por las técnicas AMPSO, ModBPSO, MBPSO

Para MBPSO [16]	Valor
$Iter_{max}$	20
V_{max}	6
Para AMPSO [15]	Valor
$[a,b,c,d]$	[0,1,1,0]
intervalos	Igual al valor de SU
Para ModBPSO [14]	Valor
Velocidad $[v_{max}, v_{min}]$	[-6,6]
r_{mu}	0.02

Tabla 6. Evaluación de aptitud para $SU=20$, $PU=6$

$\alpha=\beta$ dB	Técnica	Aptitud (Mbps)	Enlaces secundarios	iteraciones
6	MBPSO	11357.84	20	88
	AMPSO	10987.32	20	75
	ModBPSO	10547.78	20	35
	SCPSO	9286.54	18	1
8	MBPSO	11748.95	20	58
	AMPSO	11545.34	20	34
	ModBPSO	11367.21	20	56
	SCPSO	9546.39	16	13
10	MBPSO	11337.41	20	75
	AMPSO	11245.98	20	89
	ModBPSO	10431.74	20	78
	SCPSO	9477.38	16	9
12	MBPSO	11779.87	20	77
	AMPSO	12032.67	20	64
	ModBPSO	10356.12	19	34
	SCPSO	9323.47	16	18
14	MBPSO	11053.89	20	45
	AMPSO	11256.89	20	89
	ModBPSO	9876.25	18	78
	SCPSO	5856.60	13	23

Es importante resaltar que incrementar los valores de α y β representa un requerimiento mayor de QoS para servicios que así lo demanden. Así, para umbrales de SIR altos, la Tabla 7 sugiere que para mantenerlos, se debe

reducir el número de femtoceldas que deben desplegarse en la HetNet.

Una densidad menor de femtoceldas genera una menor interferencia en la HetNet lo que permite mantener requerimientos altos de QoS. En

Tabla 7. Evaluación de aptitud para $SU=100$, $PU=30$

$\alpha=\beta$ dB	Técnica	Aptitud (Mbps)	Enlaces secundarios	iteraciones
6	MBPSO	36886.17	67	46
	AMPSO	35652.23	68	67
	ModBPSO	33589.12	63	87
	SCPSO	31286.76	51	13
8	MBPSO	34993.85	64	76
	AMPSO	33562.21	62	87
	ModBPSO	34901.19	59	92
	SCPSO	32467.16	44	23
10	MBPSO	28973.70	51	13
	AMPSO	32681.98	53	43
	ModBPSO	27987.65	41	86
	SCPSO	18848.65	27	19
12	MBPSO	22761.72	44	54
	AMPSO	27497.51	53	72
	ModBPSO	No converge	-	-
	SCPSO	No converge	-	-
14	MBPSO	21569.56	38	61
	AMPSO	23763.49	43	78
	ModBPSO	No converge	-	-
	SCPSO	No converge	-	-

contraste, tasas de datos altas solo pueden alcanzarse en condiciones de interferencia baja.

Por ejemplo, observando los resultados de la Tabla 7, si un proveedor de servicio de comunicaciones móviles requiere ofrecer servicios en una HetNet con una QoS de 8 dB (equivalente a servicios de voz) y la red macrocelular solo cuenta con 30 canales disponibles para compartir, entonces solo deben existir como máximo 64 femtoceldas desplegadas.

Por otra parte, si se requiere mantener en la HetNet un $S/I/R$ mayor o igual a 14 dB, entonces el proveedor solo deberá desplegar un máximo de 43 femtoceldas en el sistema.

5. Conclusiones

En este trabajo se resolvió un problema de espectro compartido aplicando las técnicas MBPSO, AMPSO, ModBPSO. Los resultados demuestran que es posible que distintas redes comparten espectro simultáneamente, y en consecuencia se logre incrementar el uso del espectro. Además, se demuestra que al mejorar la característica de exploración y explotación de las técnicas heurísticas basadas en PSO, se obtienen mejores soluciones. Los resultados demuestran que las técnicas MBPSO y AMPSO mantienen el compromiso entre lograr una máxima tasa de

datos del sistema con mínima interferencia para los casos de interferencia baja y alta y para exigencias de QoS desde 6 dB hasta 14 dB.

Para algunos casos, las soluciones obtenidas con MBPSO mejoran aproximadamente en un 25% la tasa de datos de la red obtenida con SCPSO, mientras que en otros casos es del 80% ($SU=20$, $PU=6$, $SIR=12\text{dB}$). MBPSO y AMPSO son técnicas confiables que puede aplicarse para aquellos casos en que la complejidad computacional del problema es alta.

Referencias

1. Liu, Y., Cai, L.X., Shen, X., & Luo, H. (2013). Deploying cognitive cellular networks under dynamic resource management. *IEEE Wireless Communications*, Vol. 20, No. 2, pp.82–88. DOI: 10.1109/MWC.2013.6507398.
2. Jorswieck, E.A., Badia, L., Fahldieck, T., Karipidis, E. & Luo, J. (2014). Spectrum sharing improves the network efficiency for cellular operators. *IEEE Communications Magazine*, Vol. 52, No. 3, pp. 129–136.
3. Wang, W., Yu, G., & Huang, A. (2013). Cognitive radio enhanced interference coordination for femtocell networks. *IEEE Communications Magazine*, Vol. 51, No. 6, pp. 37–43. DOI: 10.1109/MCOM.2013.6525593.
4. Tadrous, J., Sultan, A., & Nafie, M. (2011). Admission and Power Control for Spectrum Sharing Cognitive Radio Networks. *IEEE Transactions on Wireless*, Vol. 10, No. 6, pp. 1945–1955. DOI: 10.1109/TWC.2011.040411.101571.
5. Erpek, T., McHenry, M.A., & Stirling, A. (2011). Dynamic spectrum access operational parameters with wireless microphones. *IEEE Communications Magazine*, Vol. 49, No. 3, pp. 38–45. DOI: 10.1109/MCOM.2011.5723798.
6. Senthuran, S., Anpalagan, A., & Das, O. (2012). Throughput Analysis of Opportunistic Access Strategies in Hybrid Underlay-Overlay Cognitive Radio Networks. *IEEE Transactions on Wireless Communications*, Vol. 11, No. 11, pp. 2024–2035. DOI: 10.1109/TWC.2012.032712.101209.
7. Hamouda, S., Zitoun, M., & Tabbane, S. (2013). A new spectrum sharing trade in Heterogeneous Networks. *IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1–5. DOI: 10.1109/VTCFall.2013.6692046.
8. Waheed, M. & Cai, A. (2009). Evolutionary algorithms for radio resource management in cognitive radio network. *Performance Computing and Communications Conference (IPCCC 2009)*, pp. 431–436.
9. Fei, W. & Zhen, Y. (2008). An ACO Algorithm for Sum-Rate Maximization Problem in Opportunistic Spectrum Allocation. *4th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM'08)*, pp.1–4. DOI: 10.1109/WiCom.2008.322.
10. Jiang, H., Li, P., Li, Z., Tong, E., Pan, Z., Liu, N., & You, X. (2014). Improved MPSO based eICIC algorithm for LTE-a ultra dense HetNets. *Global Communications Conference (GLOBECOM), IEEE*, pp. 3678–3683. DOI: 10.1109/GLOCOM.2014.7037379.
11. Kennedy, J. & Eberhart, R. (1995). Particle swarm optimization. *IEEE International Conference on Neural Networks*, Vol. 4, pp. 1942–1948.
12. Martínez-Vargas, A. & Andrade, Á.G. (2013). Comparing Particle swarm optimization variants for a cognitive radio network. *Applied Soft Computing*, Vol. 13, No. 2, pp. 1222–1234. DOI: 10.1016/j.asoc.2012.10.016.
13. Arani, B., Mirzabeygi, P., & Panahi, M. (2013). An Improved PSO Algorithm with a Territorial Diversity-Preserving Scheme and Enhanced Exploration-Explotation Balance. *Swarm and Evolutionary Computation*, Vol. 11, pp. 1–15. DOI: 10.1016/j.swevo.2012.12.004.
14. Mirjalili, S. & Lewis, A. (2013). S-shaped versus V-shaped transfer functions for binary particle swarm optimization. *Swarm and Evolutionary Computation*, Vol. 9, pp. 1–14. DOI: 10.1016/j.swevo.2012.09.002.
15. Pampara, G., Franken, N., & Engelbrecht, A.P. (2005). Combining particle swarm optimisation with angle modulation to solve binary problems. *IEEE Congress on Evolutionary Computation*, pp. 89–96. DOI: 10.1109/CEC.2005.1554671.
16. Ji, Z., Tian, T., He, S., & Zhu, Z. (2012). A memory binary particle swarm optimization. *IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–5.
17. Deep, K. & Bansal, J.C. (2008). A Socio-Cognitive Particle Swarm Optimization for Multi-Dimensional Knapsack Problem. *Emerging Trends in Engineering and Technology Conference (ICETET)*, pp. 355–360. DOI: 10.1109/ICETET.2008.163.
18. Nápoles, G., Grau, I., Bello, M., & Bello, R. (2014). Toward Swarm Diversity: Random Sampling in Variable Neighborhoods Procedure Using a Lévy Distribution. *Computación y Sistemas*, Vol. 18, No. 1, pp. 79–95. DOI: 10.13053/CyS-18-1-2014-020.

- 19. 3GPP (2010).** 3rd generation partnership project; technical specification group radio access network; evolved universal terrestrial radio access (EUTRA); further advancements for E-UTRA physical layer aspects (release 9), *Tech. Spec. 36.814 V9.0.0.*

Esteban Martínez realizó estudios de Ingeniería en Cibérnetica Electrónica en el Centro de Enseñanza Técnica y Superior (CETYS Universidad) de Mexicali, Baja California. Actualmente está por obtener su grado de maestría en ciencias en la Universidad Autónoma de Baja California (UABC).

Ángel G. Andrade obtuvo su Doctorado y Maestría en Ciencias en Electrónica y Telecomunicaciones en el Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE), México. Actualmente es profesor-investigador en la Facultad de Ingeniería de la Universidad Autónoma de Baja California

(UABC). Sus líneas de investigación se centran en el análisis y desarrollo de estrategias para lograr la compartición de espectro entre sistemas inalámbricos. Es miembro del Sistema Nacional de Investigadores (SNI) de CONACYT.

Anabel Martínez-Vargas realizó estudios de Ingeniería en Sistemas Computacionales en el Instituto Tecnológico de San Luis Potosí, México. Tiene Maestría en Computación con especialidad en Redes y Conectividad por la Universidad de Colima, México. Es Doctora en Ciencias por la Universidad Autónoma de Baja California, México. Pertenece al Sistema Nacional de Investigadores (SNI) de CONACYT. Actualmente es investigadora en la Universidad Politécnica de Pachuca (UPP). Sus líneas de interés son metaheurísticas y optimización aplicados al problema de acceso dinámico de espectro.

*Artículo recibido 25/09/2015; aceptado 12/01/2016.
Autor de correspondencia es Esteban Martínez.*