

# DOCUMENTO DE DISCUSIÓN

**DD/14/04**

**Optimización de un problema de asignación  
generalizada a partir de un algoritmo de colonia de  
hormigas que incorpora un mecanismo de difusión**

*Luciano Stucchi*

[stucchi\\_l@up.edu.pe](mailto:stucchi_l@up.edu.pe)

Lima, marzo de 2014

## II.b. Asignación de expedientes a analistas:

Los problemas de asignación generalizada tienen muchas aplicaciones. Entre ellas está la asignación de celulares a torres de control (Shyong, Lin, & Hsiao, 2006), la asignación de puertas de embarque a aviones (Pintea, Pop, Chira, & Dumitrescu, 2008), de usuarios (antenas) a satélites (Kim, Kim, Mani, & Kim, 2007), entre otros. En todos estos casos se trata de procesos que involucran agentes no-humanos como encargados de realizar las actividades. Este sesgo proviene de dos factores: el primero, que es mucho más sencillo monitorear maquinarias y equipos que agentes humanos; el segundo, que los agentes humanos son mucho menos regulares para realizar una actividad que las máquinas.

En los negocios es muy común, sin embargo, encontrar actividades de una misma naturaleza y repetitivas que deben ser realizadas por un conjunto de especialistas. Cuando los volúmenes son pequeños, dicha asignación no resulta difícil de hacerse, pero cuando las magnitudes escalan a órdenes medianos se convierten en un problema difícil de gestionar. En las universidades, por ejemplo, se encuentran situaciones de este tipo: asignación de tutorados a tutores, distribución de clases a profesores, asignación de aulas a clases, asignación de expedientes para ser estudiados por analistas, entre otros.

En este artículo se estudia el caso de una universidad que tiene una oficina de pensiones, encargada de asignar a cada nuevo estudiante de pregrado en una escala, de acuerdo a los recursos económicos que tiene su familia. Esta distribución se realiza luego de que un analista especializado realiza un estudio sobre toda la información relevante que brinda la familia, relacionada esta con los ingresos, recursos, necesidad y extras de la misma. La oficina encargada será el área con la función de recopilar esta información, elaborar los expedientes, asignarlos entre sus analistas y hacer la evaluación del reporte de cada uno de ellos, finalizando en la asignación de cada alumno a una escala específica.

Históricamente, la asignación ha sido hecha por la dirección de esta oficina de manera manual, aunque a veces distinguiendo entre expedientes difíciles, intermedios y fáciles, donde los primeros son asignados a los analistas de mayor antigüedad y experiencia, los fáciles a los analistas nuevos y los intermedios de manera uniforme entre todos. Esta asignación, al ser manual y en muchos casos aleatoria, no termina por ser equitativa habiendo en ocasiones una sobrecarga en algunos analistas y una mayor holgura en otros, lo que fuerza a hacer redistribuciones posteriores.

Así, este proceso administrativo puede modelarse como un GAP donde se tienen  $n$  expedientes, cada uno identificable con una dificultad  $r_{ij}$ , que estará establecida según la información que tenga. Cada expediente requiere de un tiempo  $t_{ij}$  de análisis. Este tiempo puede estimarse según la dificultad del expediente y el analista que lo tenga a cargo, de acuerdo al histórico recopilado en base a análisis anteriores. Se cuenta con  $m$  analistas encargados de los expedientes, cada uno con una cantidad máxima  $r_j$  de "dificultad" disponible. Este valor de "dificultad" será un parámetro establecido entre los analistas para clasificar el nivel de trabajo que requiere manipular un documento. Esto significa que se quiere hacer una repartición equitativa entre los analistas según la dificultad de los expedientes, de modo que ninguno termine saturado por estos. El objetivo del problema será entonces encontrar una solución que minimice el tiempo máximo de trabajo de todos los analistas, manteniendo una distribución pre-establecida de dificultad.

Este problema es un GAP puesto que se establece la siguiente equivalencia y se someten dichas variables a las ecuaciones descritas antes.

Variable	Conjunto	GAP	Asig. de expedientes a analistas
$n$	$B$	Tareas	Expedientes
$m$	$A$	Agentes	Analistas
$g$	$G$	Ganancia/Inversión	Tiempo
$r$	$R$	Recursos	Dificultad del expediente
$p$	$P$	Límite de recursos	Dificultad límite que un agente puede manejar

Tabla 1. Se muestra la equivalencia entre las variables que describen un GAP y las que describen el problema de asignación de expedientes a analistas. Se ve la correspondencia entre: Tareas  $\rightarrow$  Expedientes, Agentes  $\rightarrow$  Analistas, Ganancia/Inversión  $\rightarrow$  Tiempo y Recursos  $\rightarrow$  Dificultad del expediente. Se considera aquí que, dado que se busca minimizar el tiempo invertido por los analistas, el problema que se va a resolver es una minimización de  $G$ .

### II.c. Diffusive Ant Colony System:

Los problemas NP-hard son especialmente atractivos para ser resueltos por métodos metaheurísticos (Michalewicz, Schmidt, Michalewicz, & Chiriach, 1007). El ACO (Dorigo & Stützle, 2004) es un método que replica el comportamiento de las hormigas, cuando intercambian información de su ambiente para hallar el camino más corto desde sus nidos hasta las zonas donde recogen comida. Este proceso es emergente pues se produce por la emisión de feromonas por parte de las hormigas y la posterior interacción entre hormigas y feromonas. Entre las muchas versiones que existen del ACO, aquí se recoge la conocida como ACS (Dorigo & Gambardella, 1997), por su versatilidad, simpleza y robustez.

Para resolver un problema mediante el ACS es necesario formular este como si fuera un problema de vendedor viajero o TSP (por sus siglas en inglés). El TSP es un problema clásico, en el que se tiene a un agente viajero que debe recorrer una serie de ciudades, representadas por nodos en una red, y se busca encontrar el recorrido más corto que conecte todos los nodos de la red. Asimismo, el ACS es capaz de resolver un TSP al encontrar la secuencia óptima de nodos minimizando la extensión total de esta. Un GAP puede plantearse como TSP siguiendo la formulación hecha por (Shyong *et. al.*, 2006). En este caso tanto los agentes como las tareas se consideran como nodos componentes de la red, pero como las soluciones están en función de la asignación de tareas a agentes, la única longitud que se toma en cuenta es la de los arcos que conectan los nodos tarea con los nodos agente y no los contrarios. Dado que estos arcos agente-tarea no ofrecen ninguna ventaja a la solución final del problema, la matriz de adyacencia toma la forma descrita en la Figura 1.

Las asignaciones de agente a tarea, como no reflejan ningún significado en el problema real, se completan con valores numéricos muy pequeños, de modo que no afecten el resultado final. No se les coloca el valor de 0 para no interferir en el cálculo de las probabilidades, las cuales se obtendrán luego y serán inversamente proporcionales al peso de las asignaciones. Una vez establecido el problema de esta forma, como un TSP, se procede a la implementación del ACS.

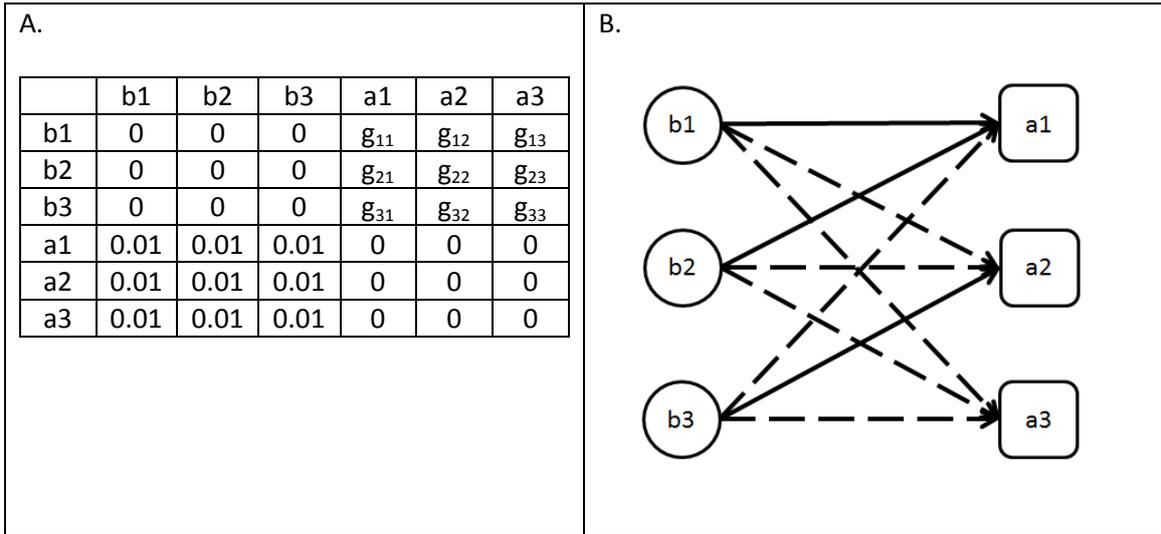


Figura 1. Representación de un GAP como un TSP. (A) Se muestra la matriz de adyacencia, donde se evidencia que los agentes solo pueden conectarse con actividades y viceversa. Eso hace que la matriz esté formada por dos submatrices. Los arcos  $\{a_j, b_i\}$  se escoge que tengan valores muy pequeños, de modo que resulten invisibles al problema. (B) Representación visual del problema. Las líneas sólidas representan una solución posible, mientras que las líneas punteadas representan el resto de posibles soluciones. Los arcos  $\{a_j, b_i\}$  no se dibujan, pues son irrelevantes para efectos de la solución.

El ACS consiste en disponer de  $k$  hormigas, las cuales se localizarán sobre un nodo distinto, asignado de manera aleatoria. En cada paso temporal las hormigas escogerán un siguiente nodo, hasta completar un recorrido que incluya todos los nodos  $b$ , sin importar que los de tipo  $a$  puedan estar repetidos. Esto porque un agente puede realizar más de una tarea, o ninguna, y las tareas tienen que asignarse necesaria y únicamente una vez. La elección de cuántas y cuáles tareas realiza un agente estará condicionada a los límites establecidos por  $P$ . Cada ruta elaborada por las hormigas recibirá un peso  $T$ , al completarse, que servirá para asignar feromonas a los arcos que han sido recorridos. Estas feromonas serán asignadas de acuerdo a las actualizaciones que se explican más adelante y dependen de la estructura de la red y la longitud de las rutas. Al siguiente turno, las hormigas volverán a elaborar una ruta, siendo ahora afectada cada elección de nodo por las feromonas depositadas en el paso anterior.

La probabilidad con la que cada hormiga escoge cada siguiente nodo proviene de una heurística que combina dos efectos: explotación de la mejor solución y exploración sobre soluciones que no son las mejores. Esta combinación se hace estocásticamente determinando cuál será el nodo  $j$  al que la hormiga se dirigirá desde el nodo  $i$ , escogiendo  $j$  como:

$$j = \begin{cases} \operatorname{argmax}_{l \in \Omega} \{\tau_{il}^\alpha \eta_{il}^\beta\}, & \text{cuando } q \leq q_0, \\ p_{ij} & \text{de otro modo,} \end{cases} \quad (\text{ec. 1})$$

y

$$p_{ij} = \begin{cases} \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{h \in \Omega} \tau_{ih}^\alpha \eta_{ih}^\beta}, & \text{con } j \in \Omega, \\ 0 & \text{de otro modo,} \end{cases} \quad (\text{ec. 2})$$

donde  $\Omega$  es el conjunto de todos los nodos que pueden visitarse estando en el nodo  $i$ . Los términos  $\tau_{ij}$  y  $\eta_{ij}$  representan respectivamente la cantidad de feromonas y la preferencia de cada arco, donde esta última característica es definida por una heurística que indica cuán atractivo es un arco solo en función de su longitud. Los parámetros  $\alpha$  y  $\beta$  sirven para afectar la importancia de cada factor en la determinación de la probabilidad y se ajustan empíricamente.

La asignación de feromonas a los arcos estará sujeta a tres procesos. El primer proceso es una asignación local e inmediata al paso de cada hormiga. Esta actualización se realiza apenas una hormiga recorre un arco  $ij$  y tiene la siguiente forma:

$$\tau_{ij} = (1 - \xi)\tau_{ij} + \xi\tau_0 \quad (\text{ec. 3})$$

lo cual sirve para favorecer la exploración de arcos distintos por diferentes hormigas, en un mismo paso. Aquí  $\xi$  es otro parámetro escogido empíricamente y  $\tau_0$  será establecido en función a los nodos que tienen mayores restricciones (Dorigo & Stützle, Ant Colony Optimization, 2004). Esta actualización local se balancea con una actualización global, hecha a partir de escoger la mejor ruta de entre todas las generadas por las hormigas en un paso y aplicar la siguiente actualización:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \rho \frac{1}{C} \quad (\text{ec. 4})$$

donde los arcos  $ij$  serán solo aquellos que pertenezcan a la mejor ruta encontrada,  $C$  será la longitud de dicha ruta y  $\rho$  representa un parámetro de evaporación, el cual también se ajusta empíricamente. La evaporación es una referencia al proceso que siguen las feromonas reales cuando son depositadas por las hormigas.

Es esta referencia a la evaporación lo que sugirió la incorporación de la difusión como otro mecanismo que podría favorecer la exploración de nuevas rutas alrededor de las existentes, haciendo nuevamente un paralelo con lo que ocurre con las feromonas reales. Aunque Zhu et al. (Zhu, 2010) han postulado antes un mecanismo de difusión, el propuesto en este artículo trabaja sustantivamente diferente, tal como se muestra a continuación. Este mecanismo ha sido incorporado introduciendo la siguiente regla de actualización:

$$\tau_{ij} = (1 - \mu)\tau_{ij} + \frac{\mu}{L} \sum_{h \in \Omega} \tau_{hi} \quad (\text{ec. 5})$$

en donde  $\Omega$  es el conjunto de todos los nodos que llegan al nodo  $i$  y  $L$  es la extensión de ese conjunto. El parámetro  $\mu$  representará la cantidad porcentual de feromona que se difunde y nuevamente será un valor empírico que deberá ajustarse según los resultados experimentales.

El mecanismo se implementa únicamente desde los arcos  $\{b_i, a_j\}$ , que son los arcos de asignación tarea-agente, y no en los arcos  $\{a_j, b_i\}$  que más bien son los arcos de asignación agente-tarea. Esto significa que aquellas feromonas que se difundirán serán las de los arcos tarea-agente sobre los arcos agente-tarea y no al revés. La idea es que se exploren aquellos arcos agente-tarea que derivan en los mejores arcos de asignación tarea-agente, pues la distribución de estos últimos es la que se busca optimizar. Esto se muestra en la Figura 2. Como la asignación agente-tarea no es relevante al resultado final, pues se trata solo de puntos de conexión para la red, la difusión de las feromonas se hará de esta manera selectiva y solo irá de la sección matricial superior-derecha hacia la inferior-izquierda.

Para determinar cuál es el valor empírico de  $\mu$  que mejor rendimiento alcanza, se compara la convergencia y la rapidez para obtener el óptimo, para diferentes valores de  $\mu$ . De los resultados obtenidos se escoge como parámetro a  $\mu = 0.1$ , el cual reflejaría una difusión que, aunque muy lenta, permite garantizar una mínima exploración alrededor de las mejores conexiones —arcos tarea-agente—, pero no lo suficientemente intensa como para deshacer la información generada por el recorrido de las hormigas y provocar una convergencia rápida sobre cualquier óptimo local.

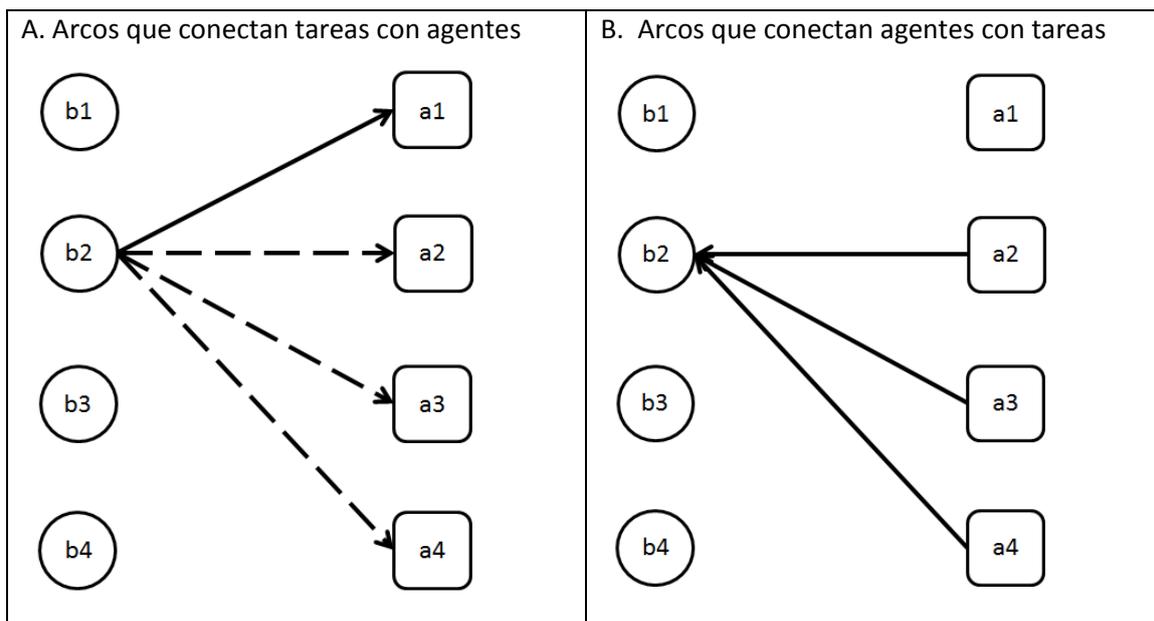


Figura 2. Exploración de arcos con difusión. (A) Se muestra con la línea sólida un arco tarea-agente que representa un arco con mayor cantidad de feromonas acumuladas que el resto de posibles conexiones, que se muestran con líneas punteadas. (B) Los arcos agente-tarea, como líneas sólidas, que serán afectados por la difusión del arco de la figura 2A. Esto significa que el sistema forzará a las hormigas a explorar los arcos agente-tarea que permitan la ocurrencia del mejor arco tarea-agente.

Este valor tan pequeño para el parámetro funciona en paralelo con lo que se observa en la evaporación, la cual también resulta óptima cuando su valor no es muy significativo, pues evita, por un lado, que la información de las hormigas se pierda por completo, pero permite, por otro lado, que los resultados poco atractivos sean olvidados con el tiempo y se refuercen únicamente las mejores rutas.

### III. Metodología:

Los expedientes pueden llegar hasta con 17 posibles documentos adjuntados. Estos recogen la información señalada de cada alumno. Como cada posible documento requiere de cierto estudio, los analistas han establecido una clasificación de acuerdo al tiempo y esfuerzo que cada uno les requiere en promedio. De este modo, los documentos se han clasificado en cuatro tipos: datos (4 documentos del total de 17), documentos fáciles (4 documentos de 17), documentos intermedios (5 documentos de 17) y documentos difíciles (4 documentos de 17), según se trate únicamente de anotar información o de documentos que requieran de algún tipo de análisis. Este análisis ha sido ponderado considerando que los documentos complejos equivalen al triple de trabajo que los sencillos, mientras los intermedios equivalen solamente al doble. Los documentos que solo son datos anotados no tienen un peso significativo, por lo cual no suman dificultad. Esto fija una escala de dificultades donde: los datos pesan 0, los documentos sencillos pesan 1, los documentos intermedios pesan 2 y los documentos complejos pesan 3. Cada expediente tendrá una dificultad entonces que será la suma aritmética de la dificultad individual de cada documento que lo compone. Esta escala fue validada consultando a los analistas por separado para ser lo más objetivos posible. Si bien las ocurrencias de expedientes con documentos de mucha complejidad se dan, son muy pocos los casos en que se obtenga el máximo de dificultad posible, de modo que la distribución se ha

normalizado en base al histórico total que se ha obtenido en todos los casos, donde la mayor complejidad encontrada en un expediente fue de 21.

A partir de esta dificultad por documento, los expedientes podrían dividirse en diferentes secciones, que serán más o menos significativas de acuerdo a cada analista. Esta división se ha dispuesto de la siguiente manera:

<b>Tipo de expediente</b>	<b>Rango de dificultad</b>
b1	1 – 5
b2	6 – 9
b3	10 – más

Tabla 2. Clasificación de los expedientes según el tipo de documentos que incluyen, considerando una ponderación 1 – 2 – 3 de cada uno de ellos.

Los analistas trabajan sus expedientes sobre una plantilla de una hoja de cálculo. Se incorporó un reloj en dicha plantilla, que tomaba el tiempo desde que el analista abría la hoja de cálculo de un expediente específico para ingresar algún dato hasta que la cerraba y el tiempo se iba sumando cada vez que se repetía el proceso. La toma del tiempo fue automática para no interferir con el trabajo de los analistas y evitar que estos se olviden de tomar el tiempo en algún momento. Si bien es posible que las eventualidades los distraigan en su trabajo, se considerará que esta introducción de "ruido blanco" es totalmente uniforme a todos los casos y por tanto no influirá realmente en los valores relativos de los datos.

Los datos utilizados en este trabajo recogen la información recabada durante tres meses, en tres grupos que corresponden a los expedientes recibidos y analizados durante agosto, setiembre y octubre del 2013. El primer grupo de datos (ago\_4x77), correspondiente a agosto, consta de 77 expedientes. El segundo (set\_4x111), correspondiente a setiembre, consta de 111 expedientes. El tercero y último, correspondiente a octubre (oct\_4x103), consta de 103 expedientes. Cada dato consigna: i) el tipo de expediente, de acuerdo a su dificultad; ii) el analista que lo tuvo a su cargo; iii) el tiempo en minutos que le tomó al analista evaluarlo.

Se trabajó primero con el juego de 77 datos, correspondiente a los expedientes analizados durante el mes de agosto. A partir de estos datos se puede encontrar la siguiente distribución de tiempos para cada analista. Esta distribución considera que todos los analistas encuentran distinta dificultad en manejar los tipos de expediente. Se ha tomado este primer conjunto de datos como referencia porque resulta bastante representativa del conjunto.

Los resultados obtenidos se han sacado para los juegos de datos set\_4x111 y oct\_4x103. Con estos datos se probará cuánto se puede mejorar la distribución de expedientes a partir del tiempo total de trabajo del equipo de analistas. La restricción que se usará es que todos trabajen con una asignación uniforme de dificultad. Para ello se sumará la dificultad individual de cada uno de los expedientes y se obtendrá la dificultad total del conjunto. Luego se dividirá este valor entre el número total de analistas, es decir, entre cuatro. Esto fijará la dificultad promedio del total. La restricción será entonces que la carga individual de dificultad no exceda en 10% esa dificultad promedio obtenida. En otras palabras, entre analistas nunca habrá una diferencia por encima del 20% entre la dificultad total que tengan que manejar. De esa forma se espera que la carga asignada sobre cada uno sea equitativa y uniforme.

El algoritmo usado para encontrar la distribución óptima fue una versión del ACS, al cual se le incorporó difusión. El algoritmo se elaboró en el software Mathematica, siguiendo la siguiente estructura:

#### Algoritmo DACS:

1. Se crea la matriz de adyacencia de la red de nodos a partir de la información del caso.
2. Se distribuyen las feromonas sobre los arcos a partir del inverso de la longitud de cada arco, según la recomendación de (Dorigo & Stützle, Ant Colony Optimization, 2004).
3. Se inicializa el contador de iteraciones  $i = 0$ .
4. Se construye la matriz de probabilidades a partir de (ec. 2).
5. Se sitúa a cada una de las  $k$  hormigas en un nodo, de forma aleatoria.
6. Cada hormiga escoge a partir de (ec. 1) el siguiente nodo de su ruta y le aplica al arco formado la actualización local de las feromonas (ec. 3).
7. Se repiten el paso 6 hasta verificar que todas las hormigas han completado una ruta.
8. Se calcula la distancia de todas las rutas, se escoge a la mejor y se almacena.
9. Se aplica la actualización global de las feromonas (ec. 4) con la mejor ruta.
10. Se aplica la difusión (ec. 5) sobre todos los arcos.
11. Se actualiza el contador de iteraciones a  $i = i + 1$ .
12. Se regresa al paso 4 y se repiten los siguientes pasos hasta que  $i = total$ .
13. Se escoge la mejor solución encontrada.

Para contrastar los resultados, la aplicación del DACS ha sido comparada con el ACS regular, con un algoritmo LP y con la distribución aleatoria con la que se realizó la toma de datos real. El ACS utilizado es idéntico al DACS sin el paso 10, que es el que introduce la difusión como mecanismo de exploración para otras rutas similares al óptimo local. Además, también se ve que es lo que ocurre cuando se trata de resolver este problema por medio de un algoritmo de programación lineal. El algoritmo LP utilizado es el que viene por defecto en el lenguaje de programación Mathematica, aplicado a programación entera.

#### IV. Resultados:

La aplicación del DACS permite hallar una distribución que reduce significativamente el tiempo de trabajo de los analistas, manteniendo una asignación uniforme de expedientes entre ellos. En los expedientes trabajados en el grupo de datos correspondiente a setiembre set\_4x111 se encontró un tiempo total de trabajo de 7978 minutos, tiempo que fue reducido por medio del DACS hasta a 5047 minutos, lográndose una mejora del 36.74%. A su vez, la dificultad promedio redujo su coeficiente de variación de 0.295 a 0.193 y la cantidad de expedientes redujo su coeficiente de variación de 0.305 a 0.296.

En los expedientes trabajados en el grupo de datos correspondiente a octubre oct\_4x103 se encontró un tiempo total de trabajo de 6962 minutos, tiempo que fue reducido por medio del DACS hasta a 4509 minutos, lográndose una mejora del 35.23%. A su vez, la dificultad promedio redujo su coeficiente de variación de 0.463 a 0.193 y la cantidad de expedientes redujo su coeficiente de variación de 0.404 a 0.356.

Por otro lado, los datos fueron probados con el algoritmo ACS, pero los resultados obtenidos no fueron tan buenos como los obtenidos por el DACS, lográndose un rendimiento de 36.68% y 34.40%, respectivamente. Los resultados fueron obtenidos en todo momento con poblaciones muestrales de 100 corridas, cada una de las cuales se mantuvo por 2500 pasos, para asegurar que exista convergencia. Los resultados mostrados vienen de haber comparado la última distribución de rutas de las hormigas.

Como punto final cabe mencionar que se buscó resolver estos dos casos también por medio de programación lineal, usando el algoritmo de programación entera built-in de Mathematica.

Para ello se corrió el algoritmo LP hasta por el doble de tiempo que le requirió al DACS y ACS en resolver el sistema. Si embargo, el algoritmo LP solo obtuvo resultados cuando los requerimientos de dificultad se establecieron por un encima de una proporción de 100% respecto a la cuarta parte de la dificultad total del conjunto. Es decir, el sistema solo pudo ser resuelto cuando las restricciones fueron liberadas en gran magnitud. En el caso de los datos de setiembre, la proporción mínima resuelta por el algoritmo de LP fue de 2 a 1. Esto significa que, teniendo una dificultad total de 259, la máxima dificultad permitida por agente era de 130, en lugar del 72 permitido por las metaheurísticas. Eso dio como resultado una distribución donde solo dos de los agentes recogían todas las tareas. En el caso de octubre, la proporción mínima resuelta fue de 2.6 a 1. Dado que la dificultad total en este juego de datos era de 242, los agentes tuvieron como límite ahora el valor de 158 y no de 67, como con el DACS y ACS. La distribución nuevamente dio solo dos agentes concentrando todas las tareas pero además teniendo uno de ellos casi el doble de tareas que el otro. Aunque los tiempos obtenidos en estos casos resultaron bastante menores a los obtenidos por medio del DACS o el ACS, las distribuciones obtenidas resultan tremendamente ineficientes por la manera como reparten la carga entre los analistas. Si bien no es preciso decir que el algoritmo LP no es capaz de encontrar mejores resultados, sí resulta pertinente aclarar que estos no han podido ser obtenidos bajo tiempos de ejecución equivalentes a aquellos destinados para ejecutar los algoritmos DACS y ACS en las mismas computadoras. Sin embargo, si los algoritmos se dejasen correr de manera indefinida, es posible que puedan encontrar una solución mejor. Pero no habría manera de saberlo de antemano.

## **V. Conclusiones:**

En este trabajo se presenta un mecanismo para mejorar el rendimiento del ACS, fundamentando en la forma cómo las feromonas, como compuesto químico, se difunden en el espacio, una vez que han sido depositadas por las hormigas. El DACS muestra mejoras significativas al momento de obtener la solución óptima, lo cual ha sido probado por medio de la evaluación de los problemas de Chu & Beasley (Beasley, 1997). Este mecanismo ha sido implementado discretamente considerando únicamente su aplicación sobre los arcos agente-tarea, a partir de la concentración de feromonas en los arcos tarea-agente, pues estos últimos son los que contienen la información que determina la solución óptima al problema de asignación. En ese sentido, se ha buscado potenciar la exploración de los arcos agente-tarea (que en sí mismos no aportan nada a la solución del problema) que se encuentran alrededor de los mejores arcos tarea-agente, de forma que converjan hacia estos. Se muestra que el mejor parámetro para implementar la difusión resulta ser  $\mu = 0.1$ , pues este ofrece la suficiente lentitud en convergencia y rapidez en la facilidad para encontrar el óptimo.

Como aplicación al DACS se ha procedido a evaluar cuán significativa resulta la optimización de un problema de asignación de expedientes en un proceso administrativo de una universidad. Este problema de asignación consiste en generar una distribución de expedientes asignados a analistas, los cuales deberán evaluarlos y resolverlos. Estos expedientes tienen un rango de dificultad, obtenido empíricamente. Los analistas, a su vez, tienen una diferente experiencia, lo cual les permite manejar los diferentes tipos de expediente con individual facilidad o dificultad. Históricamente estos problemas suelen resolverse asignando los expedientes de manera aleatoria a los analistas, distribuyéndolos uniformemente en cantidad, sin ningún otro criterio. Se han comparado los tiempos obtenidos en tres muestras mensuales de un total de 291 expedientes, de la evaluación hecha de manera aleatoria, con la evaluación que podría haberse hecho si la distribución fuese optimizada por un algoritmo metaheurístico.

Los resultados obtenidos reflejan claramente que este tipo de problemas pueden ser resueltos de manera mucho más eficiente por medio de la implementación de un algoritmo como el

DACS. Los tiempos manejados en total por los cuatro analistas del caso de estudio de este trabajo mejora en 36.74% y 35.23% en las simulaciones generadas con el DACS. La distribución de expedientes entre los analistas, midiendo la dificultad de los primeros, también se hace mucho más uniforme, reduciendo los coeficientes de variación de 0.295 a 0.193, en un caso, y de 0.463 a 0.193 en el otro. Estos resultados reflejan la importancia de implementar métodos de optimización, incluso en procesos manejados por humanos y no por maquinaria, que son los que concentran toda la atención de la investigación de operaciones. Así, cualquier proceso suficientemente estandarizado y donde se pueda contar con un mecanismo de recolección de datos, ofrece la posibilidad de una optimización basada en procesos y en algoritmos metaheurísticos.

Esta implementación ha considerado que cada analista es capaz de soportar la misma carga de dificultad que sus compañeros, sin embargo la metodología y los algoritmos permiten fácilmente la extensión a un caso donde la distribución sea heterogénea y se ajuste más particularmente a las necesidades de cada individuo.

Queda finalmente como corolario añadir que el DACS no hace sino incorporar un proceso de los sistemas biológicos, estrictamente un proceso químico, a una metodología creada y fundamentada sobre la misma base del funcionamiento de los organismos vivos. Eso da pie a seguir explorando qué otras analogías pueden seguir encontrándose entre la biología y la ingeniería, que nos permitan explotar y aprovechar los mecanismos que ya la misma naturaleza ha desarrollado.

## VI. Referencias:

- Beasley, P. C. (1997). A genetic algorithm for the generalised assignment problem. *Computers and Operations Research*, 24, 17-23.
- Cerný, V. (1985). A thermodynamical approach to the traveling salesman problem. *Journal of Optimization Theory and Applications*, 41-51.
- Dorigo, M., & Gambardella, L. (1997). Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation*, 53-66.
- Dorigo, M., & Stützle, T. (2004). Ant Colony Optimization. *The MIT Press*.
- Holland, J. (1975). Adaptation in Natural and Artificial Systems. *Ann Arbor, University of Michigan Press*.
- Kennedy, J., & Eberhart, R. (1995). Particle Swarm Optimization. *Proceedings of the IEEE International Conference on Neural Networks*, 4, 1942-1948.
- Kim, S., Kim, H., Mani, V., & Kim, C. (2007). Ant Colony Optimization for Satellite Customer Assignment. *ICUCT 2006 LNCS 4412*, 170-179.
- Kirkpatrick, S., Gelatt, C. J., & Vecchi, M. (1983). Optimization by simulated annealing. *Science*, 220, 671-680.
- Martello, S., & Toth, P. (1981). An algorithm for the generalized assignment problem. *Operations Research '81*, 589-603.
- Michalewicz, Z., Schmidt, M., Michalewicz, M., & Chiriach, C. (1007). *Adaptive Business Intelligence*. New York: Springer Berlin Heidelberg.
- Pintea, C., Pop, P., Chira, C., & Dumitrescu, D. (2008). A Hybrid Ant-Based System for Gate Assignment Problem. *HAI 2008 LNAI 5271*, 273-280.
- Quian, T., Zhang, Z., Gao, C., Wu, Y., & Liu, Y. (2013). An Ant Colony System Based on the Physarum Network. *ICSI 2013, Part 1, LNCS 7928*, 297-305.

- Randall, M. (2004). Heuristics for Ant Colony Optimization using the Generalized Assignment Problem. *Proceedings of the Congress of Evolutionary Computing, IEEE press*, 1916-1923.
- Ross, G., & Soland, P. (1975). A branch and bound algorithm for the generalized assignment problem. *Math. Prog.* 8, 91-103.
- Sahni, S., & Gonzalez, T. (1976). P-Complete Approximation Problems. *J. ACM*, 23, 555-565.
- Shyong, J., Lin, B., & Hsiao, T. (2006). Ant colony optimization for the cell assignment problem in PCS networks. *Computers & Operations Research*, 33(6), 1713-1740.
- Zhu, P. Z. (2010). A novel Ant Colony Optimization Algorithm in application of pheromone diffusion. (I. K. al., Ed.) *LSMS/ICSEE, Part II(LNCS 6329)*, 1-8.