

## **Resumen**

Al concluir un proceso de selección de personal, lo ideal sería encontrar la *mejor* asignación de personas a puestos. Para ello es necesario encontrar una composición que logre *buenos* valores de criterios, tales como la suma de las idoneidades de cada persona con el puesto a ocupar. La cantidad de combinaciones posibles, en dimensiones relativamente significativas de puestos a cubrir y empleados disponibles (situación típica en empresas medianas o grandes), puede convertirse en un problema con un espacio de soluciones demasiado grande. Esto hace que esta etapa sea prácticamente imposible de abordar, corriendo el riesgo de no hacer la asignación más adecuada de acuerdo a criterios preestablecidos. Este problema de asignación es complejo y necesita optimización.

Una gran cantidad de problemas de optimización no pueden ser resueltos usando métodos exactos, es decir, no es posible encontrar su solución óptima con esfuerzos computacionales aceptables aunque se cuenten con computadoras de alta velocidad (incluso operando en paralelo). Sin embargo, no todos los problemas combinatoriales son tan complejos de resolver y encuentran soluciones aceptables con determinadas técnicas heurísticas. Este trabajo tiene como objetivo analizar algunas de las técnicas heurísticas basadas en un punto, tales como: Escalador de Colinas (clásico, estocástico con primer ascenso, estocástico con mejor ascenso, con reiniciación), Recocido Simulado y la Búsqueda Tabú; con el objetivo de proponer un método que sirva de soporte a la toma de decisiones en un proceso de selección de personal.

**Palabras clave:** Algoritmos heurísticos, Escalador de colinas, Selección de personal.

## **Abstract**

*At the conclusion of a staff selection process, the ideal would be to find the best allocation of persons to employ. This requires finding a composition that achieves good values of criteria, such as the sum of the suitability of each person to occupy the employ. The number of possible combinations, in relatively significant dimensions of positions to fill and employees available (typical situation in medium or large companies) can be a problem with a space too large for solutions. That makes this stage it is practically impossible of approaching, at the risk of not making the most appropriate allocation according to predetermined criteria. This issue of allocation is complex and requires optimization.*

*A large number of optimization problems can not be solved using accurate, that is, you can not find your optimal solution with computational effort is acceptable though computers have high-speed (even operate in parallel). However, not all are so complex combinatorial problems to solve and find acceptable solutions with heuristic techniques. This work has as objective to analyze some of the techniques based on a heuristic point, such as: Hill Climbing (classical, with stochastic first ascent, stochastic with better promotion, with resumption), and Simulated Annealing , Tabu Search, with the objective of proposing a method used for supporting taking of decisions in a staff selection process.*

**Key words:** *Heuristic algorithms, Hill Climbing, Personnel's selection.*

## **Introducción**

Realizar una adecuada gestión de los recursos humanos, es requisito indispensable para implementar y mantener en la actualidad un sistema de calidad en las empresas, mejorar continuamente su eficacia e incrementar la satisfacción de los clientes. Para ello es necesario garantizar que el personal (que realiza trabajos que afectan la calidad de los productos que elaboran o servicios que brindan) sea competente, con la educación, formación, habilidades y experiencias apropiadas.

Como se puede ver, el trabajador ya no es visto como un ente inanimado o como una parte más del inventario de recursos disponibles de una empresa; su rol dentro de los objetivos organizacionales ha cambiado. Hoy por hoy, se habla del componente humano como la principal ventaja competitiva del 3<sup>er</sup> milenio, capaz de crear y difundir uno de los elementos más importantes y distintivos –“el conocimiento”– ; por lo que su gestión tiene que ser asumida en concordancia con los nuevos parámetros de dirección empresarial [Ballivián, 2006].

## **Desarrollo**

### **Selección de personal**

Ya sea la salida de un trabajador, por despido, jubilación, liberación, renuncia, etc.; la creación de un puesto, por causas funcionales, estructurales o tecnológicas; las vacantes generadas por un desarrollo profesional del personal y las causas ajenas a la organización: bajas, accidentes, enfermedades, etc.; pueden originar un proceso de selección [Arias, 2006].

Como bien citaran [Fernández & Baeza, 2003], el objetivo de la selección es encontrar el candidato que se ajuste lo más *rigurosamente* posible al perfil requerido para determinado cargo, o conjunto de cargos similares si se sigue la tendencia actual del proceso.

El proceso de selección debe transitar por al menos cinco fases que pudieran variar en función de cada empresa, la cual debe determinar los procedimientos adecuados que le brinden mejores resultados. Dichas fases son:

1. Viabilidad y planificación del proceso.
2. Reclutamiento.
3. Preselección.
4. Selección.
5. Acogida y Seguimiento

Específicamente en la 4ta fase está incluido el análisis y la toma de decisiones, lo cual implica la asignación de n personas a m puestos teniendo en cuenta los resultados obtenidos (en las evaluaciones) por dichas personas a lo largo del proceso. Esta asignación se puede realizar a partir de tres perspectivas:

1. Buscar la persona ideal para un puesto dado: es decir dado un puesto seleccionar entre los candidatos cuál sería el idóneo.
2. Buscar el puesto ideal para una persona dada: el proceso sería algo similar al anterior lo que se partiría de las características del candidato y los requerimientos de los puestos por los cuales el optó, en caso de que fuera más de uno, y se verifica en cual de ellos es más apropiado.
3. Obtener un cubrimiento óptimo de personas con puestos: en este caso se trata de encontrar una solución que de a la empresa una visión global de cuál debe ser la ubicación más acertada de los candidatos en determinados puestos.

En general las investigaciones en esta área se enfocan en la primera perspectiva. No se ha encontrado ningún trabajo que desarrolle un modelo formal de asignación de personas a puestos a partir de las competencias identificadas que desarrolle e implemente estas tres perspectivas.

En una decisión tan crítica como esta, lo ideal sería encontrar la *mejor* asignación de personas a puestos. Para ello es necesario encontrar una composición que logre *buenos* valores de criterios, tales como la suma de las idoneidades de cada persona con el puesto a ocupar (considerando las competencias definidas en el mismo como las más adecuadas).

La cantidad de combinaciones de asignaciones posibles, en dimensiones relativamente significativas de puestos a cubrir y empleados disponibles (situación típica en empresas medianas o grandes), puede convertirse en un problema con un espacio de soluciones demasiado grande. Esto hace que esta etapa sea prácticamente imposible de abordar sin la ayuda de sistemas automáticos de soporte a la decisión o se corre el riesgo de no hacer la asignación más adecuada de acuerdo a criterios preestablecidos. Este problema de asignación de  $n$  personas a  $m$  puestos es complejo (teniendo en cuenta que el espacio de búsqueda puede ser muy grande) y necesita optimización. Para conseguir este objetivo es necesario realizar un análisis de los métodos de optimización y búsqueda que mejor se adecuen a las características de este problema.

### **Optimización**

Optimización en el contexto científico es el proceso de tratar de encontrar la mejor solución posible para un determinado problema. Los problemas de optimización tratan de encontrar el valor de unas variables de decisión para los que una determinada función objetivo alcanza su valor máximo o mínimo [Marti a, 2003].

Algunas clases de problemas de optimización son relativamente fáciles de resolver, como es el caso de los problemas lineales (tanto la función objetivo como las restricciones son expresiones lineales). Otra variante de optimización es la discreta ó combinatoria. En ella las variables de decisión son discretas y la función objetivo es una expresión compleja: sumatorias, productorias, etc. Se aplica a problemas difíciles de resolver en los cuales el número de elementos del espacio de soluciones es muy elevado y no se puede garantizar el encontrar la mejor solución posible en un tiempo razonable. Estos problemas necesitan ser resueltos de forma eficiente teniendo en cuenta que se hace impracticable la evaluación de todas sus soluciones para determinar el óptimo [Marti a, 2003], [Ochoa, 2004], [Formella, 2006].

Ante la ausencia de un procedimiento exacto que permita obtener en todas las circunstancias una solución óptima en un tiempo razonable, es importante contar con un algoritmo heurístico que aporte una alternativa, con una alta confianza de que es la mejor solución posible o está muy próxima a serlo [Marti a, 2003]. En estos métodos la rapidez del proceso es tan importante cómo la calidad de la solución obtenida [Moreno & otros, 2005].

### **Metaheurísticas**

Las meta heurísticas son el resultado de aplicar la estrategia general de la Inteligencia Artificial en el área de las heurísticas. Históricamente su paradigma central son las denominadas metaheurísticas de búsqueda. Básicamente se trata de estrategias para recorrer el espacio de soluciones del problema transformando de forma iterativa soluciones de partida [Moreno & otros, 2005].

Los procedimientos de búsqueda o mejora local basan su estrategia en el estudio de soluciones del vecindario o entorno de la solución que realiza el recorrido. Comienzan con una solución del problema y la mejoran progresivamente. El procedimiento realiza en cada paso un movimiento de una solución a otra con mejor valor. El método finaliza cuando, para una solución, no existe ninguna solución accesible que la mejore [Marti a, 2003].

Los algoritmos metaheurísticos generalmente se clasifican en dos grupos principales [Paredes & Fajardo, 2008] [Rosete, 2000]: basados en un punto y basados en poblaciones de puntos. Entre los basados en un punto se encuentran: Escaladores de Colinas,

Búsqueda Tabú, Recocido Simulado, entre otros. Algunos de los basados en una población son: Algoritmos Genéticos, Algoritmo de Estimación de Distribución (o *Estimation of Distribution Algorithms* (EDA) por su nombre en inglés), Optimización Basada en Colonias de Hormigas (OCH), entre otros. En este trabajo solamente serán abordados los algoritmos metaheurísticos basados en un punto.

### **Escalador de colinas (HILL CLIMBING)**

El Escalador de Colinas (EC) se usa frecuentemente cuando se tiene una buena función heurística<sup>1</sup> para evaluar los estados y no se dispone de otro conocimiento útil. La idea es desplazarse en el espacio en la dirección de mejoramiento del valor del costo con la intención de llegar a un estado de costo óptimo. Para ello utiliza la información brindada por las soluciones anteriores que han sido visitadas [Russel & Norvig, 1996].

Una mejor idea para entender este método es describir la situación de la vida real a la cual se asemeja. Un escalador de colinas, el cual padece de amnesia y se encuentra cerca del Monte Everest, en medio de una espesa neblina que sólo le permite ver a unos pocos metros de su ubicación actual y su objetivo es alcanzar la cima. La estrategia que sigue el escalador para alcanzar su meta es irse moviendo hacia los lugares que desde su posición actual le parecen que indican un ascenso hacia la cima [Paredes & Fajardo, 2008] [Rosete, 2000] [González García, 2006].

El EC es un algoritmo de mejoramiento iterativo. Comienza a partir de un punto (actual) en el espacio de búsqueda. En cada iteración, un nuevo punto es seleccionado de la vecindad del punto actual. Si el nuevo punto es mejor, se transforma en un punto actual, sino otro punto vecino es seleccionado y evaluado. El método termina cuando no hay mejorías, o cuando se alcanza un número predefinido de iteraciones [Russel & Norvig, 1996].

El EC, en dependencia de la forma en que toma la solución candidata en cada iteración puede verse desde los siguientes enfoques [Rosete, 2000] [González García, 2006].

- ✓ Escalador de colinas clásico
- ✓ Escalador de colinas estocástico con mejor ascenso (ECE-MA).
- ✓ Escalador de colinas estocástico con primer ascenso (ECE-PA).

### **Escalador de colinas clásico.**

Éste es el método original y es un método clásico de búsqueda en Inteligencia Artificial. Se toma el mejor punto de la vecindad, revisándola de manera *exhaustiva*. Si ocurre que el valor de la evaluación de todas las soluciones candidatas son menores que el valor de la actual, entonces la solución actual es un máximo local. El escalador de colinas nunca saldrá de este punto y por tanto la búsqueda se detiene [Coz, 2008] [Rich & Knight, 1994] [Rosete, 2000].

### **Escalador de colinas estocástico con mejor ascenso.**

Es una adaptación del original al contexto de optimización y se relaciona con problemas donde *la vecindad a analizar es muy grande*, haciendo muy costoso el proceso de análisis exhaustivo del método clásico. En este método sólo se analiza una parte *aleatoria* de la vecindad la cual es escogida después que se aplican un subconjunto aleatorio de operadores posibles a esta. Luego se escoge el mejor de los puntos revisados como próximo estado. El hecho de que la selección de un nuevo estado no mejore el valor de la solución actual no implica que esta solución sea un máximo local ya que no se analizó toda la vecindad. El criterio de

---

<sup>1</sup> es un recurso que permite "enfocar" el proceso de búsqueda hacia una rápida solución; o sea, dirige la búsqueda evitando el examen de nodos poco prometedores.

parada de este algoritmo es cuando se hagan varias iteraciones sin variar la solución actual [Rosete, 2000] [Marti, 2003] [González García, 2006] [Coz, 2008].

### **Escalador de colinas estocástico con primer ascenso.**

A diferencia del anterior, este algoritmo sólo tiene en cuenta un punto de la vecindad, mejorando el costo de cada iteración. La selección del operador a aplicar (y por tanto de la solución explorada) también se hace de modo aleatorio. Su nombre se debe a que abandona la solución actual cuando encuentra la primera solución candidata que sea mejor que esta. Los EC mejoran su comportamiento si también se aceptan en cada iteración las soluciones candidatas con valor de evaluación igual al de la solución actual, ya que esto permitiría recorrer las mesetas [Rosete, 2000] [Marti, 2003]

Los EC son métodos de propósito general muy sencillos. Su limitación esencial es la propiedad de converger al máximo local más cercano, sin garantizar de ningún modo que este sea el óptimo global [Russel, 1996]. Sin embargo, muchas veces esta limitación se exagera [Rosete, 2000].

Las metaheurísticas proponen tres formas de escapar de los óptimos locales:

- 1) volver a iniciar la búsqueda desde otra solución.
- 2) modificar la estructura de entornos que se aplica.
- 3) permitir movimientos que no sean de mejora.

Es por ello que se han desarrollado varios métodos de optimización que tratan de resolver este problema (convergencia al óptimo local más cercano) de diferentes maneras, como el Escalador de colinas con reiniciación, el Recocido Simulado, la búsqueda Tabú, entre otros.

### **Escalador de colinas con reiniciación.**

En esencia este método funciona como una repetición de cualquiera de los enfoques anteriores en los casos que se detiene la búsqueda por los motivos siguientes:

- ✓ cuando se ha alcanzado un punto de máximo local.
- ✓ cuando se han realizado varias iteraciones sin mejorar la solución actual.

La repetición viene dada por la iniciación del algoritmo desde un punto distinto seleccionado aleatoriamente [Rosete, 2000] [Marti, 2003] [González García, 2006]. De las escaladas realizadas se guarda el mejor resultado. Pueden usarse un número fijo de iteraciones o puede continuar hasta que no se hayan utilizado al menos  $n$  estados iniciales [De la Fuente & otros, 2006].

### **Recocido o enfriamiento simulado (SIMULATED ANNEALING).**

Kirpatrick, Gelatt y Vecchi proponen en 1983 un procedimiento para obtener soluciones aproximadas a problemas de optimización, llamado Simulated Annealing. Esta nueva herramienta fue definida para ser empleada específicamente en la solución de problemas combinatoriales de gran complejidad. Surgió del campo de la termodinámica como consecuencia de la comparación de los problemas formulados en este campo con los del campo de la investigación de operaciones. Es una metodología simple y de gran potencialidad para ser aplicada a una gran variedad de problemas [Hincapié & otros, 2004].

Es similar al ECE-PA, solamente variando la regla de aceptación de la solución candidata. En lugar de empezar otra vez al azar y quedar atrapado una y otra vez en máximos o mínimos locales, puede ser conveniente descender unos cuantos pasos y así poder escapar de los fenómenos locales. Por tanto, no sólo se aceptará la nueva solución si su valor de evaluación es mejor, también podrán aceptarse soluciones peores. Esta ley de aceptación permitirá que se puedan aceptar soluciones peores en los estados *iniciales e intermedios* de la optimización, lo cual garantiza que el Recocido Simulado no se quede estancado en un extremo local. Si la búsqueda realmente está yendo hacia una buena solución, estos movimientos deben realizarse de un modo controlado. En los

estados finales, el método se comporta prácticamente como un EC [Rosete, 2000] [De la Fuente & otros, 2006] [Konar, 2000] [Morgan, 2006].

Actualmente se han propuesto variantes del Recocido Simulado, una de las más conocidas es Aceptación por Umbral (o *Threshold Accepting* por su nombre en inglés). Consiste en aceptar un movimiento siempre y cuando rebese un cierto umbral. Se conoce que se pierde mucho más rápido las propiedades de convergencia hacia un mínimo global. Una alternativa a esto es cambiar el valor del umbral al hacer la búsqueda. En caso de que se dejen de aceptar soluciones en el rango de la vecindad, entonces se aumenta el umbral buscando escapar de los mínimos globales y en caso contrario se reduce [Paredes & Fajardo, 2008].

En numerosos problemas de optimización ha sido probado con éxito, mostrando gran habilidad para evitar quedar atrapado en óptimos locales [Marti a, 2003]. Esto lo logra al aceptar movimientos de no mejora en los estados intermedios. La probabilidad de aceptar o rechazar un movimiento de no mejora depende no de la iteración (tiempo transcurrido) sino de lo sucedido en la búsqueda.

### **Búsqueda TABÚ**

Los orígenes de la Búsqueda Tabú (Tabú Search) pueden situarse en diversos trabajos publicados a finales de los 70. Oficialmente, el nombre y la metodología fueron introducidos posteriormente por Fred Glover en 1989. La Búsqueda Tabú es una técnica para resolver problemas combinatorios de gran dificultad que está basada en principios generales de Inteligencia Artificial (IA) [Marti a, 2003].

Su objetivo es dirigir y orientar la búsqueda teniendo en cuenta la historia de esta y para ello utiliza una lista denominada Tabú. Para evitar que el proceso regrese a los óptimos locales y entre en un ciclo repetitivo se almacenan en dicha lista las últimas operaciones realizadas de los estados anteriores, para excluir el uso de estos operadores en el estado actual y garantizar que no se retroceda a los estados visitados [Marti a, 2003] [Rosete, 2000] [Marti, 2003].

Se puede decir que hay un cierto aprendizaje y que la búsqueda es inteligente. La cantidad de operaciones posibles a almacenar en la lista tabú así como la forma de actualizarla repercuten de forma vital en el proceso de búsqueda. Si la lista es muy pequeña se producen ciclos, y si es muy grande se vuelve muy restrictiva. Una manera eficaz para evitar esta dificultad es utilizar una lista del tabú con tamaño variable donde cada elemento de la lista pertenece a ella para un número de iteraciones (limitado por valores máximos y mínimos dados). También puede considerarse que tras un cierto número de iteraciones la búsqueda está en una región distinta y puede liberarse del estado tabú a las soluciones antiguas. Aunque el método descarta los ya visitados puede avanzar hacia soluciones que no mejoren la solución actual [Glover & Melián, 2003].

Todos los algoritmos metaheurísticos antes explicados están basados en un punto, ya que mantienen una sola solución. Se basan en el EC modificando: condición de parada, la regla de aceptación de la solución candidata o la definición de la vecindad [Rosete, 2006]. Como todos son algoritmos de optimización de propósito general tienen un objetivo común: llegar a un valor cercano al óptimo. Ahora, ¿cuál es el mejor?

Según el Teorema NFL “*No Free Lunch Theorem*” ningún algoritmo es mejor que otro en la totalidad de los problemas en los que son aplicables. Es decir, si un algoritmo A es mejor que uno B sobre un grupo determinado de problemas, entonces debe esperarse que existan un conjunto igual de problemas donde ocurra lo contrario [Wolpert & Macready, 1997]. Lo que hace que un algoritmo se comporte mejor o no con respecto a un problema determinado, es la correcta selección de la función objetivo, así como los operadores que permiten variar de un estado a otro del problema.

En el Instituto Superior Politécnico José Antonio Echevarría (CUJAE), específicamente en la Facultad de Ingeniería Informática, en junio del 2008 fue desarrollada por [Paredes & Fajardo, 2008] una biblioteca de clases que implementa un modelo de

unificación de los algoritmos metaheurísticos basados en un punto, específicamente: Escalador de Colinas, Búsqueda Aleatoria, Recocido Simulado y Búsqueda Tabú. La misma combina las características similares entre los algoritmos antes mencionados, partiendo de que ninguno es mejor que otro en la totalidad de los problemas. Fue implementada sobre una plataforma libre y constituye un excelente material de consulta.

### **Algoritmos híbridos**

Es importante considerar que los métodos basados en búsqueda local requieren de la exploración de un gran número de soluciones en poco tiempo, por ello es crítico el reducir al mínimo el esfuerzo computacional de las operaciones que se realizan a menudo. Los diferentes métodos heurísticos tienen sus ventajas y desventajas. Por esta razón es frecuente encontrar la utilización de híbridos de varios métodos en la solución de un problema concreto, buscando complementar las fortalezas que puedan tener los diferentes métodos. Para ello es necesario conocer lo que es común en todos los procedimientos y en qué radican sus diferencias.

Los algoritmos metaheurísticos basados en un punto, tienen específicamente características comunes; lo que permite combinarlas. Por ejemplo, siempre tiene que ser especificado el o los *criterios de parada* para terminar la búsqueda. El más común es una cantidad máxima de iteraciones, la cual siempre debe existir, independientemente de que se especifique o no, para no caer en un ciclo infinito [Paredes & Fajardo, 2008]. Pero existen otras condiciones, tales como:

- ✓ Alcanzar una solución que sea mejor que un cierto valor fijado al inicio.
- ✓ No obtener una nueva mejor solución luego de una cierta cantidad de iteraciones.
- ✓ Si se está utilizando la Búsqueda tabú, que todos los vecinos del paso actual están incluidos en la lista tabú.

Con respecto a la cantidad de iteraciones, no es recomendable utilizar una cantidad fija predefinida. Esta puede ser estimada a partir del tamaño del espacio de búsqueda. De no ser así, se corre el riesgo de que sea muy pequeña y la solución finalmente encontrada no esté ni cerca del óptimo; ó que sea muy grande y la búsqueda termine siendo demasiado exhaustiva. Los criterios de parada pueden ser varios, a partir de la combinación de varias especificaciones.

Luego de definida la condición de parada, es necesario construir el *estado inicial*. Este puede ser definido de forma aleatoria o bajo algún criterio específico del problema a resolver. A partir de la solución inicial y luego de aplicar ciertos operadores, se generará una vecindad (que también puede ser de forma aleatoria). Dentro de esta vecindad, bajo diferentes criterios que dependerán del objetivo de la optimización (maximizar, minimizar), se debe seleccionar una solución candidata. Esta nueva solución puede ser *mejor o peor* que la actual, en dependencia del algoritmo que se este utilizando. Por ejemplo: El Escalador de Colinas Estocástico con Primer Ascenso y el Recocido Simulado seleccionan de la vecindad cualquier valor. También existe la variante de aceptar soluciones candidatas con valor de evaluación *igual* al de la solución actual, ya que esto permitiría recorrer las mesetas. Por tanto, no siempre se pasa a un estado mejor de la solución. Además está la variante que consiste en aceptar un movimiento siempre y cuando rebase un cierto umbral, lo cual puede implicar que en varias iteraciones no se modifique la solución actual.

Este proceso de aceptación de cambio de la solución actual, se repite mientras no se cumplan las condiciones de parada. En caso contrario, el algoritmo se detiene quedando la solución actual como el resultado final. A excepción del Recocido Simulado y la Búsqueda Tabú donde siempre se devuelve el mejor estado explorado, que no tiene por qué coincidir con el último [Paredes & Fajardo, 2008].

### **Propuesta de algoritmo para la asignación de personal**

El algoritmo heurístico a utilizar sería una variante de un Escalador de Colinas (mejoramiento iterativo) que combine varios de sus métodos. Se toma del Escalador de colinas estocástico con primer ascenso la característica de que sólo tiene en cuenta un punto de

la vecindad y abandona la solución actual cuando encuentra la primera solución candidata que sea mejor que esta. Con el objetivo de no retroceder a estados ya visitados se puede llevar una Lista tabú con las últimas operaciones realizadas. Si la búsqueda se detiene (condiciones de parada) con una solución poco factible se puede aplicar el Escalador de colinas con reiniciación, haciendo que el algoritmo se inicie nuevamente desde un punto distinto seleccionado aleatoriamente, para obtener nuevas soluciones.

Este algoritmo empezaría con una solución inicial que sería una distribución aleatoria de personas por puesto y luego busca en su vecindad (igualmente de forma aleatoria) una mejor solución, es decir, una distribución que maximice la función objetivo (coeficiente de adecuación de la persona al puesto de trabajo). Si la encuentra, reemplaza su solución actual por la nueva y continúa buscando nuevas soluciones en su vecindad, hasta que varias iteraciones completas consecutivas no produzcan un cambio en el estado actual (todos los vecinos del paso actual están incluidos en la lista tabú.). Al terminar este proceso, se debe haber encontrado una solución, que si no es la mejor, debe ser lo más acertada posible. De no serlo (a partir de un umbral definido), se puede realizar un reinicio aleatorio.

Para generar la distribución inicial, se utiliza una matriz de puestos y candidatos, donde cada casilla toma valor 1 ó 0 en dependencia de si está asignado o no ese candidato a ese puesto, garantizando que un candidato sea asignado a un solo puesto y viceversa. En caso de que la cantidad de puestos no sea igual a la de candidatos, se completan los que faltan con posiciones ficticias y el coeficiente de adecuación utilizado en el cálculo de la función objetivo para esa persona en ese puesto toma valor 0.

Si al finalizar la corrida, la solución incluye algunos de los siguientes casos:

- ✓ a un determinado perfil le corresponde un candidato ficticio o viceversa
- ✓ el número de ajuste persona – puesto es cero

dicha asignación será desechada debido a que no aporta nada a la solución.

## CONCLUSIONES

Las técnicas heurísticas permiten encontrar soluciones de buena calidad para problemas de gran tamaño que no pueden ser resueltos por metodologías exactas y cuya solución por un método exhaustivo es imposible. Existe una infinidad de problemas teóricos y prácticos que involucran a la optimización, tanto en la industria como en la ciencia, en los cuales estos algoritmos pueden ser aplicados. En este artículo se propone sea aplicado un algoritmo híbrido (que resalta las mejores fortalezas) a la asignación de personas a puestos, implícita en la toma de decisiones de cualquier proceso de selección de personal.

## Referencias Bibliográficas

1. [Arias, 2006] Arias Bravo, Yanet. “Modelo de selección de personal por competencias utilizando lógica difusa”. Tesis de maestría en Informática Aplicada. CUJAE. Diciembre, 2006.
2. [Ballivián, 2006] Ballivián, Rafael. “Desarrollo de un manual de selección de personal con un enfoque de procesos para la empresa de Correos de Cuba”. Tesis de Maestría en opción al título de Master en Gestión de Recursos Humanos. Instituto Superior Politécnico “José Antonio Echeverría”. Ciudad Habana. Cuba. Marzo 2006.
3. [Coz, 2008] Coz Velasco, Juan. “Sistemas Inteligentes. Búsqueda Heurística” Escuela Politécnica Superior de Ingeniería de Gijón. Universidad de Oviedo. 2008.
4. [De la Fuente & otros, 2006] Aparicio de la Fuente, Amador; Gonel Crespo, Alfredo Javier. “Estudio de los algoritmos de mejora iterativa” 2006. Disponible en: [www.infor.uva.es/~calonso/IAI/TrabajoAlumnos/iterativos.pdf](http://www.infor.uva.es/~calonso/IAI/TrabajoAlumnos/iterativos.pdf)

5. [Fernández & Baeza, 2003] Fernández Ignacio. & Baeza Ricardo. “Aplicación de modelo de competencias: experiencias en algunas empresas Chilena” [www.gestiopolis.com/recursos/documentos/fulldocs/rrhh1/modcompechile.htm](http://www.gestiopolis.com/recursos/documentos/fulldocs/rrhh1/modcompechile.htm)
6. Chile, 2003.
7. [Formella, 2006] Formella, D.A. “Técnicas avanzadas de optimización”. Universidad de Vigo, Departamento de Informática Área de Lenguajes y Sistemas Informáticos. 2006.
8. [González García, 2006] González García, Ariel. “Planificación inteligente del mantenimiento de equipos médicos”. Tesis de maestría en Informática Aplicada. CUJAE. Diciembre, 2006.
9. [Glover & Melián, 2003] F. Glover, B. Melián “Búsqueda Tabú”. Revista Iberoamericana de Inteligencia Artificial 19 29-48. Año 2003.
10. [Hincapié & otros, 2004] Hincapié Isaza, Ricardo Alberto. Ríos Porras, Carlos Alberto; Gallego, Ramón Alfonso. “Técnicas heurísticas aplicadas al problema del cartero viajante” Grupo de Investigación en Planeamiento de Sistemas Eléctricos. Universidad Tecnológica de Pereira. Mayo 2004. Disponible en: [www.utp.edu.co/~planeamiento/prod\\_aca/articulos/Tecnicas\\_heuristicas\\_TSP4.pdf](http://www.utp.edu.co/~planeamiento/prod_aca/articulos/Tecnicas_heuristicas_TSP4.pdf)
11. [Konar, 2000] Konar, A., “Artificial Intelligence and Soft Computing, in Behavioral and Cognitive Modeling of the Human Brain”, D.o.E.a.T.-c.E.J. University, Editor. 2000, CRC Press: Calcutta, India. p. 788.
12. [Martí, 2003] Martí Cunquero, Rafael. “Procedimientos Metaheurísticos en Optimización Combinatoria” <http://www.uv.es/~rmarti/paper/docs/heur1.pdf> Departamento de Estadísticas e Investigación de Operaciones. Facultad de Matemática. Universidad de Valencia. 2003.
13. [Martí a, 2003] Martí Cunquero, Rafael. “Algoritmos Heurísticos en Optimización Combinatoria”. Revista Iberoamericana de Inteligencia Artificial, Pág. 123-130, 2003.
14. [Moreno & otros, 2005] Moreno Pérez, José Andrés; Melián Batista, Belén. “Metaheurísticas para la planificación logística” Grupo de Computación Inteligente. Universidad de La Laguna. Febrero de 2005.
15. [Morgan, 2006] Morgan, W. “Simulated Annealing & other Combinatorial Approximation Algorithms” Concurrent Systems Engineering, 2006.
16. [Ochoa, 2004] Ochoa, Gabriela. “Optimización, Búsqueda” [www ldc.usb.ve/~gabro/teaching/EC](http://www ldc.usb.ve/~gabro/teaching/EC) 15 de septiembre de 2004.
17. [Paredes & Fajardo, 2008] Paredes, David; Fajardo, Jenny. “Biblioteca de clases para la unificación de algoritmos meta heurísticos basados en un punto” Trabajo de diploma para optar por el título de Ingeniería en Informática. CUJAE. Junio, 2008.
18. [Rich & Knight, 1994] Rich, Elaine; Knight, Kevin. “Inteligencia artificial”, 2a. ed., McGraw-Hill, España, 1994.
19. [Russel & Norvig, 1996] Russel, S; Norvig, P. “Inteligencia Artificial. Un enfoque moderno” Prentice Hall, 1996.
20. [Rosete, 2000] Rosete Suárez, Alejandro. “Una solución flexible y eficiente para e; trazado de grafos basada en el escalador de colinas estocástico” Tesis presentada en opción al grado científico de Doctor en Ciencias Técnicas, CUJAE, La Habana, 2000.
21. [Wolpert & Macready, 1997] Wolpert, David; Macready, William. “No Free Lunch Theorems for Optimization” IEEE Transactions on Evolutionary Computation, 1997.