

Optimización con Colonia de Hormigas para la Planificación Óptima de la Fuerza de Trabajo

Ant Colony Optimization for Optimal Workforce Planning

Juan C. Pisco-Vanegas, Abraham A. Bayas-Zamora, Rodolfo Najarro Quintero
Amilkar Y. Puris Cáceres, Pavel Novoa-Hernández

Resumen— En este trabajo, el problema de la planificación de la fuerza de trabajo en empresas es enfocado como un modelo de optimización matemática. A partir de la complejidad y la naturaleza del modelo de dicho problema, se propone como método de solución la metaheurística Optimización con Colonias de Hormigas y una variante de este enfoque. Los experimentos computacionales desarrollados indican que la propuesta es efectiva en comparación con otros algoritmos del estado del arte.

Palabras Clave— Planificación de la fuerza de trabajo, metaheurísticas, optimización con colonia de hormigas.

Abstract— In this paper the problem of obtaining the Optimal Workforce Planning for enterprises is addressed as a mathematical optimization problem. Given the complexity and specific features of the corresponding optimization model, the Ant Colony Optimization metaheuristic was applied as the solution method. From computational experiments, we have shown the effectiveness of our proposal versus other state-of-art algorithms.

Index Terms— Workforce planning, metaheuristics, ant colony optimization.

I. INTRODUCCIÓN

LA planificación de la fuerza de trabajo (*Workforce Planning*) es una de las actividades más importantes y a la vez más difíciles de concretar en una empresa [1]. Las principales causas que justifican dicha complejidad son: el tamaño de la empresa (expresado a grosso modo, en número de trabajadores y operaciones) y la incertidumbre. En esencia, este problema se puede ver como un proceso de decisión que involucra elementos de personal (*staffing*) y planificación

(*scheduling*).

Planificar la fuerza de trabajo implica algunas características especiales que están ausentes en los demás tipos de problemas de asignación de recursos debido a que el entorno de decisión tiende a ser muy dinámico y los planificadores de la fuerza de trabajo se enfrentan a un conjunto muy heterogéneo de empleados, donde se deben tener en cuenta no solo las preferencias de los diferentes empleados y las limitaciones sindicales, sino también las diferentes habilidades que los trabajadores pueden poseer [1].

Dada la naturaleza cuantitativa de este escenario de decisión y su presencia desde el origen de la empresa misma, ha sido abordado desde varios enfoques de la Investigación de Operaciones (IO). De acuerdo a [2], los trabajos existentes se pueden clasificar en cuatro grandes grupos: 1) modelos de cadenas de Markov, 2) modelos de simulación por computadora, 3) modelos de optimización, y 4) gestión de cadena de suministros mediante dinámica de sistemas.

Los principales estudios se han realizado a partir de los modelos de optimización, enfatizando en las técnicas de Investigación de Operaciones como se presenta en [3], donde se realiza una recopilación de las principales variantes del problema WPP y las técnicas de solución utilizadas.

Pocos trabajos se han encontrado que propongan modelos de optimización basados en técnicas metaheurísticas [4] para el problema WPP mono objetivo. Solo en el trabajo [5] se presenta dos versiones paralelas de un Algoritmo Genético (AG) y del Recocido Simulado (RS) para reducir el costo computacional de la búsqueda y ofrecer mejores soluciones.

Los trabajos más recientes utilizan un enfoque multiobjetivo del problema con una visión más práctica del entorno de aplicación. Por ejemplo, en [6] se utilizan las metaheurísticas Colonia de Abejas y Evolución Diferencial y en [7] la metaheurística GRASP.

La presente investigación pretende contribuir al desarrollo teórico de una propuesta algorítmica para la solución del problema WPP mono objetivo. Concretamente se analizará de manera experimental, la metaheurística Optimización con Colonia de Hormigas (ACO) [8] y una variante de exploración en dos etapas (TS-ACO) [9] que ha sido aplicada a varios

Article history:

Received 01 March 2017

Accepted 22 May 2017

Los autores J.C. Pisco-Vanegas (jpisco@uteq.edu.ec), A. A. Bayas-Zamora (abayas@uteq.edu.ec), R. Najarro-Quintero (rnajarro@uteq.edu.ec), A. Puris Cáceres (apuris@uteq.edu.ec), y Pavel Novoa-Hernández (pnovoa@uteq.edu.ec) son docentes de la Universidad Técnica Estatal de Quevedo. Av. Quito km. 1 1/2 vía a Santo Domingo de los Tsáchilas. Quevedo-Los Ríos-Ecuador. Tel: (+593) 5 3702-220 Ext. 8001

problemas de optimización combinatoria con similares características que el WPP y los resultados han sido prometedores [10].

Para una mejor comprensión de la investigación, el resto del trabajo queda organizado de la forma siguiente. En la Sec. II se define matemáticamente el problema de la planificación de la fuerza de trabajo. En la Sec. III se describen los fundamentos de la metaheurística Optimización con Colonia de Hormigas, variante dos etapas. La manera en que el modelo matemático de la Sec. II es transformado para su solución por parte de la técnica empleada, es descrito en la Sec. IV. Para validar la propuesta, se realizó el estudio experimental descrito en la Sec. V. Finalmente, en la Sec. VI se brindan las conclusiones principales que fueron obtenidas.

II. EL PROBLEMA DE LA PLANIFICACIÓN DE LA FUERZA DE TRABAJO

En el problema de Planeamiento de la Fuerza de Trabajo (Worker Planning Problem, WPP) [11] se tiene un conjunto de trabajos $J = \{1, \dots, m\}$ que deben completarse durante el próximo período de la planificación (por ejemplo, una semana). Cada trabajo j requiere un número de horas d_j durante el período de la planificación.

Dado el conjunto $I = \{1, \dots, n\}$ de obreros disponibles. La disponibilidad de cada obrero i durante el período de la planificación es s_i horas. Por las razones de eficiencia, un obrero debe realizar un número mínimo de horas (h_{min}) de cualquier trabajo al que se asigne y, al mismo tiempo, ningún obrero puede ser asignado a más de un número de trabajos (j_{max}) durante el período de la planificación.

Los trabajadores tienen diferentes habilidades, por lo tanto, sea A_i el conjunto de trabajos para los cuales el trabajador i está calificado para realizar. No más de t trabajadores pueden ser asignados durante el período de planeamiento. En otras palabras, a lo sumo t trabajadores pueden ser seleccionados del conjunto I de n trabajadores y el subconjunto seleccionado debe ser capaz de completar todos los trabajos. El objetivo es encontrar una solución factible que optimice la función objetivo. El término c_{ij} se utiliza como el costo de asignarle al trabajador i el trabajo j . Luego, el planteamiento matemático del problema quedaría como:

Variables de decisión

$y_i = \begin{cases} 1, & \text{si el trabajador } i \text{ es seleccionado} \\ 0, & \text{si no se selecciona el trabajador } i \end{cases}$

$x_{ij} = \begin{cases} 1, & \text{si se asigna el trabajador } i \text{ al trabajo } j \\ 0, & \text{si no se asigna el trabajador } i \text{ al trabajo } j \end{cases}$

z_{ij} , número de horas que tiene el trabajador i para desarrollar el trabajo j .

A_j , conjunto de trabajos para los cuales el trabajador i está calificado.

Q_j , conjunto de trabajadores calificador para realizar el trabajo j .

Función objetivo

$$\text{Minimizar } \sum_{i \in I} \sum_{j \in A_i} C_{ij} x_{ij} \quad (1)$$

Sujeto a:

$$\sum_{j \in A_i} z_{ij} \leq s_i y_i, \quad \forall i \in I \quad (2)$$

$$\sum_{j \in Q_j} z_{ij} \geq d_j, \quad \forall j \in J \quad (3)$$

$$\sum_{j \in A_j} x_{ij} \leq j_{MAX} y_i, \quad \forall i \in I \quad (4)$$

$$h_{MIN} x_{ij} \leq z_{ij} \leq s_i x_{ij} \quad \forall i \in I, \quad j \in A_i \quad (5)$$

$$\sum_{i \in I} y_i \leq t \quad (6)$$

$$x_{ij} \in \{0,1\} \quad \forall i \in I, \quad j \in A_i \quad (7)$$

$$y_i \in \{0,1\} \quad \forall i \in I \quad (8)$$

$$z_{ij} \geq 0 \quad \forall i \in I, \quad j \in A_i \quad (9)$$

En este modelo, la función objetivo (1) minimiza el costo de la asignación total. La restricción (2) limita el número de horas para cada trabajador seleccionado. Si el obrero no es escogido, entonces esta restricción no permite ninguna asignación de horas al trabajador. La restricción (3) enfatiza en los requisitos del trabajo, como ha sido especificado el número de horas necesitado completar cada trabajo durante el período de la planificación. La restricción (4) pone límite al número de veces que un trabajador puede ser asignado a realizar trabajos. La restricción (5) limita el mínimo de horas que un trabajador puede ser asignado a laborar en un trabajo, o sea, una vez asignado el trabajador debe realizar un número mínimo de horas; también esta restricción no permite que se le asignen horas de trabajo a un trabajador que no ha sido seleccionado para realizar un trabajo. La restricción (6) limita el número de obreros escogido durante el período planeando.

III. OPTIMIZACIÓN BASADA EN COLONIA DE HORMIGAS EN DOS ETAPAS

La metaheurística Optimización basada en Colonia de Hormigas (Ant Colony Optimization, ACO) [12] es un modelo que imita el comportamiento de las hormigas en su ambiente natural, específicamente la habilidad que presentan en la búsqueda de sus alimentos. Desde su surgimiento en la década de los 90, ha sido utilizada para la solución de numerosos problemas de optimización combinatoria, como el Viajante de Comercio (*Travelling Salesman Problem*) [13], Ordenamiento Secuencial (*Sequential Ordering Problem*) [14], Ruteo de Vehículos (*Vehicle Routing Problem*) [15], Asignación Cuadrática (*Quadratic Assignment Problem*) [16], Redes de Telecomunicaciones (*Telecommunications Networks*) [12] entre otros.

Por su parte la Optimización basada en Colonia de Hormigas en dos Etapas [17], es una variante de exploración para los algoritmos de la metaheurística ACO, que se basa en el principio de Divide y Vencerás. Esta alternativa divide el proceso de exploración en dos fases, en la primera, solo un subconjunto de la colonia se utiliza para explorar una parte del problema, luego las mejores soluciones encontradas en esta

etapa, forma los estados de partida, donde las restantes hormigas de la colonia, inician su recorrido hasta completar las soluciones. Este mecanismo ha demostrado mejorar la eficiencia de los algoritmos ACO y ha sido probado en varios problemas de estudio [17] [18].

En su estructura, se utiliza un parámetro $r \in (0,1)$ que representa la dimensión de la exploración en cada etapa. Para ello, este parámetro afecta directamente, al tamaño de la colonia, la dimensión del problema y la condición de parada del algoritmo. La Fig. 1 muestra el pseudocódigo general de la estrategia. Donde los parámetros cs representa la condición de parada del algoritmo, m la cantidad de hormigas y nn el tamaño de las soluciones.

Algoritmo TS-ACO
<p>parámetros (factor (r), cantidad de soluciones parciales almacenadas (ns))</p> <p>Etapa 1. Exploración</p> <p><i>Paso 1:</i> Calcular los parámetros para la primera etapa</p> <ol style="list-style-type: none"> 1.1 $m1 = \text{round}(r * m)$ (cantidad de hormigas) 1.2 $cs1 = \text{round}(r * cs)$ (cantidad de iteraciones) 1.3 $nn1 = \text{round}(r * nn)$ (magnitud o tamaño de las soluciones) <p><i>Paso 2:</i> Aplicar un algoritmo ACO utilizando ($m1$, $cs1$, $nn1$) donde:</p> <ol style="list-style-type: none"> 2.1 las $m1$ hormigas parten de un estado seleccionado de forma aleatoria. 2.1 construyen soluciones de tamaño $nn1$ 2.3 el algoritmo se aplica $cs1$ iteraciones <p><i>Paso 3:</i> Almacenar las ns mejores sub soluciones encontradas</p> <p>Etapa 2. Intensificación</p> <p><i>Paso 1:</i> Calcular los parámetros para la segunda etapa.</p> <ol style="list-style-type: none"> 1.1 $m2 = m - m1$ (cantidad de hormigas) 1.2 $cs2 = cs - cs1$ (cantidad de iteraciones) 1.3 $nn2 = nn - nn1$ (magnitud o tamaño de las soluciones) <p><i>Paso 2:</i> Aplicar un algoritmo ACO utilizando ($m2$, $cs2$, $nn2$,) donde:</p> <ol style="list-style-type: none"> 2.1 las $m2$ hormigas se ubican aleatoriamente en una sub solución entre las ns obtenida en la etapa 1 2.2 incorporan a cada solución los restantes $nn2$ estados. 2.3 el algoritmo se aplica $cs2$ iteraciones <p>Fin_TS-ACO</p>

Fig 1. Estrategia de exploración en dos etapas TS-ACO.

Aplicar esta estrategia a la exploración de los algoritmos ACO trae consigo algunas ventajas, como son [9]:

- 1- Presenta menor complejidad computacional.
- 2- Tiene bien delimitado los mecanismos de exploración y explotación.
- 3- Mayor robustez, porque permite aplicar en cada etapa un algoritmo ACO diferente.

Como desventaja se puede evidenciar que no es aplicable a todo tipo de problema combinatorio. Este debe de separable y

brindar la posibilidad de evaluar soluciones en cualquier etapa del proceso de construcción.

IV. MODELO COMPUTACIONAL DE LA PROPUESTA

Para aplicar la estrategia TS-ACO al problema de Planificación de la Fuerza de Trabajo es necesario definir 4 aspectos fundamentales:

La representación del problema en forma de grafo: la estructura de las instancias del problema contiene entre otras cosas una matriz de costos C , donde las filas representan trabajadores y las columnas los trabajos. Bajo esta representación, una celda $C_{ij} = 0$ determina que el trabajador i no está capacitado para realizar el trabajo j , por lo que no puede formar parte de una solución. Bajo este supuesto, se construye un grafo $G=(N,A)$, donde los nodos N representan las celdas de la matriz de costos (*trabajador-trabajo*) y los arcos (A) la posibilidad de alcanzar otra celda. Los nodos que identifican celdas con $C_{ij} > 0$ se encuentran totalmente conexo, mientras que los nodos con costo 0 se encuentran aislados (sin conexión). Con esta representación las hormigas pueden moverse por el grafo construyendo una solución que garantice asignaciones consistentes.

La representación de los rastros de feromona o información memorística de la colonia: Para los rastros de feromona se emplea una matriz cuadrada $N \times N$, donde N representa la cantidad de nodos del grafo. Los valores en la matriz determinan la preferencia de la colonia de hormigas por asignar un trabajador a un trabajo. Los nodos inaccesibles estarán representados por el valor 0 en dicha matriz de feromona.

Función heurística o de visibilidad de la colonia: Este elemento es elemental para el buen desempeño del algoritmo y luego de algunos estudios se definió por la ecuación (10):

$$\eta_{p,k} = 1 - \frac{|T_{TRABAJADOR(p)} - T_{TRABAJO(k)}|}{C_{pk}} \quad (10)$$

donde $T_{TRABAJADOR(p)}$ representa el tiempo disponible que le queda al trabajador p para terminar su jornada, mientras que $T_{TRABAJO(k)}$ determina el tiempo que le queda al trabajo k para ser completado (esta información está disponible en las instancias que se utilizan en el análisis experimental). Como se puede apreciar, esta función favorece la asignación que conlleve a reducir el tiempo de terminación tanto de los trabajadores como de los trabajos pendientes, teniendo en cuenta el costo de la asignación.

Selección del Algoritmo ACO: Dentro de esta meta heurística existen varios algoritmos documentados, en nuestro caso seleccionamos el algoritmo Max-Min Ant System, por ser de los más utilizados en el estado del arte. Este algoritmo introduce una mejora al algoritmo Ant System para controlar la convergencia prematura. Para ello, utiliza una cota superior e inferior que controlan que los valores de feromona, (ver ecuación 11):

$$\begin{aligned} \tau_{MAX} &= 1/(1 - CEvap) * 1/Cost(S) \\ \tau_{MIN} &= \tau_{MAX}/10 * Nt \end{aligned} \quad (11)$$

20	20	10	1000	54	3	4													
0	0	0	0	0	0	32	0	36	0	52	40	0	56	0	36	0	0	0	0
0	39	0	0	0	51	27	0	0	0	0	0	0	0	35	37	0	0	0	0
0	38	43	0	0	0	0	0	0	41	55	50	55	55	35	0	0	61	0	0
0	0	0	0	45	0	0	63	34	0	57	50	0	0	36	0	0	0	35	0
0	0	0	59	0	57	0	58	0	46	0	44	55	0	0	43	41	0	41	0
0	0	45	0	49	60	40	0	0	47	57	0	0	63	0	39	47	0	42	0
40	0	0	0	0	52	0	57	0	37	0	0	49	54	0	0	0	0	0	57
0	0	0	49	44	0	0	0	0	0	53	0	0	0	36	0	41	59	0	0
38	0	0	0	0	0	33	0	0	0	0	0	0	0	33	32	0	60	34	0
0	0	40	0	0	0	37	61	0	48	0	48	59	58	45	0	42	63	47	58
0	0	0	0	41	58	0	65	0	45	64	49	60	0	46	40	0	69	38	0
0	0	39	0	44	61	39	0	44	50	57	47	0	0	38	0	0	42	62	0
45	0	37	0	45	0	39	62	38	44	0	53	58	61	0	0	44	0	0	0
0	34	0	0	0	0	0	53	0	0	0	0	0	0	32	0	38	0	37	57
0	40	34	0	0	52	29	61	37	0	0	44	0	0	0	0	0	0	0	0
0	0	0	0	0	56	0	0	41	0	0	0	0	0	35	0	41	0	39	60
0	0	0	0	0	0	0	0	37	42	55	46	0	0	0	0	0	57	0	57
0	0	37	58	0	52	0	0	35	0	62	0	53	0	0	39	0	0	0	63
46	47	48	0	0	55	0	62	0	0	63	48	63	63	46	0	0	64	43	60
0	0	0	0	0	0	0	0	0	35	0	46	0	54	0	0	0	55	0	0
20																			
16																			
16																			
32																			
20																			
32																			
12																			
36																			
16																			
20																			
40																			
16																			
36																			

Fig 2. Esquema de una instancia del problema WPP.

TABLA I.
PARÁMETROS UTILIZADOS EN EL ESTUDIO EXPERIMENTAL.

	m	cs	nn	r	ns	CEvap	α	β
MMAS	20	Cantidad de iteraciones 1000	Todas las tareas terminadas	-	-	0.7	2	3
TS-MMAS	20	Cantidad de iteraciones 1000	Todas las tareas terminadas	0.2	10	0.7	2	3
P-TS-MMAS	20	Cantidad de iteraciones 1000	Todas las tareas terminadas	0.2	-	0.7	2	3

donde $CEvap$ representa la constante de evaporación de la feromona, $Cost(S)$ el valor del costo de la mejor solución (S) encontrada por el algoritmo y Nt la cantidad de trabajadores utilizados en la instancia. Otro elemento importante que incorpora el algoritmo *Max-Min Ant System* es la manera en que actualiza los rastros de feromona. La ecuación (12) se utiliza para disminuir (proceso de evaporación) los valores de la matriz τ y la ecuación (13) aumenta dicho valor en los componentes de la solución S , con un valor proporcional al costo de esta ($Cost(S)$).

$$\tau_{ij} = (1 - CEvap) * \tau_{ij} \quad (12)$$

$$\tau_{rs} = \tau_{rs} + Cost(S), \forall a(r, s) \in S \quad (13)$$

Por último, el algoritmo utiliza una función de transición de estado para que cada hormiga se desplace por el grafo construyendo su solución. Dicha función (ecuación 14) determina un valor probabilístico para cada estado vecino N_i^k , utilizando para ello la información heurística y el valor de feromona, donde α y β representan parámetros de influencia del algoritmo.

$$p_{ij} = \frac{(\tau_{ij})^\alpha * (\eta_{ij})^\beta}{\sum_{j \in N_i^k} (\tau_{ij})^\alpha * (\eta_{ij})^\beta}, \forall j \in N_i^k \quad (14)$$

En este trabajo se estudia además una variante de la exploración en dos etapas basada en feromona (P-TS-MMAS), en la cual no se almacenan las mejores ns soluciones encontradas en la primera etapa, por lo cual, las hormigas que exploran en la segunda etapa comienzan de un nodo del grafo seleccionado aleatoriamente como sucede con las hormigas de la primera etapa. Lo único que se reutiliza es la información memorística almacenada en la matriz de feromona reduciendo de esta forma los niveles de intensificación del algoritmo aumentando la exploración en la segunda etapa.

En términos algorítmicos la variante P-TS-ACO aplica los mismos pasos que el algoritmo TS-ACO descrito en la Fig. 1 omite el paso 3 de la etapa 1 y los pasos 2.1 y 2.2 de la etapa 2.

V. ESTUDIO EXPERIMENTAL

TABLA II.
RESULTADOS DE CADA ALGORITMO INVOLUVRADO EN EL ESTUDIO

Instancia	MMAS	TS-MMAS	P-TS-MMAS	AG	RS
s20_01	892.9	923.8	897.7	963	939
s20_02	956	965.5	944	994	952
s20_03	959	983.8	972.2	1156	1095
s20_04	994.3	1022.1	951.5	1201	1043
s20_05	1013	1038.5	993.9	1098	1099
s20_06	950.3	962.6	978.9	1193	1076
s20_07	963.7	971.1	934.2	1086	987
s20_08	1073.3	1083.5	994	1287	1293
s20_09	938.8	980.4	909.9	1107	1086
s20_10	891.3	915.8	890.5	1086	945
u20_01	1280.9	1307.1	951.9	1631	1586
u20_02	1061.1	1167.4	960.1	1264	1276
u20_03	1065.8	1120.4	1025.1	1539	1502
u20_04	1207.9	1357.1	1023	1603	1653
u20_05	1021.3	1195.5	969.2	1356	1287
u20_06	1088.7	1068.9	1016.4	1205	1193
u20_07	993.5	1096.8	970.5	1301	1328
u20_08	1308.7	1257.9	940.7	1106	1141
u20_09	821.3	864.4	832.4	1173	1055
u20_10	964.6	997.1	945.2	1214	1178

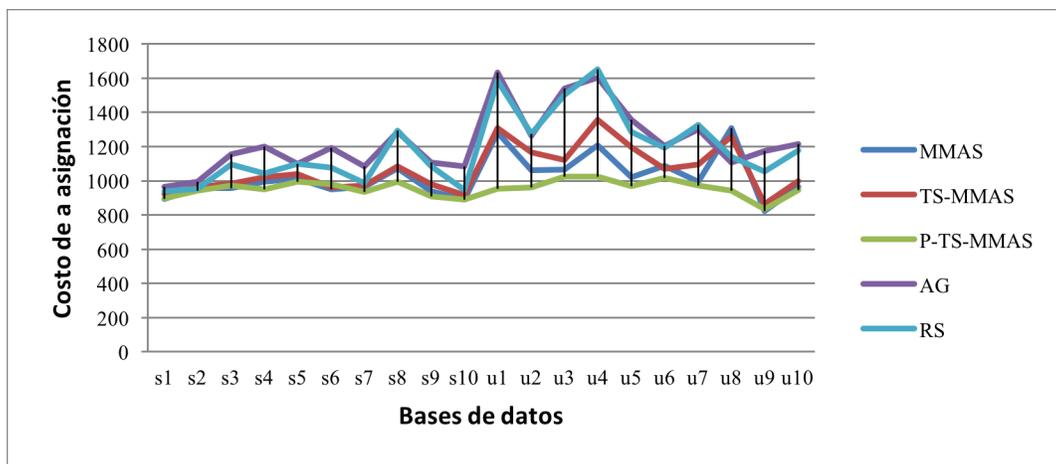


Fig 3. Resultados comparativos entre los algoritmos considerados.

TABLA III.
RESULTADOS DEL TEST DE HOLM

	AG		RS		MMAS		TS-MMAS	
	Valor p	$\alpha/4$	Valor p	$\alpha/3$	Valor p	$\alpha/2$	Valor p	$\alpha/1$
P-TS-MMAS	2.0841E-11	0.0125	1.1980E-08	0.0166	4.6525E-04	0.025	0.1095	0.05

Para el estudio experimental, se implementó¹ el algoritmo Max-Min Ant System de la meta heurística ACO y además de las dos alternativas de la exploración en dos etapas TS-MMAS y P-TS-MMAS presentadas anteriormente. Las instancias utilizadas se obtuvieron de [5] así como los resultados de un Algoritmo Genético y el Recocido Simulado en su versión secuencial.

A continuación, se presenta una breve descripción de las instancias utilizadas (ver Fig. 2):

- 1- La primera fila representa por su orden; el número de trabajadores, número de trabajos, número máximo de trabajos que puede contener una solución, un dato que no se utiliza, tiempo máximo que puede ser asignado un trabajador, máximo número de tareas que puede realizar un trabajador, tiempo mínimo que puede ser asignado un trabajador.
- 2- Segundo, una matriz de costo de orden cantidad de trabajadores x cantidad de trabajos, donde el valor cero significa que el trabajador no está calificado para realizar dicho trabajo, como se había explicado anteriormente.

¹ El código fuente está desarrollado en java y puede ser entregado a petición de los interesados con los autores.

3- Por último, el vector columna que aparece representa la duración de los trabajos en cuanto a tiempo.

Por su parte la Tabla I muestra los parámetros utilizados por cada algoritmo propuesto en este trabajo. Estos valores fueron el resultado de un estudio de parámetros más extenso.

La Tabla II muestra los valores promedio sobre 25 ejecuciones independientes de cada algoritmo propuesto, así como los presentados en [5].

Además, se realizó un análisis estadístico para determinar si las diferencias obtenidas a favor del P-TS-MMAS son realmente significativas. Para ellos se aplicó el test no paramétrico de Friedman [19] para k muestras relacionadas con un valor de significancia de 0.05 donde el valor p obtenido ($3.8339E-11$) determina que existen diferencias significativas en el grupo.

Seguidamente se aplicó el test de comparaciones múltiples de Holm [20] tomando como muestra de control el algoritmo P-TS-MMAS y aplicándose sucesivas comparaciones con cada uno de los algoritmos restantes para un $\alpha = 0.05$, donde la aparición de un valor p menor que su correspondiente α/i , determina la existencia de diferencias significativas a favor del algoritmo de control.

Para este caso de estudio, se puede apreciar en la Tabla III, que el algoritmo P-TS-MMAS es mejor significativamente que las dos variantes de referencia (Algoritmo Genético y Recocido Simulado) y que la propuesta MMAS. Solo no supera significativamente los resultados de la otra variante de la exploración en dos etapas estudiada en este trabajo.

Estos resultados demuestran que la exploración en dos etapas aumenta el nivel de exploración del espacio de búsqueda, garantizando soluciones de mejor calidad en el escenario de estudio.

VI. CONCLUSIONES

Como resultado de este trabajo se logró realizar una modelación del Problema de Planeamiento de las Fuerzas de Trabajo que permitió la aplicación de la Meta-heurística ACO específicamente el algoritmo MMAS. En este modelo se definió una función heurística que dirige la exploración hacia los estados con menor tiempo de terminación teniendo en cuenta los costos de dicha asignación.

Por otra parte, se presentó una modificación a la estrategia de exploración en dos etapas, la cual no utiliza implícitamente las mejores sub soluciones obtenidas en la primera etapa, solo la información memorística almacenada en la matriz de feromona. Esta modificación garantiza aumenta los niveles de exploración en la segunda etapa, obteniendo los mejores resultados en el análisis comparativo realizado para un conjunto de 20 instancias del problema.

En el análisis estadístico desarrollado se pudo constatar que los resultados obtenidos por P-TS-MMAS son significativamente mejores que dos algoritmos de referencia del estado de arte (Algoritmo Genético y Recocido Simulado), así como la versión original del MMAS.

Como líneas futuras se plantea estudiar mecanismos auto-adaptativos que posibiliten el ajuste de los parámetros decisivos de la metaheurística durante la ejecución. Existen evidencias sólidas [21], [22] que indican que este tipo de

técnica de control de parámetros resulta efectiva en contextos complejos como el tratado en el presente trabajo.

REFERENCIAS

- [1] P. De Bruecker, J. Van den Bergh, J. Beliën, and E. Demeulemeester, "Workforce planning incorporating skills: State of the art," *Eur. J. Oper. Res.*, vol. 243, no. 1, pp. 1–16, 2015.
- [2] J. Wang, "A review of operations research applications in workforce planning and potential modelling of military training," DSTO, 2005.
- [3] E. D. Philippe De Bruecker, Jorne Van den Bergh, Jeroen Beliën, "Workforce planning incorporating skills: State of the art," *Eur. J. Oper. Res.*, vol. 243, no. 1, pp. 1–16, 2015.
- [4] I. Boussaid, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Inf. Sci. (Nijl)*, vol. 237, pp. 82–117, 2013.
- [5] E. Alba, G. Luque, and F. Luna, "Parallel metaheuristics for workforce planning," *J. Math. Model. Algorithms*, vol. 6, no. 3, pp. 509–528, 2007.
- [6] K. M. Iman Fozveh, Hooman Salehi, "Presentation of Multi-Skill Workforce Scheduling Model and Solving the Model Using Meta-Heuristic Algorithms," *Mod. Appl. Sci.*, vol. 2, no. 10, pp. 194–205, 2016.
- [7] E. M. G.-N. Diego Novoa, Camilo Olarte, David Barrera, "A GRASP-based approach to the multi activity combined timetabling and crew scheduling problem considering a heterogeneous workforce," *Int. J. Ind. Eng. Comput.*, vol. 7, no. 4, pp. 597–606, 2016.
- [8] M. Dorigo and T. Stützle, *Ant colony optimization*. MIT Press, 2004.
- [9] A. Puris, "Desarrollo de meta-heurísticas poblacionales para la solución de problemas complejos," Universidad Central "Marta Abreu" de Las Villas, 2010.
- [10] A. Puris, R. Bello, and F. Herrera, "Analysis of the efficacy of a Two-Stage methodology for ant colony optimization: Case of study with TSP and QAP," *Expert Syst. Appl.*, vol. 37, no. 7, 2010.
- [11] T. Glover, F. Kochenberger, G. Laguna, M. Wubben, "Selection and assignment of a skilled workforce to meet job requirements in a fixed planning period," in *MAEB*, 2004, pp. 636–641.
- [12] M. Dorigo, M. Birattari, and T. Stützle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006.
- [13] A. Puris, R. Bello, Y. Martínez, and A. Nowe, *Two-stage ant colony optimization for solving the traveling salesman problem*, vol. 4528 LNCS, no. PART 2. 2007.
- [14] L. M. Gambardella and M. Dorigo, "HAS-SOP: Hybrid Ant System For The Sequential Ordering Problem." 1997.
- [15] J. E. Bell and P. R. McMullen, "Ant colony optimization techniques for the vehicle routing problem," *Adv. Eng. Informatics*, vol. 18, no. 1, pp. 41–48, 2004.
- [16] T. Stützle and H. H. Hoos, "MAX-MIN Ant System," *Futur. Gener. Comput. Syst.*, vol. 16, no. 8, pp. 889–914, 2000.
- [17] A. Puris, R. Bello, and F. Herrera, "Analysis of the efficacy of a Two-Stage methodology for ant colony optimization: Case of study with TSP and QAP," *Expert Syst. Appl.*, vol. 37, no. 7, pp. 5443–5453, 2010.
- [18] R. Bello, A. Puris, A. Nowe, Y. Martínez, and M. M. García, "Two step ant colony system to solve the feature selection problem," in *PROGRESS IN PATTERN RECOGNITION IMAGE ANALYSIS AND APPLICATIONS PROCEEDINGS*, 2006, vol. 4225, pp. 588–596.
- [19] G. G. Brunk, "A test of the Friedman-Savage gambling model," *Q. J. Econ.*, vol. 96, no. 2, pp. 341–348, 1981.
- [20] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 Special Session on Real Parameter Optimization," *J. Heuristics*, vol. 15, pp. 617–644, 2009.
- [21] P. Novoa-Hernández, C. C. Corona, and D. A. Pelta, "Self-adaptive, multipopulation differential evolution in dynamic environments," *Soft Comput.*, vol. 17, no. 10, pp. 1861–1881, Mar. 2013.
- [22] P. Novoa-Hernández, C. C. Corona, and D. A. Pelta, "Self-adaptation in dynamic environments - a survey and open issues," *Int. J. Bio-Inspired Comput.*, vol. 8, no. 1, pp. 1–13, 2016.



Juan Carlos Pisco Vanegas. Es docente de la Facultad de Ciencias de la Ingeniería de la Universidad Técnica Estatal de Quevedo, Los Ríos, Ecuador. Es Licenciado y Máster en Tecnologías en las ramas de Electromecánica, Electrotecnia y Electrotecnología (SPBGPU).

por la Universidad de Granada (España). Ha sido Profesor Auxiliar en la Universidad de Holguín en el período 2007-2015. Sus intereses investigativos, relacionados con la Soft Computing, incluyen: problemas dinámicos de optimización, meta-heurísticas, problemas de optimización multi-objetivo, entre otros.



Abraham A. Bayas-Zamora. Es profesor de Física y Álgebra de la facultad de Ciencias Agrarias de la Universidad Técnica Estatal de Quevedo. Recibió el grado de Ingeniero Civil, especialidad estructuras en la Universidad Técnica de Ambato, y en el año 2016 el grado de Magister en Gerencia de Innovaciones Educativas en la Universidad Técnica Estatal de Quevedo. Se inició como docente en la Facultad de Ingeniería Agronómica en la Universidad técnica de Ambato, luego se desempeñó como Docente de Matemáticas y Física en el Sistema Nacional de Nivelación y Admisión del Ecuador (Senescyt). El principal interés ha sido la docencia, aunque también ha dedicado su tiempo a la producción y mejoramiento del cacao CNN51.



Rodolfo Najarro Quintero. Es Ingeniero Mecánico por el Instituto Politécnico José A. Echevarría, La Habana, Cuba 1997. Se Diplomó en Marketing Internacional por la Universidad de La Habana, Cuba 1999. En 2015 obtuvo el título de Master en Conectividad y Redes de Ordenadores por la Universidad Técnica Estatal de Quevedo, Ecuador. Es docente de la Universidad Técnica Estatal de Quevedo.



Amilkar Puris Cáceres. Ph.D. en Ciencias Técnicas por la Universidad Marta Abreu de las Villas, Cuba. Sus principales investigaciones han sido en el área de las Meta-heurísticas Poblacionales para la solución de problemas complejos. Actualmente se desempeña como docente e investigador en la Universidad Técnica Estatal de Quevedo, Ecuador.



Pavel Novoa-Hernández. Actualmente trabaja como Profesor Agregado de la Universidad Técnica Estatal de Quevedo, Los Ríos, Ecuador, y como Profesor Invitado en la Universidad Estatal de Milagro. Es Ingeniero Informático por la Universidad de Holguín (Cuba) desde el 2007. Obtuvo el título de Master en Ciencias de la Computación e Inteligencia Artificial por la Universidad Central de Las Villas (Cuba), en el 2010. En el 2013 obtuvo el grado de Doctor (Ph.D) en Tecnologías de la Información y la Comunicación