International Journal of Mathematical Archive-5(9), 2014, 183-192
IJMA Available online through www.ijma.info ISSN 2229 – 5046

A NOVEL MULTI-ANT COLONY OPTIMIZATION FOR MULTI-OBJECTIVE RESOURCE ALLOCATION PROBLEMS

R. M. Rizk-Allah*

Department of Basic Engineering Science, Faculty of Engineering, Shebin El-Kom, Minoufia University, Egypt.

(Received On: 30-07-14; Revised & Accepted On: 16-08-14)

ABSTRACT

This paper presents a novel multi-ant colony optimization (NM-ACO) for solving multi-objective resource allocation Problems (MORAPs). The proposed algorithm differs from the traditional ones in its design the vector of colonies associated with the vector of objective functions as well as the MORAP is formulating as a connected graph ,where the ant construct the solution by assigning the amount of resource to the i^{th} stage by roaming on connected graph . On the other hand, the local search scheme which makes the ants moves to new rich regions. Moreover, the proposed algorithm introduced an extended memory to store global Pareto solutions to reduce computational time. Effectiveness and efficiency of proposed algorithm was validated by comparing the result of NM-ACO with multi-objective hybrid genetic algorithm (mohGA) which was applied to MORAP later. Also the comparative study demonstrated the superiority of the proposed algorithm and confirm its potential to solve the multi-objective problems.

***Keywords**: Artificial intelligence; Ant colony optimization; Multi-objective optimization; Resources allocation problem.*

1. INTRODUCTION

Decision makers usually need to allocate constrained resource among activities for optimizing the objectives. For instance, project budgeting [2] maximizes the profit return by allocating a fixed amount of budget money among a number of projects, software testing [3,4] guarantees the maximum reliability by allotting testing resource to program modules, task allocation [19] allocates a given number of tasks to a number of distributed processors for minimizing the incurred cost, health care financing [13,17] allocates health care resource across competing programs promising improved health for patients, just to name a few. These real-world scenarios can be all described by the resource allocation problem (RAP).

Several formulations for the RAP have been proposed in accordance with different problem scenarios. Single objective RAP seeks to optimize a single goal, such as benefit maximization or cost minimization [21]. Multiple-objective RAP (MORAP) optimizes a set of goals simultaneously which may involve benefit-type objectives to be maximized and cost-type objectives to be minimized. Linear RAP optimizes linear objectives while nonlinear resource allocation problem deals with nonlinear objective functions, which can be solved using analytical approach, however, the nonlinear form has been shown to be NP-hard [12]. The limited resource to be allocated can be either discrete or continuous, and the amount of resource units to be allocated to an activity may be constrained in a specified range. A comprehensive survey related to RAP can be found in [12].

The development of meta-heuristic optimization theory has been flourishing [14] . Many meta-heuristic paradigms such as genetic algorithm [10], simulated annealing [15], and tabu search [9] have shown their efficacy in solving computationally intensive problems. Among them, K. Fan *et al.* [8] proposed a modified binary particle swarm optimization and Chi and Mitsuo [5] presented a hybrid genetic algorithm for the same problem.

In this paper we present a new algorithm for solving MORAP based on the ACO which was recently developed by Dorigo [7]. The proposed algorithm differs from the traditional ones in its design the vector of colonies associated with the vector of objective functions as well as inclusion of local search scheme which makes the ants move to new rich regions. Moreover, the proposed algorithm introduced an extended memory to store global Pareto solutions to reduce computational time.

Corresponding Author: R. M. Rizk-Allah*

This paper is organized as follows. In Section 2 we describe some preliminaries on Mathematical formulation of MORAP and the Pareto optimal solutions. Section 3 presents the proposed algorithm for tackling the MORAP in details. Section 4 reports the comparative performance of proposed NM-ACO with Hybrid genetic algorithm and also convergence analysis. Finally, Section 5 concludes this work.

2. PRELIMINARIES

2.1. Mathematical formulation

The mathematical model for the K -objectives resource allocation problem (K-RAP) to assign M staffs to N different projects is formulated as a multicriteria integer programming model as follows:

$$Max\ z_1(\mathbf{y}) = \sum_{i=1}^{N} f_1(y_i)$$

$$\vdots$$

$$Min\ z_K(\mathbf{y}) = \sum_{i=1}^{N} f_K(y_i) \tag{1}$$

$$subject\ \ to$$

$$G_0(y) = \sum_{i=1}^{N} g_i(y_i) \le M \quad ; \ y_i = 0,1,...,M \ \ \forall i$$

According to above-mentioned mathematics model, we can extend it to handle a bicriteria resource allocation problem, to assign M workers to N different jobs for maximizing the benefit and minimizing the cost subject to one resource constraint. The Notations used in the MORAP is described as in Figure 1 and the MORAP is formulated as a bicriteria integer programming model as follows:

$$Max\ z_1(\mathbf{x}) = \sum_{i=1}^{N} \sum_{j=0}^{M} p_{ij} x_{ij} \tag{2}$$

$$Min\ z_2(\mathbf{x}) = \sum_{i=1}^{N} \sum_{j=0}^{M} c_{ij} x_{ij} \tag{3}$$

Subject to

$$G_0(\mathbf{x}) = \sum_{i=1}^{N} \sum_{j=0}^{M} j x_{ij} \le M; \tag{4}$$

$$G_i(\mathbf{x}) = \sum_{j=0}^{M} x_{ij} = 1, \ \forall i; \tag{5}$$

$$x_{ij} = 0\ or\ 1 \ \ \forall i, j.$$

Indices	
k	index of objective function, $k = 1,2,...,K$.
i	index of district shop, $(i = 1,2,..., N)$
j	index of salesclerk, $(j = 1,2,..., M)$
Parameter	
N	total number of district shop
M	total number of salesclerk
c_{ij}	cost of district i when j salesclerks are assigned
p_{ij}	profit of district i when j salesclerks are assigned
Decision variables	
$x_{ij} = \begin{cases} 1, & \text{if } j \text{ salesclerks are assigned to district } i, \\ 0, & \text{otherwise.} \end{cases}$	

Figure-1. Notations used in the MORAP formulation

The objective function (2) is to maximizing the benefit for all the jobs. And the objective (3) is to minimizing the total costs for all the workers. Constraint (4) ensures that we cannot assign the workers more than the total numbers of workers. Constraint (5) ensures that for each job i we just can only assign workers for it one time. The Pareto optimal solutions are usually characterized as solutions of the multi-objective programming problem [13]. Therefore, in implementation of NM-ACO algorithms, a module for handling Pareto optimal solutions is added. It consists of three steps:
Step 1: Designing the vector of colonies associated with the objective functions.
Step 1: Adapting the local search scheme to make the ants move to new rich regions.
Step 2: Select Pareto solutions based on the procedure described in subsection 2.2.

2.2. The Pareto-optimal solutions

For a problem having more than one objective function, any two solutions $\mathbf{x}_1$ and $\mathbf{x}_2$ can have one of two possibilities – one dominates the other or none dominates the other. The definitions are described as follows [13]:

Definition 1 (Dominance): For minimal problem, a solution $\mathbf{x}_1 \in \Omega$ ($\Omega \in \mathrm{Z}$ is the feasible region and $\mathrm{Z} = \{0,1,2,...,...\}$ is the set if positive integer including 0, i.e., $\Omega = \left\{ \mathbf{x} \in \mathrm{Z} | G_0(\mathbf{x}) = \sum_{i=1}^{N} \sum_{j=0}^{M} j x_{ij} \le M \wedge G_i(\mathbf{x}) = \sum_{j=0}^{M} x_{ij} = 1, \forall i \right\}$) is dominating a solution $\mathbf{x}_2 \in \Omega$ (briefly written as $\mathbf{x}_1 \succ \mathbf{x}_2$ for minimization) if and only if it is superior or equal in all objectives and at least superior in one objective. This can be expressed as:

$$\mathbf{x}_1 \succ \mathbf{x}_2 \quad \text{, if} \begin{cases} \forall i \in 1,2,...,K: \ f_i(\mathbf{x}_1) \le f_i(\mathbf{x}_2), \\ \wedge \exists\, j \in 1,2,...,K: \ f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2). \end{cases} \tag{6}$$

Definition 2 (Pareto-optimality): Let $\mathbf{x}_1 \in \Omega$ be an arbitrary decision vector.

(a) The decision vector $\mathbf{x}_1 \in \Omega$ is said to be non-dominated regarding the set $\Omega' \subseteq \Omega$ if and only if there is no vector $\mathbf{x}_2$ in Ω' which can dominate $\mathbf{x}_1$. Formally, $\nexists \mathbf{x}_2 \in \Omega'$, $\mathbf{x}_2 \succ \mathbf{x}_1$.

(b) The decision (parameter) vector $\mathbf{x}_1$ is called Pareto-optimal if and only if $\mathbf{x}_1$ is non-dominated regarding the whole parameter space Ω.

3. THE PROPOSED NM-ACO FOR MORAP

Ant colony optimization [6] is one of the most recent meta-heuristic techniques for approximate optimization. The inspiring source of ACO algorithms are real ant colonies. More specifically, ACO is inspired by the ants' foraging behavior. At the core of this behavior is the indirect communication between the ants by means of chemical pheromone trails, which enables them to find short paths between their nest and food sources. This characteristic of real ant colonies is exploited in ACO algorithms in order to solve optimization problems such as the traveling salesman problem [1], scheduling problem [11], minimum weight vertex cover problem [16], and curve segmentation problem [20], multi-objective optimization problem [18], just to name a few.

The proposed algorithm differs from the traditional ones in its design the vector of colonies associated with the vector of objective functions as well as inclusion of local search scheme which makes the ants move to new rich regions. Moreover, the proposed algorithm introduced an extended memory to store global Pareto solutions to reduce computational time. Generally, we can describe the steps of the proposed algorithm as follows:

3.1. The implementation

Implementing the ACO for a MORAP requires a representation of N stages for each ant, with each stage $n, n = 1,2,..., N,$ has a set of M states (i,e., connected graph $G = (N, M)$ see Figure 2). Consequently, a feasible and complete solution of the formulated multi-objective resource allocation problem is considered as a permutation of resource allocation.

3.2. The Initialization

The NM-ACO algorithm assumes that K colonies $C_1, C_2,..., C_{\mathbf{K}}$ each of size m aim to optimize simultaneously K-objective functions. Each colony $C_k^{(t=0)}$ is initialized randomly within the feasible solution space (satisfying all constraints). Each part of this solution is termed state (node) where each node denotes K criteria with their associated initial pheromone concentration $\left\{\tau_{ij}^{(t=0)}\right\}_{k=1}^{K}$ and the associated heuristic value η_{ij}^k where $i = 1,2,..., N,\ j = 1,2,..., M$.

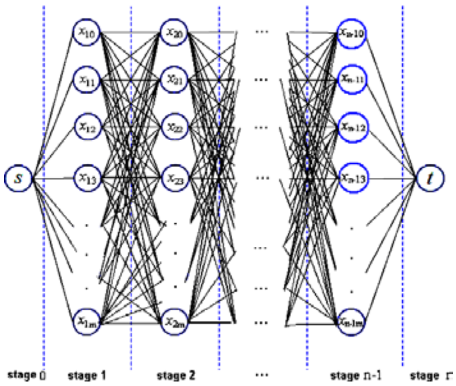

Figure-2. Graph representation of MORAP

3.3. Construction of solutions

Each ant constructs a solution as follows. First, one of the nodes of the MORAP graph is randomly chosen as start node. Then, the ant builds a tour in the MORAP graph by roaming in each construction step from its current node to another node that is feasible for ant according to $Tabu_a$ (i.,e., $Tabu_a$ is the memory of ant a saving the index of feasible allocations and also saving the selected nodes by ant a). At each stage the traversed node is added to the solution under construction. This way of constructing a solution implies that an ant has a memory to store the already visited nodes.

Each solution construction step is performed as follows. Assuming the ant to be in node i, the subsequent construction step is done with probability as in Eq(7):

$$p_{ij}^k(t) = \begin{cases} \dfrac{[\tau_j^k(t)]^\alpha [\eta_j^k(t)]^\beta}{\sum\limits_{h \in tabu_a} [\tau_h^k(t)]^\alpha [\eta_h^k(t)]^\beta} & \text{if } \forall\, j,h \in tabu_a \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

$$\eta_j^1(t) = z_1(\mathbf{x}),\ \eta_j^2(t) = 1/(\varepsilon + z_2(\mathbf{x}))$$

where the exponents α and β are positive parameters whose values determine the relation between pheromone information τ_j^α and heuristic information η_j^β. ε is the small positive number; the reason due to which ε is added in the formula $\eta_j^2(t)$ to avoid division by 0.

3.4. Evaluation of non-dominated solutions

The set of non-dominated solutions is stored in an archive (denoted by $A^{(t)}$). During the optimization search, this set, which represents the Pareto front, is updated. At each iteration, the current solutions obtained are compared to those stored in the Pareto archive; the dominated ones are removed and the non-dominated ones are added to the archive. The mechanism of evaluating the non-dominated solutions and updating the archive is described as in Figure 3.

Input: $A^{(t)} = \phi; D = \phi$; $C = \{C_l\}_{l=1}^K, C_l = \{\mathbf{x}_a\}_{a=1}^m$.
$i = 1$.
If $\mathbf{x}_i \prec \forall \mathbf{x}_j \ \wedge i \neq j$ **then**
$D = D \cup \{\mathbf{x}_i\}$
Else
$A^{(t)} = A^{(t)} \cup \{\mathbf{x}_i\}$
End ; $i = i + 1$
output : $A^{(t)}$
	Input: $A^{(t)}, \mathbf{x}$
	If $\nexists \mathbf{x}^1 \in A^{(t)} \middle| \mathbf{x}^1 \succ \mathbf{x}$ **then**
	$A^{(t)} = A^{(t)} \cup \mathbf{x}$
	Else if $\exists \mathbf{x}^1 \in A^{(t)} \middle| \mathbf{x} \succ \mathbf{x}^1$ **then**
	$A^{(t)} = \{A^{(t)} \cup \mathbf{x}\} \setminus \{\mathbf{x}^1\}$
	Else
	$A^{(t)} = A^{(t)}$
	End
	output : $A^{(t)}$

Figure-3. The pseudo code of evaluating the non-dominated solutions (left) and updating the archive (right)

All these non-dominated solutions are assumed to constitute the non-dominated front in the population in a specified generation.

3.5. Pheromone update mechanism

When updating pheromone trails, one has to decide on which of the constructed solutions laying pheromone. The quantity of pheromone laying on a component represents the past experience of the colony with respect to choosing this component. Then, at each cycle every ant constructs a solution, and pheromone trails are updated. Once all ants have constructed their solutions, pheromone trails are updated as usually in equation 2: first, pheromone trails are reduced by a constant factor to simulate evaporation to prevent premature convergence; then, some pheromone is laid on components of the best solution. The mechanism of update the Pheromone is as follows:

Before activating the next iteration, the pheromone trails are evaporated by applying the usual rule on every state x_{ij} as in Equation (10) with $\rho \in (0,1]$ being the pheromone evaporation rate:

$$\tau_{ij}^{k}(t+1) = (1-\rho)\tau_{ij}^{k}(t) \tag{8}$$

The reason for allowing pheromone evaporation is to avoid too strong influence of the old pheromone to avoid premature solution stagnation then, every ant that generated a solution in the non-dominated front at the current iteration is allowed to update the pheromone matrices, $\tau^{k}, k = 1,2,...,K$, as in Equation (9):

$$\tau_{ij}^{k}(t+1) = \tau_{ij}^{k}(t) + \nabla\tau_{i}^{k}(t) \quad ,t = 0,1,...,T, k = 1,2,...,K. \tag{9}$$

where T is the number of iterations (generation cycles); τ_{ij} is the revised concentration of pheromone associated with option x_{ij} at iteration t- which means assignment of j amount of resource to the job i-and $\nabla\tau_{ij}$ is change in pheromone concentration which is calculated as in Equation (10)

$$\nabla\tau_{ij}^{k} = \sum_{a=1}^{m}\begin{cases} Q/f_{a}^{k}(\mathbf{x}_i) \quad \mathit{for\,Max} \;\; f_{a}^{k}(\mathbf{x}_i) \\ Q * f_{a}^{k}(\mathbf{x}_i) \quad \mathit{for\,Min} \;\; f_{a}^{k}(\mathbf{x}_i) \end{cases} \begin{matrix}\text{if the sate } \mathrm{x_{ij}} \text{ belongs to} \\ \text{non-dominated solution}\end{matrix} \\ \quad 0 \qquad\qquad\qquad\qquad\qquad\qquad \mathit{otherwise} \tag{10}$$

where Q is a constant called the pheromone reward factor; and $f_{a}^{k}(\mathbf{x}_i)$ is the value of the objective function (solution performance) calculated for ant a. It is noted that the amount of pheromone gets higher as the solution improves. Therefore, for minimization problems, Equation (10) shows the pheromone change as proportional to the inverse of the fitness. In maximization problems, on the other hand, the fitness value itself can be directly used.

3.6. Evolving ant position

Local search was introduced by Dorigo [6] which is known as daemon actions. Therefore in this step we introduce a new technique mechanism for local search scheme where the search carried around the found solution by all colonies after shuffling together. The mechanism start by choosing any node randomly (x_{ij}) then an increment ($d\mathbf{x}$) is calculated based on the lower and the upper bound of workers. The new solution must be feasible (i.e., satisfied the constraints (Ω)), therefore the ants move in new directions in search of newer and richer stocks of food sources. The pseudo code of the local search scheme is shown in Figure 4. The pseudo code of the proposed NM-ACO algorithm is given in Figure 5.

Input: choose x_{ij} randomly; number of maximum iteration (T).

Set $t = 0$

Generate $d\mathbf{x}$ ($d\mathbf{x} = 0.5*(M)*(0.9)^{h}$)

If $\exists x'_{ij} = \mathrm{int}(x'_{ij} + d\mathrm{x}) \,|\, x'_{ij} \in \Omega$,then $x_{ij}^{new} = x'_{ij}$, (int : mean take the integer number form calculation)

Else if $\exists x'_{ij} = \mathrm{int}(x'_{ij} - d\mathrm{x}) \,|\, x'_{ij} \in \Omega$,then $x_{ij}^{new} = x'_{ij}$

Else $x_{ij}^{new} = [\,]$ then $t = t+1$

End if

output : x_{ij}^{new}

Figure-4. the pseudo code of the local search scheme

Input:
Initialize the parameters for NM-ACO ($m, \alpha, \beta, K, \max iter.$).
Initialize the quantity of pheromone of all states $\tau_{ij}^{k}(0)$ for all colonies.
$X_k = \phi \quad , t = 0$

Repeat
while $T <= \max iter$ not completed ***do***
for $k = 1:K$ ***do***
$x = \phi$
for $i = 1:m$
for $j = 1:N$ ***do***
while solution not completed ***do***
$x_{ij} =$ Construct the solutions for all colonies according to the Equation (7).
$x_{ij} = x \cup x_{ij}$
$x = x_{ij}$
end while
end for
end for
$X_k = x$
$X = \{X_k\}$
end for
Shuffled the solutions of all colonies: $X = [X_1, X_2, ..., X_K]^T$
Calculate objective functions corresponded to the constructed solutions
Determine the non-dominated solutions**.**
Apply the local search scheme for all shuffled the solutions.
Calculate objective functions after applying the local search scheme
Determine the non-dominated solutions .
Update the pheromone quantity (Equations (8) , (9) and (10)).
$t = t + 1$
End

output : Pareto optimal solutions

Figure-5. The pseudo code of the proposed NM-ACO algorithm

4. EXPREMENTAL RESULTS

The proposed algorithm had been coded in MATLAB 7.2 and executed on 2.80 GHz Pentium. The proposed algorithm contains number of parameters. These parameters affect the performance of the proposed ant algorithm. Extensive experimental tests were conducted to see the effect of different values on the performance of the proposed algorithm. Based upon these observations, the following parameters are set as in Table 1.

Table-1: The parameter settings of NM-ACO

Parameters	Values
Number of objective functions (K)	2
Number of ants per colony (m)	50
Number of iterations (T)	5
ρ	0.5
α	1
β	2
Q	100
$\tau_{ij}^{k}(0)$	10

Consider multi-objective human resource allocation problem, which was extract from [5] and was solved with hybrid genetic algorithm, this problem is allocating two managers and their salesclerks of each district to new four shops of other districts for expanding the business extension. We can use the experimental example, and show the average profits and costs of the some company in the past 4 years [5].

Now, the company expands its business extension abroad. For example, the company wants to set up the subsidiary in Japan, Australia, Europe, China, and so forth. Supposing that we do not consider other special factors, the maximum benefit and minimum cost are obtained only from the past 4 years. How to make up and allocate the salesclerks (shop assistants) of every group or team to sell? The problem is that we allocate two managers and their salesclerks of each district to new 4 shops of other districts for expanding the business extension abroad. Table 2 provides the expected costs and Table 3 provides the expected profits. The following Tables 4 and 5 as shows an example of the nondominated (Pareto) solution by mohGA and proposed NM-ACO.

4.1 Performance assessment

To compare performances, we use the C- metric introduced in [22]. Using metric C two sets of nondominated solutions can be compared to each other. In this metric the size of the dominated area in the objective space is taken under consideration; thus the C-metric $C(A,B)$ calculates the proportion of solutions in B, which are weakly dominated by solutions of A as in the Equation (11):

$$C(A,B)=\frac{\left|\left\{b\in B \,\middle|\, a\in A : a\succeq b\right\}\right|}{|B|} \tag{11}$$

The function *C* maps the ordered pair (*A*, *B*) into the interval [0, 1].

The value *C* (*A*, *B*) = 1 means that all objective vectors in *B* are dominated by *A*.

1. The value *C* (*A*, *B*) = 1 means that all objective vectors in *B* are weakly dominated by *A*.
2. The value *C* (*A*, *B*) = 0 represent the situation when none of the points in *B* are dominated by *A*.
3. *C* (*A*, *B*) is not necessary equal to 1 – *C* (*B*, *A*).

Figure 6 shows that proposed NM-ACO dominates 34% of Pareto optimal front constructed by mohGA (Solutions 4, 12,13,15,16 and 17 of Table 4) and also none of Pareto optimal front generated by proposed NM-ACO was dominated by mohGA. This result can approve that proposed NM-ACO approximately outperform the mohGA in solving multi-objective resource allocation problem.

Table-2: Expected costs (c_{ij})

No. of district: i	Salesclerk: j												
	0	1	2	3	4	5	6	7	8	9	10	11	12
1	16	20	19	31	28	35	47	53	61	68	72	80	79
2	21	24	26	31	35	41	47	51	55	66	63	71	77
3	17	21	23	31	36	51	48	59	60	73	81	87	93
4	14	22	23	21	31	42	46	51	63	70	84	82	91

Table-3: Expected profits (p_{ij})

No. of district: i	Salesclerk: j												
	0	1	2	3	4	5	6	7	8	9	10	11	12
1	0	57	58	66	65	83	91	97	114	106	121	129	138
2	0	73	77	85	91	97	101	113	121	127	134	141	147
3	0	70	73	79	82	91	96	99	101	112	123	127	131
4	0	62	63	71	77	87	91	106	103	118	129	134	140

Table-5: The nondominated solutions of by the proposed NM-ACO

Solution k	jx_{1j}	jx_{2j}	jx_{3j}	jx_{4j}	$0\le j\le 48$	Overall cost	Overall profit
1	11	8	1	7	27	207	426
2	5	3	1	3	12	108	309
3	8	2	2	3	15	131	335
4	5	4	2	2	13	116	310
5	10	12	4	11	37	267	484
6	5	9	2	7	23	175	389
7	7	12	1	7	27	202	420
8	6	12	1	3	22	166	379
9	1	12	2	3	18	141	348
10	8	10	10	11	39	287	505
11	8	12	6	10	36	270	486

12	2	4	1	3	10	96	290
13	11	12	2	12	37	271	489
14	5	5	2	3	15	120	324
15	2	2	2	3	9	89	279
16	6	10	2	3	21	154	369
17	10	11	6	7	34	242	464
18	8	12	6	9	25	256	475
19	11	10	7	7	35	253	468
20	2	2	1	0	5	80	205
21	10	11	6	12	39	282	498
22	10	10	6	3	29	204	422
23	2	1	1	3	7	85	272
24	5	1	2	3	11	103	300
25	5	8	4	3	20	147	357
26	8	12	11	11	42	307	522
27	11	12	8	11	42	299	511
28	8	7	1	7	23	184	403
29	8	12	9	7	36	262	479
30	2	1	0	0	3	74	131
31	1	1	1	0	3	79	200
32	6	12	1	7	26	196	414
33	5	11	1	3	20	148	365
34	5	1	1	3	10	101	297
35	11	12	11	7	41	295	509
36	5	7	4	7	23	173	384
37	6	10	3	7	26	192	410
38	5	10	2	9	26	191	408
39	0	1	0	0	1	71	73
40	8	2	1	3	14	129	332
41	0	1	1	2	4	84	206
42	0	2	0	0	2	73	77
43	2	11	2	3	18	134	343
44	11	11	9	11	42	306	516

Table-4: The nondominated solutions of by mohGA

Solution k	jx_{1j}	jx_{2j}	jx_{3j}	jx_{4j}	$0 \le j \le 48$	Overall cost	Overall profit
1	0	1	0	1	2	325	544
2	0	1	3	0	4	314	531
3	0	2	4	2	8	286	502
4*	4	2	2	3	11	275	489
5	0	3	9	1	13	258	478
6	4	1	11	1	17	235	459
7	0	4	10	3	17	227	450
8	0	1	11	6	18	217	440
9	0	2	9	11	22	195	413
10	7	1	11	5	24	178	400
11	7	2	9	8	26	160	373
12*	11	4	10	5	30	149	357
13*	9	2	11	9	31	136	341
14	10	3	11	9	33	127	326
15*	6	10	11	8	35	125	315
16*	9	8	11	8	36	118	304
17*	8	11	11	9	39	94	279
18	11	11	10	9	41	88	274

* Indicate the dominated solutions

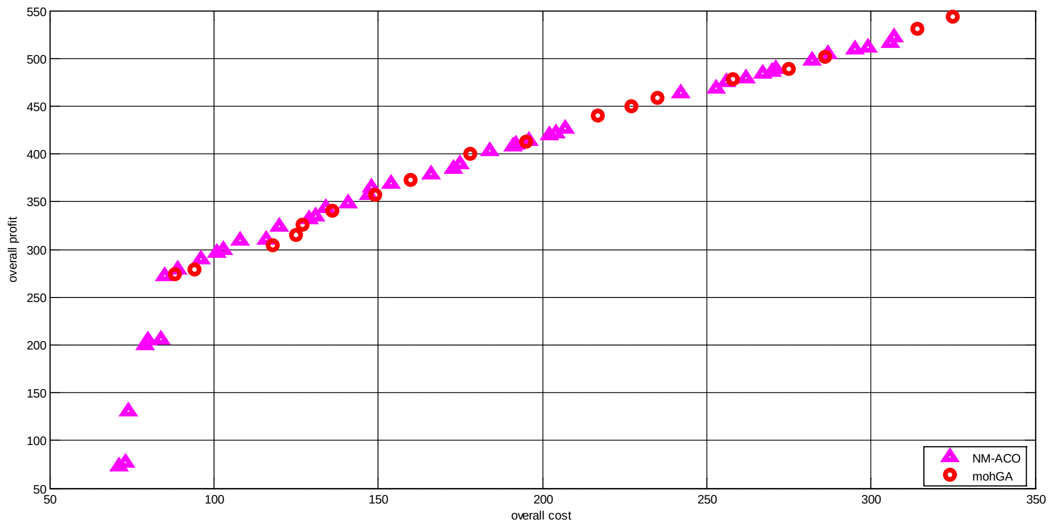

Figure-6. The simulation results by the proposed NM-ACO and mohGA

It can be shown from Figure 6 that the proposed NM-ACO has satisfactory diversity and distribution characteristics than mohGA for MORAP. The superiority of the proposed NM-ACO compared to mohGA is more pronounced in the MORAP as the proposed NM-ACO captures the true Pareto optimal front in this problem with satisfactory diversity and distribution.

In this section, a comparative study has been carried out to assess the proposed approach concerning large-size problem of the Pareto set, multistage decision-making model, and computational time. On the one hand, evolutionary techniques suffer from the large-size problem of the Pareto set. Therefore, the proposed approach has been used to reduce the Pareto set to a manageable size. However, the proposed approach capable to help an inexperienced decision maker to make correct decision to design problems by visualizing the Pareto front, which maintains the diversity of the solutions and good distribution over the non dominated front. On the other hand, classical techniques aim to give a single point (solution) at each iteration of problem solving. On the contrary, the proposed approach generates a set of solutions (Pareto set) at each iteration, and according to decision maker preference, one single point has been selected. Accordingly, it provides the facility to save computing time. Finally, the feasibility of using the proposed approach to handle multiobjective resource allocation problems has been empirically approved. The above discussions confirm that the proposed approach is better for the multiobjective resource allocation problems.

5. CONCLUSIONS

The multi-objective allocation problem addresses the important issue which seeks to find the expected objectives by allocating the limited amount of resource to various activates. The proposed algorithm is parameterized by the number of ant colonies and the number of pheromone trails. The proposed algorithm differs from the traditional ones in its design the vector of colonies associated with the vector of objective functions as well as inclusion of local search scheme which makes the ants move to new rich regions. Moreover, the proposed algorithm introduced an extended memory to store global Pareto solutions to reduce computational time. In the last Section of paper we compare the result of mohGA with proposed NM-ACO which were applied in the same problem, this comparing manifest that proposed NM-ACO outperform mohGA. The main features of the proposed algorithm could be summarized as follows.

(a) The proposed algorithm has been effectively applied to solve the MORAP, with no limitation in handing higher-dimensional problems.
(b) The proposed algorithm proves its versatility and robustness to tackle the resource allocation problems.
(c) The non-dominated solutions in the obtained Pareto optimal set are well distributed and have satisfactory diversity characteristics.
(d) The ability to find a solution by exploring the whole population and efficiently handle the problem with less computational time.
(e) Simulation results verified the validity and the advantages of the proposed approach.

We believe our model is likely to be recommended as a practical decision making tool for the management salesclerks of a firm. Thus, the boss of company can obtain a best decision maker, so that could obtain maximum profit and minimum cost for expanding the optimal business extension abroad.

In future, it would be interesting to see the application of other meta-heuristics or evolutionary optimization approaches like Genetic Algorithm, Tabu search, Simulated Annealing, etc. to tackle the balanced allocation problem. Ant colony optimization technique is increasingly getting attention from the research communities across the world to efficiently tackle NP-hard problems. Apart from the resouece allocation problem, it can also be applied in other location allocation problems, grouping problems, scheduling problems and other combinatorial optimization problems.

REFENCECS

1. U. Aybars, A. Dogan, An interactive simulation and analysis software for solving TSP using Ant Colony Optimization algorithms, Advances in Engineering Software 40 (2009) 341–349.
2. A. Basso, L.A. Peccati, Optimal resource allocation with minimum activation levels and fixed costs, European Journal of Operations Research 131 (2001) 536–549.
3. O. Berman, M. Cutler, Resource allocation during tests for optimally reliable software, Comput. Oper. Res. 31 (2004) 1847–1865.
4. Y.S. Dai, M. Xie, K.L. Poh, B. Yang, Optimal testing–resource allocation with genetic algorithm for modular software systems, J. Systems Software 66 (2003) 47–55.
5. M. L. Chi, G. Mitsuo, Multi-criteria human resource allocation for solving multistage combinatorial optimization problems using multiobjective hybrid genetic algorithm, Expert Systems with Applications 34 (2008) 2480–2490.
6. M. Dorigo, C. Blum, Ant colony optimization theory: a survey, Theoretical Computer Science 344 (2005), 243–278.
7. M. Dorigo, Optimization, learning, and natural algorithms, Ph.D. Thesis, Dip. Elettronica e Informazione, Politecnico di Milano, Italy, 1992.
8. K. Fan , Weijia You, Yuanyuan Li, An effective modified binary particle swarm optimization (mBPSO) algorithm for multi-objective resource allocation problem (MORAP), Applied Mathematics and Computation 221 (2013) 257–267.
9. F. Glover, Tabu search – Part I, ORSA Journal of Computing 1 (1989) 190–206.
10. D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Wesley, Reading, Massachusetts, 1997.
11. P. Hadi, T. M. Reza, Solving a multi-objective open shop scheduling problem by a novel hybrid ant colony optimization. Expert Systems with Applications 38 (2011) 2817–2822.
12. T. Ibaraki, N. Katoh, Resource Allocation Problems: Algorithmic Approaches, MIT Press, Boston, 1988.
13. M. Johannesson, M.C.Weinstein, On the decision rules of cost-effectiveness analysis, J. Health Econom. 12 (1993) 459–467.
14. D. Kalyanmoy, Multi-Objective Optimization using Evolutionary Algorithms, Wiley-Interscience Series in Systems and Optimization. John Wiley & Sons, Chichester, 2001.
15. S. Kirkpatrick, C. Gelatt Jr., M. Vecchi, Optimization by simulated annealing, Science 220 (1983) 671–680.
16. S.J. Shyu, P.Y. Yin, B.M.T. Lin, T.S. Hsiao, An ant colony optimization algorithm for the minimum weight vertex cover problem, Annals of Operations Research 131 (2004) 283–304.
17. A.A. Stinnett, A.D. Paltiel, Mathematical programming for the efficient allocation of health care resource, J. Health Econom. 15 (1996) 641–653.
18. Y. Yahong, W. Guiling, C. Jianping, D. Wei , Multi-objective optimization based on ant colony optimization in grid over optical burst switching networks, Expert Systems with Applications 37 (2010) 1769–1775.
19. P.Y. Yin, S.S. Yu, P.P. Wang, Y.T. Wang, A hybrid particle swarm optimization algorithm for optimal task assignment in distributed systems, Comput. Standards and Interfaces 28 (2006) 441–450.
20. P.Y. Yin, Ant colony search algorithm for optimal polygonal approximation of plane curves, Pattern Recognition 36 (2003) 1783– 1797.
21. H. Zhengxing, W.M.P. vander Aalst, L. Xudong, D.Huilong, Reinforcement learning based resource allocation in business process management, Data & Knowledge Engineering 70(2011)127-145.
22. E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: a comparative case study, the strength Pareto approach, IEEE Transactions on Evolutionary Computation 3 (1999) 257-271.

Source of support: Nil, Conflict of interest: None Declared