

Algoritmo de Colonias de Abejas Artificiales para la composición de equipos médicos*

José A. Delgado-Osuna¹, Manuel Lozano², and Carlos García-Martínez³

¹ H.U. Reina Sofía

`jantonio.delgado.sspa@juntadeandalucia.es`,

² Dpto. de Ciencias de la Computación e I. A., Universidad de Granada

`lozano@decsai.ugr.es`,

³ Dpto. de Informática y Análisis Numérico, Universidad de Córdoba

`cgarcia@uco.es`

Resumen En este trabajo se propone un algoritmo de colonias de abejas artificiales para el problema de la composición de equipos médicos con los principios de equidad y eficiencia. El objetivo principal consiste en componer grupos de profesionales con diferentes habilidades y que al mismo tiempo sean lo más eficiente posible. El algoritmo de abejas artificiales es un modelo de *swarm intelligence* inspirado en el comportamiento de las abejas cuando realizan la búsqueda de comida. Nuestra propuesta explota el conocimiento útil sobre el problema que se obtiene de la exploración del vecindario de las fuentes de comida, aplicando una destrucción parcial de las soluciones escogidas y una reconstrucción heurística de las mismas. Los resultados se han comparado con los del algoritmo estado del arte para este problema y con los de un algoritmo genético.

Keywords: Asistencia Sanitaria, Algoritmo de Colonia de Abejas Artificiales, Optimización

1. Introducción

La asistencia sanitaria es un campo de aplicación muy complejo y de muy alta prioridad, cuya prestación implica problemas con un gran número de restricciones y de incertidumbre, numerosas variables de decisión relacionadas, objetivos que entran en conflicto y, por supuesto, depende de interacciones entre personas. En [2], Aringhieri ilustra el problema de la Composición de Equipos Médicos desde el punto de vista de los principios de Equidad y Eficiencia (CEM^{ee}). En este contexto, la equidad hace referencia a la prestación de los mismos servicios a cualquier persona, con independencia de sus circunstancias tales como edad, ingresos, lugar de residencia, etc. Por otro lado, la eficiencia busca la prestación de servicios sanitarios de alta calidad.

* Este trabajo ha sido financiado por el proyecto TIN2012-37930 del Ministerio de Economía y Competitividad.

El algoritmo de colonia de abejas artificiales [1,6,7] (ABC; *Artificial Bee Colony*) es una metaheurística basada en poblaciones relativamente nueva que se inspira en el comportamiento de los enjambres de abejas cuando buscan comida. Consta de un conjunto de fuentes de comida y de tres clases de abejas (obreras, supervisoras y *scouts*). Las fuentes de comida representan soluciones al problema tratado. Su posición codifica la configuración de la solución mientras que la calidad está asociada con la cantidad de néctar que tiene. Las abejas simbolizan agentes que operan sobre las fuentes de comida para generar nuevas soluciones candidatas. Las obreras exploran el vecindario de la fuente que tienen asignada y comparten información con las supervisoras. Éstas tienden a seleccionar buenas fuentes de comida a partir de las cuales continuar con la exploración del vecindario. Las *scouts* se encargan de buscar nuevas fuentes candidatas de una forma más exploradora, normalmente considerando el espacio de soluciones completo del problema. Gracias a su éxito y simplicidad, el ABC ha sido centro de atención en los últimos tiempos y se ha utilizado en las propuestas del estado del arte de muchos problemas [8]. Nuestro objetivo con este trabajo es desarrollar un algoritmo ABC competitivo para el problema CEM^{ee} .

El resto del artículo se estructura de la siguiente manera: la Sección 2 describe el problema CEM^{ee} e introduce el algoritmo estado del arte para el mismo. La Sección 3 detalla nuestra propuesta de algoritmo ABC para el problema. La Sección 4 expone los experimentos que analizan nuestro ABC- CEM^{ee} en diferentes contextos que tienen una dificultad cada vez mayor. El modelo se compara además con un algoritmo genético y con el algoritmo estado del arte para este problema. La Sección 5 finaliza con las conclusiones del trabajo.

2. Composición de Equipos Médicos con Equidad y Eficiencia

Aringhieri formuló el problema como sigue a continuación [2]. Sea P un conjunto de profesionales sanitarios. Las eficiencias individuales de los profesionales están determinadas por un valor real $e_p \in \mathbb{R}_+$ ($p = 1, \dots, |P|$), que representa con qué eficiencia realiza su trabajo el individuo p . La diversidad entre las habilidades de los profesionales p y q se representa por parejas de valores $d_{pq} \in \mathbb{R}_+$ ($p, q \in \{1, \dots, |P|\}$; $d_{pq} = d_{qp}$ and $d_{pp} = 0$), que indican cómo de heterogéneas son esas habilidades. Por lo tanto, el objetivo es componer T equipos (C_t , $t = 1, \dots, T$), cada uno con M_t miembros, donde las diversidades de los equipos, $D_t = \frac{1}{2} \sum_{p \in C_t} \sum_{q \in C_t} d_{pq}$, superen un cierto umbral D_{min} , y las eficiencias de los mismos, $E_t = \sum_{p \in C_t} e_p$, se maximicen. La Figura 1 muestra una instancia del problema y dos soluciones. Suponiendo que $D_{min} = 15$, la solución B es mejor que la A porque, aunque la eficiencia del segundo equipo es menor, la solución A no satisface la restricción de diversidad.

La formulación matemática del problema es la que se muestra en la Figura 2, donde cada variable binaria x_{pt} indica si un profesional p está asignado al equipo C_t ($x_{pt} = 1$) o no ($x_{pt} = 0$). Las restricciones (2) garantizan que ningún profesional está asignado a más de un equipo. Las restricciones (3) verifican que cada

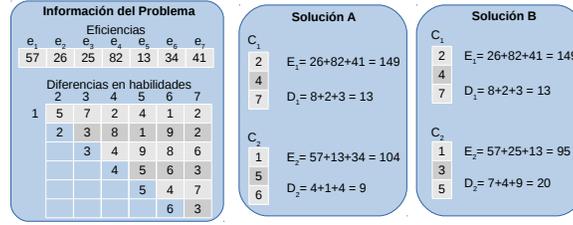


Figura 1. Instancia del CEM^{ee} y dos soluciones.

equipo tiene el número correcto de miembros. Las restricciones (4) establecen el umbral de diversidad a alcanzar por los equipos. Finalmente, el objetivo es maximizar las eficiencias de los equipos maximizando aquella con el valor más bajo (Ecuación 1).

$$\begin{aligned} \max_{t=1, \dots, T} \quad & \min_{p=1, \dots, |P|} \sum_{p=1}^{|P|} e_p x_{pt} & (1) \\ \text{sujeto a:} \quad & \sum_{t=1}^T x_{pt} \leq 1, \quad \forall p = 1, \dots, |P| & (2) \\ & \sum_{p=1}^P x_{pt} = M_t, \quad \forall t = 1, \dots, T & (3) \\ & \sum_{p=1}^{P-1} \sum_{q=p+1}^P d_{pq} x_{pt} x_{qt} \geq D_{min}, \quad \forall p, q = 1, \dots, |P|, \quad \forall t = 1, \dots, T & (4) \\ & x_{pt} \in \{0, 1\}, \quad \forall p = 1, \dots, |P|, \quad \forall t = 1, \dots, T & (5) \end{aligned}$$

Figura 2. Modelo matemático para el problema CEM^{ee} [2]

El único método para este problema, a nuestro conocimiento, es un algoritmo híbrido que combina una heurística voraz de inicialización, un optimizador local y una búsqueda tabú (HVBT) [2], que mostró un mejor rendimiento que la resolución del modelo matemático con CPLEX. El algoritmo voraz genera una solución candidata introduciendo iterativamente en la solución parcial la mejor asignación esperada. Esta asignación incluye un profesional libre en un equipo, que aún no está completo, de tal modo que se incremente la eficiencia y la diversidad del equipo. A continuación, el optimizador local y la búsqueda tabú exploran el vecindario de la solución para intentar conseguir mejores configuraciones, intercambiando profesionales entre distintos equipos, o uno de un equipo y otro aún no asignado a ninguno. Además, la búsqueda tabú utiliza un sistema de penalizaciones para explorar también soluciones que no satisfacen todas las restricciones.

Es interesante destacar que la búsqueda de diversidad en el CEM^{ee} se sitúa en la satisfacción de restricciones de un umbral mínimo, a diferencia de la mayoría

de problemas de diversidad encontrados en la literatura [9,11], en los que se sitúa en la función objetivo, frecuentemente en la búsqueda de una diversidad máxima.

3. ABC para la Composición de Equipos Médicos

El algoritmo de nuestro modelo, $ABC-CEM^{ee}$, comienza generando un conjunto de NP fuentes de comida, ubicando profesionales de forma aleatoria en los equipos hasta satisfacer las restricciones (3; $|C_t| = M_t$). Estas soluciones se optimizan posteriormente mediante un procedimiento de búsqueda local con una probabilidad p_{ls} . A continuación, el $ABC-CEM^{ee}$ itera por las diferentes etapas hasta alcanzar el criterio de parada:

- *Fase de las abejas obreras*: cada obrera explora las proximidades de su fuente de comida (Sección 3.1). La nueva solución se optimiza localmente con probabilidad p_{ls} y se acepta si es mejor que la fuente de comida previa.
- *Fase de las abejas supervisoras*: estas abejas escogen una fuente de comida aplicando un torneo binario. A continuación, las supervisoras aplican los mismos pasos que las abejas obreras.
- *Fase de las abejas scout*: las fuentes de comida que no se han mejorado durante L_{iter} iteraciones consecutivas son abandonadas y reemplazadas por una nueva solución generada de forma aleatoria, la cual es optimizada con una probabilidad p_{ls} .

Al final, el algoritmo devuelve la mejor solución generada, que es aquella que, satisfaciendo todas las restricciones, maximiza la eficiencia del equipo que tiene menor valor de la misma.

Nuestro procedimiento de búsqueda local comparte la idea expuesta en [2] de mejorar al peor equipo. Sin embargo, el método que se expone en [2] se centra en incrementar el criterio de eficiencia de la solución, mientras que el nuestro prioriza el cumplimiento de las restricciones de diversidad (4) y una vez satisfechas, busca mejores resultados de eficiencia.

3.1. Operador de vecindario destructivo-constructivo

En este trabajo aplicamos la metodología, introducida por Rodríguez et al. [13], de una destrucción parcial y una reconstrucción heurística de la fuente de comida para explorar su vecindario. En concreto, las abejas obreras observan las diferencias que existen entre su fuente de comida actual y otra escogida aleatoriamente. En este contexto, las diferencias se definen como profesionales asignados a diferentes equipos. Las asignaciones distintas se eliminan de la fuente de comida actual, generando así una solución incompleta. Después, la solución parcial se reconstruye aplicando el operador voraz heurístico utilizado en [2]. Aringhieri aplica este método sobre una solución vacía para generar una inicial completa. Es necesario destacar que este método es determinístico, salvo por la aleatoriedad con la que se resuelven los empates, lo cual hace que diferentes invocaciones con soluciones vacías puedan producir resultados similares. Sin

embargo, en nuestro caso, el método está condicionado por la solución parcial provista, por lo que diferentes invocaciones con diferentes soluciones parciales pueden producir resultados muy distintos.

3.2. Manejo de restricciones en ABC-CEM^{ee}

Cuando las abejas obreras y las supervisoras generan nuevas soluciones candidatas, a través del operador de vecindario destructivo-constructivo o de la posterior optimización local, éstas podrían no satisfacer las restricciones de diversidad (4). Aún así, las abejas obreras y supervisoras tienen que evaluar si actualizan la fuente de comida o si descartan la nueva solución. Se pueden dar tres situaciones posibles, que se detallan en [3]:

- Si ambas soluciones satisfacen las restricciones del problema, la abeja selecciona la solución que tiene el mayor valor objetivo, es decir, aquella que tiene mayor eficiencia (Ecuación (1)).
- Si sólo una solución satisface las restricciones del problema, esa es la seleccionada por la abeja con independencia de su valor objetivo.
- Si ambas soluciones violan las restricciones de diversidad, su calidad es penalizada en proporción a la severidad de dichas violaciones. La solución escogida será aquella cuyo valor objetivo penalizado sea mayor:

$$f' = \left(\min_{t=1, \dots, T} E_t \right) - \left(\sum_{t=1}^T \max(0, D_{min} - D_t) \right) \quad (6)$$

4. Experimentos

En esta sección se detallan los experimentos que se han realizado para evaluar los beneficios de nuestro algoritmo ABC-CEM^{ee}. La Sección 4.1 describe la configuración de los experimentos. Nuestros objetivos son: (1) determinar la influencia de la configuración de parámetros sobre el ABC-CEM^{ee} (Sección 4.2) y (2), comparar los beneficios de ABC-CEM^{ee} con los de otras metaheurísticas, concretamente con un algoritmo genético generacional (AGG) y con el estado del arte para el problema, HVBT [2] (Sección 4.3).

4.1. Configuración de los experimentos

Nuestro ABC ha sido implementado en Java SE 6. Los experimentos se han ejecutado en un PC con un procesador de cuatro núcleos Intel(R) CoreTM i5 a 2.9GHz, con 8GB de RAM y con sistema operativo MAC OS X 10.9.3 64-bits. Cada ejecución del algoritmo se ha llevado a cabo utilizando un único hilo durante $|P|/2$ segundos como máximo (criterio de parada).

Hemos utilizado las mismas 80 instancias del problema descritas en [2]⁴ y tres umbrales de diversidad distintos ($D_{min}^{\%}$), haciendo un total de 240 casos

⁴ Las instancias del problema se encuentran disponibles en <http://www.uco.es/grupos/kdis/kdiswiki/index.php/ABC-MCEE>

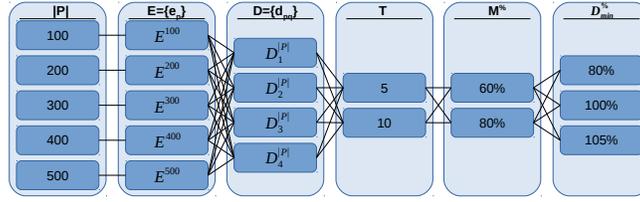


Figura 3. Valores de los parámetros de las instancias del problema consideradas

(ver Figura 3). Estas instancias están compuestas por la combinación del número de profesionales, $|P| \in \{100, 200, 300, 400, 500\}$, un vector de eficiencias, $E^{|P|}$, y cuatro matrices de diversidad, $D_i^{|P|}$ con $i \in \{1, 2, 3, 4\}$, para cada número de profesionales, número de equipos a componer $T \in \{5, 10\}$, número de profesionales requeridos por equipo calculado como un porcentaje de la ratio de profesionales por equipo, $M = M\% \cdot \frac{|P|}{T}$ and $M\% \in \{60\%, 80\%\}$, y porcentaje de la diversidad mínima requerida sobre la diversidad media esperada, obtenida mediante la Ecuación (7), $D_{min}^{\%} \in \{80\%, 100\%, 105\%\}$.

$$D_{min} = D_{min}^{\%} \frac{M \cdot (M - 1) \cdot \bar{D}}{2}, \text{ con } \bar{D} = \frac{\sum_{p,q \in P} d_{pq}}{|P| \cdot (|P| - 1)} \quad (7)$$

Las instancias del problema han sido clasificadas y analizadas por separado en función de este último parámetro en *fácil* ($D_{min}^{\%} = 80\%$), *exigente* ($D_{min}^{\%} = 100\%$) y *difícil* ($D_{min}^{\%} = 105\%$).

Para analizar el significado de los resultados experimentales hemos utilizado el test de Wilcoxon, que compara el resultado de dos algoritmos sobre diferentes problemas. En todos los casos, hemos considerado un factor de relevancia estadística igual al 5% y la mínima eficiencia del equipo (Ecuación 1) cuando todos los equipos satisfacen la restricción de diversidad (Restricciones 4), y 0 cuando las restricciones no se satisfacen.

4.2. Ajuste de parámetros

De cara a encontrar una buena configuración del algoritmo $ABC-CEM^{ee}$, se ha realizado un estudio sobre las combinaciones de los siguientes valores: $NP = 20$, $p_{ls} = \{0, 0, 1, 0, 5\}$ y $L_{iter} = \{20, 50\}$. Los resultados, no presentados aquí por las restricciones de espacio, mostraron que la configuración más competitiva es aquella en la que la probabilidad de búsqueda local es $p_{ls} = 0$ (no aplica optimización local) y el máximo número de iteraciones sin mejorar una fuente de comida antes de abandonarla es $L_{iter} = 20$. De aquí podemos concluir que dedicar los recursos computacionales a la ejecución de nuestro operador de vecindario destructivo-constructivo es claramente más ventajoso que dedicarlos a la búsqueda local. Hemos escogido esta configuración para el resto de experimentos y haremos referencia a ella como $ABC-CEM^{ee}$ para simplificar.

4.3. Comparación con otras metaheurísticas

Hemos implementado otros dos métodos: AGG, que es un algoritmo genético diseñado por nosotros para este problema, y HVBT [2], que es el estado del arte para el problema CEM^{ee} . AGG es una algoritmo genético generacional con elitismo y una población de 80 cromosomas. En cada iteración, se aplica una selección por torneo binario para escoger 80 parejas de soluciones, las cuales van a producir nuevas soluciones candidatas para la población de la siguiente generación con una probabilidad de cruce de 0.8. Si no se produce el cruce, el primer padre seleccionado es el que pasa a la siguiente generación. Se ha utilizado como operador de cruce nuestro operador de vecindario destructivo-constructivo. A continuación, los genes de los cromosomas son mutados con una probabilidad $1/|P|$, asignando de esta manera profesionales a equipos escogidos de forma aleatoria. Dado que estos operadores podrían generar soluciones no factibles respecto a las Restricciones (3), demasiados o muy pocos profesionales en algunos equipos, se utiliza después un operador de reparación que, en primer lugar, elimina aleatoriamente profesionales de los equipos sobrecargados hasta que ningún equipo tenga más de M_t profesionales y, después, inserta aleatoriamente profesionales aún no asignados a equipos no completos hasta que tengan exactamente M_t profesionales.

Ambas metaheurísticas se ejecutaron sobre las instancias del problema con la misma configuración experimental descrita en la Sección 4.1. El ajuste de parámetros de HVBT es el que se indica en [2] (excepto para el criterio de parada, que se ha establecido en $|P|/2$ segundos igual que en $ABC-CEM^{ee}$ y AGG). Los resultados globales de los métodos HVBT y $ABC-CEM^{ee}$ se presentan en la Tabla 1. Los de AGG se omiten por restricciones de espacio y ser significativamente inferiores. Para cada escenario del problema (global, fácil, exigente y difícil), se muestra el porcentaje de instancias para las que se han obtenido soluciones factibles con cada método ($\%ABC-CEM^{ee}$ y $\%HVBT$), junto con los resultados del test estadístico de Wilcoxon al comparar los métodos. Si $R-$ (asociado a HVBT) es menor que $R+$ (asociado con $ABC-CEM^{ee}$) y que el valor crítico, entonces el test encuentra diferencias estadísticas a favor de $ABC-CEM^{ee}$; si $R+$ es menor que $R-$ y que el valor crítico, las diferencias favorecen a HVBT; si tanto $R+$ como $R-$ no son menores que el valor crítico, entonces el test no encuentra diferencias estadísticas. La última columna indica si $ABC-CEM^{ee}$ es mejor (+), peor (-) o sin diferencias significativas (~) que HVBT.

Cuadro 1. $ABC-CEM^{ee}$ versus HVBT

	$\%ABC-CEM^{ee}$	$\%HVBT$	$R+$	$R-$	Valor Crítico	Diff
Global	62.92 %	62.08 %	23057	5863	12349.775	+
$D_{min}^{\%} = 80 \%$	100 %	100 %	3112	128	1211.358	+
$D_{min}^{\%} = 100 \%$	83.75 %	86.25 %	2341	899	1211.358	+
$D_{min}^{\%} = 105 \%$	5 %	0 %	1777	1463	1211.358	~

Los resultados de la Tabla 1 muestran que HVBT y ABC-CEM^{ee} consiguen resolver casi el mismo porcentaje de instancias del problema, aunque HVBT no puede resolver ninguno de los casos más difíciles. El test de Wilcoxon indica que ABC-CEM^{ee} consigue estadísticamente mejores resultados que HVBT en todos los escenarios excepto en el más difícil. Aunque ambos métodos generan soluciones factibles que satisfacen las Restricciones (2-5) en las mismas instancias, ABC-CEM^{ee} es capaz de mejorar aún más la eficiencia de los equipos. Sin embargo, dados los malos resultados que se obtienen en el escenario más difícil, el test no puede encontrar diferencias de rendimiento relevantes en este caso (última fila de la tabla).

Las gráficas de la Figura 4 muestran la convergencia del rendimiento medio de los algoritmos normalizada en el intervalo [0,1] sobre las instancias del problema, globalmente y separadas atendiendo a la dificultad. Adicionalmente, las gráficas de la Figura 5 muestran la diversidad media normalizada en el intervalo [0,1].

Se observa que HVBT comienza el proceso de búsqueda con mejores resultados tanto de eficiencia como de diversidad respecto a ABC-CEM^{ee} y a AGG en casi todos los casos. Esto se debe a que HVBT aplica el método constructor voraz y un optimizador local para generar la solución inicial de la fase tabú, mientras que ABC-CEM^{ee} y AGG generan las soluciones iniciales de forma aleatoria. Una diferencia particular es que HVBT no tiene como objetivo encontrar equipos más diversos una vez que el umbral mínimo se ha alcanzado. Incluso, en algunos casos, HVBT sacrifica la diversidad de los equipos si con ello obtiene otros más eficientes (ver la gráfica para $D_{min}^{\%} = 80\%$ en la Figura 5). Sin embargo, ABC-CEM^{ee} y AGG continúan mejorando la diversidad de los equipos durante el proceso de búsqueda. Esto se debe a que se aplica el operador destructivo-constructivo repetidamente, el cual busca equipos eficientes y diversos. Creemos que esta característica contribuye a que ABC-CEM^{ee} satisfaga la restricción de diversidad en cantidades mayores, lo cual puede ser posteriormente beneficioso para mejorar la eficiencia sin el riesgo de violar la restricción de diversidad. Finalmente, esto permite a ABC-CEM^{ee} obtener mejores resultados que HVBT, incluso partiendo de resultados peores.

Si comparamos AGG y ABC-CEM^{ee}, observamos que ABC-CEM^{ee} hace mejor uso del operador de vecindario, principalmente debido a las siguientes tres diferencias estructurales entre ellos:

- El marco de trabajo del ABC asegura que todas las soluciones de la población se sometan al operador de vecindario al menos una vez por iteración, en la fase de las abejas obreras, mientras que esto es más estocástico en AGG debido al operador de selección.
- Las soluciones de la población del AGG son demasiado rápidamente reemplazadas por nuevas (aunque la probabilidad de cruce sea inferior a 1 y se incluya elitismo). Por contra, parece que se obtienen mejores resultados a través de la aplicación repetida del operador sobre las mismas soluciones o sobre las soluciones mejoradas. Esto ocurre con mayor frecuencia en ABC-CEM^{ee} porque las soluciones sólo se abandonan en la fase de las abejas scout, menos frecuentemente que en AGG.

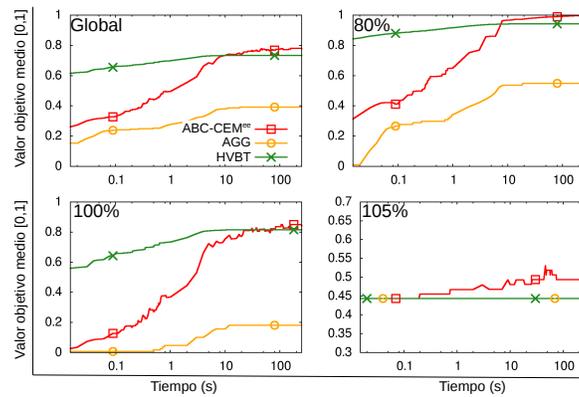


Figura 4. Gráficas de convergencia normalizadas del ABC, HVBT y AGG

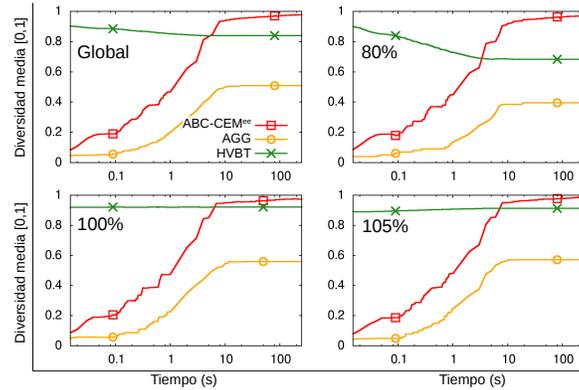


Figura 5. Diversidad de los equipos normalizada para ABC, HVBT y AGG

- El marco del ABC ofrece un componente natural para evitar el estancamiento, la fase de las abejas scout, más efectiva en este caso que el operador de mutación del AGG.

5. Conclusiones

Hemos propuesto un algoritmo de colonias de abejas artificiales para el problema de la composición de equipos médicos con los principios de equidad y eficiencia. Nuestro método aplica un operador de vecindario destructivo-constructivo para generar nuevas soluciones candidatas cuando las abejas exploran el vecindario de las fuentes de comida. Este operador busca asignaciones de profesionales a equipos que incrementen tanto la diversidad como la eficiencia de dichos equipos. Se ha demostrado que la propuesta de ABC diseñada es

competitiva con respecto al algoritmo del estado del arte para este problema. Por lo tanto, concluimos que esta metaheurística es una buena opción para este problema.

Creemos que el marco del ABC con el operador de vecindario destructivo-constructivo debe ser estudiado con más profundidad, diseñando operadores específicos para otros problemas como la planificación de máquinas paralelas [12] o el problema de la mochila cuadrática múltiple [4], y de analizar el diseño híbrido de algoritmos ABC con métodos de optimización relevantes tales como el enfriamiento simulado [5]. También sería interesante estudiar la inclusión de más información en relación con la composición de equipos médicos y otros grupos de profesionales, como por ejemplo, el uso de etiquetas difusas para indicar la eficiencia y la diversidad de los profesionales [10], o su caracterización mediante vectores multidimensionales de características.

Referencias

1. B. Akay and D. Karaboga. A modified artificial bee colony algorithm for real-parameter optimization. *Information Sciences*, 192:120–142, 2012.
2. R. Aringhieri. Composing Medical Crews with Equity and Efficiency. *Central European Journal of Operations Research*, 17(3):343–357, 2009.
3. K. Deb. An efficient constraint handling method for genetic algorithms. *Computer methods in applied mechanics and engineering*, 186(2):311–338, 2000.
4. C. García-Martínez, F. Glover, F. J. Rodríguez, M. Lozano, and R. Martí. Strategic oscillation for the quadratic multiple knapsack problem. *Computational Optimization and Applications*, 58:161–185, 2014.
5. C. García-Martínez, M. Lozano, and F. J. Rodríguez. A simulated annealing method based on a specialised evolutionary algorithm. *Applied Soft Computing*, 12(2):573–588, 2012.
6. D. Karaboga and B. Akay. A modified artificial bee colony (ABC) algorithm for constrained optimization problems. *Applied Soft Computing*, 11:3021–3031, 2011.
7. D. Karaboga and B. Basturk. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39:459–471, 2007.
8. D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga. A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, 42:21–57, 2014.
9. R. Martí, M. Gallego, A. Duarte, and E. Pardo. Heuristics and metaheuristics for the maximum diversity problem. *Journal of Heuristics*, 19(4):591–615, 2013.
10. D. Millán-Ruiz and J. Hidalgo. A Memetic Algorithm for Workforce Distribution in Dynamic Multi-Skill Call Centres. In *Proc. of the European Conference on Evolutionary Computation in Combinatorial Optimisation*, pages 178–189, 2010.
11. O. Prokopyev, N. Kong, and D. Martinez-Torres. The equitable dispersion problem. *European Journal of Operational Research*, 197:59–67, 2009.
12. F. J. Rodríguez, M. Lozano, C. Blum, and C. García-Martínez. An iterated greedy algorithm for the large-scale unrelated parallel machines scheduling problem. *Computers and Operations Research*, 40(7):1829–1841, 2013.
13. F. J. Rodríguez, M. Lozano, C. García-Martínez, and J. D. González-Barrera. An artificial bee colony algorithm for the maximally diverse grouping problem. *Information Sciences*, 230(0):183–196, 2013.