Appl. Math. Inf. Sci. **11**, No. 6, 1681-1692 (2017)

Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/110615

K-means-Clustering Based Evolutionary Algorithm for Multi-objective Resource Allocation Problems

*A. A. Mousa, M. A. El-Shorbagy** *and M. A. Farag*

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

Received: 24 Apr. 2015, Revised: 10 May 2017, Accepted: 14 May 2017

Abstract: The process of finding the optimal allocation of limited resources to a number of tasks for optimizing multiple objectives is called multi-objective resource allocation problem (MORAP). This paper presents K-means-clustering based on one of the evolutionary algorithm, genetic algorithm(GA), to solve MORAP. Using the K-means-clustering algorithm to divide the population to a specific number of sub-populations each of them with dynamic size. Therefore, different operators of GA (crossover&mutation) can be implemented to each subpopulation instead of applied the same GA operators to the all population. The aim of dynamic clustering is to preserve and introduce diversity into solutions, instead of the solutions becoming similar each other. Two problems taken from the literature are used to compare the performance of the proposed algorithm with the competing algorithms. Moreover, an example of optimum utilization of human resource in reclamation of derelict land in Toshka-Egypt is solved by our approach. The results of different test problems have showed the superiority of our algorithm to solve MORAP.

Keywords: Dalgaard-Strulik model, energy, economic growth, time delay, limit cycle

1 Introduction

Resource allocation is part of resource management. It is employed to allocate the available amount of resources economically. In management sciences, resource allocation is the scheduling of resources and activities while taking into consideration both the resource availability and the project time in an economic way [1]. MORAP is the procedure of allocating amount of resources among the various tasks in order to meet the expected objectives. Resources may be capital, man-power, raw materials or anything else in limited supply that can be used to achieve the objectives. The objectives may be minimizing costs, maximizing profits, or accomplishing the best possible quality or anything else [2]. MORAP has a variety of applications such as:

1. Assignment marketing resources [3]: the characteristics key of marketing managers is to take the decision to allocate scarce marketing resources, e.g., retail shelf-space or merchandise inventories, selling hours, advertising dollars.
2. Stochastic network systems [4]: for getting the optimum performance of the network by distributing both resource and duration of activities in proper direction.
3. Scheduling allocation of workers in the manual labor environment [5]: for assigning the workers into the jobs to minimize human cost, to reduce production duration and to control production overwork.
4. Portfolio optimization [6]: for making optimized portfolios on allocating funds to stocks or bonds to maximize profit for a given level of risk, or to reduce risk for a target rate of return.
5. Health care resource allocation [7]: Discounting costs and health benefits in cost-effectiveness analysis for the health care resource allocation.
6. Traffic accidents sanitary [8]: creating new protocols and applications to improve assistance in traffic accidents.
7. Energy resources allocation [9,10]: satisfying optimization model for sustainable gas resources allocation
8. Dynamic PERT networks [11]: developing a multi-criteria model for the resource allocation problem in a dynamic PERT network.

There are infinite variety of applications can be handled this way [12,13].

---

* Corresponding author e-mail: mohammed_shorbagy@yahoo.com

Traditionally, MORAP have been solved using methods in operations research [14-20].For instance, in [14] Zhao et al. proposed an integer programming approach for integrated resource assignment and job scheduling for a multiple job-agents-system. While, in [15] a branch-and-bound approach is presented for activity scheduling of a system of the PERT/CPM variety where the project duration is minimized. Furthermore, in [16] Lai and Li developed a new procedure based on dynamic programming for solving the multi-criteria resource allocation problem. In addition, Bretthauer and Shetty [17] presented a new algorithm to solve the resource allocation problem with nonlinearity, which is defined as the optimization of a convex function over a convex constraint subject to bounded integer variables. Firstly the authors introduce a pegging algorithm for continuous variable problem, and then implement the pegging method in a branch and bound algorithm for solving the integer variable problem. A multi-dimensional fairness procedure is proposed for auction-based multi-attribute resource allocation which faces the problem in a more holistic way, taking into account priorities throughout all of the auction process [20]. A comprehensive survey of the state of the art in resource allocation problem can be found in [13]. But none of these methods is computationally tractable for any real-life application size, thus rendering them impractical for real-life application [21].

For MORAPs, evolutionary algorithms (EAs) methods provide more realistic techniques for finding a solution. There has been an increasing interest in studying EAs for optimization problems due to its importance in real life applications. EAs are conceptually different from the traditional programming techniques; where these methods are incorporate certain biological, molecular, and neurological phenomena. The reasons for their popularity are as follows [22]: (1) EAs do not require any derivative information, (2) EAs are very simple to implement, (3) EAs are flexible with having a wide-spread applicability. Also, one of the basic advantages of EAs is that can be find a high quality solutions with reasonable computational times.

Recently, several EAs have been proposed to solve MORAPs including GA [23,24,25,26], simulated annealing (SA) [27], ant colony optimization (ACO) [28, 29], particle swarm optimization (PSO) [30,31,32], variable neighborhood search [33] and cuckoo search algorithm (CSA) [34]. In [24], the authors proposed a multi-objective hybrid GA approach based on the multistage decision making model to get efficiently a set of non-dominated solutions. In [26], a two-stage procedure is proposed to MORAP: firstly, NSGA-II is applied to obtain the non-dominated solution set; secondly, entropy weight and TOPSIS (technique for order performance by similarity to ideal solution) method are implemented to determine the best compromise solution. In [27], a hybrid SA technique is presented to a complex energy resource management problem including a large number of resources. In [29], a modified version of ACO is proposed in order to get a set of non-dominated solution; where the algorithm efficiency is increased by increasing the ants learning. In [31], a hybrid PSO is proposed; where a hill-climbing heuristic is embedded into the PSO for speeding the convergence. Also, in [32], the authors proposed a modified binary PSO algorithm for solving the MORAP. In [33], variable neighborhood search is used to solve the MORAP with two important but conflicting objectives: maximization of efficiency and minimization of cost. Finally, in [34] a penalty based CSA is proposed to find the optimal solution of reliability allocation problems.

Clustering is the process of dividing the data to groups that has similar objects. Each group is called cluster contains the objects that are similar between themselves in the same cluster, and different to the objects in the other cluters [35]. There are many clustering techniques [36]. The K-means-clustering is the most commonly used clustering algorithm due to its simplicity and accuracy [37]. Several clustering algorithms have been proposed such as: iterative self-organizing data analysis technique [38], clustering large applications based up on randomized search [39], parallel-cluster [40], density-based spatial clustering of applications with noise [41] and balanced iterative reducing and clustering using hierarchies [42].

This paper presents K-means-clustering based on genetic algorithm (GA) to solve MORAP. K-means-clustering algorithm divided the population of GA to a specific number of sub-populations each of them with a dynamic size. So, different GA operators can be implemented to each subpopulation instead of applied the same GA operators to the all population. In addition, the dynamic clustering is preserve and introduce diversity into solutions, instead of the solutions becoming similar each other. Test problems are used to compare the performance of the proposed algorithm with the competing algorithms. The results of different test problems have demonstrate the superiority of our algorithm to solve MORAP.

This paper is structured as follows: section 2 describes the formulation of the multi-objective resource allocation problem (MORAP). In section 3, genetic algorithm procedures is displayed. In section 4, clustering technique is introduced. The proposed approach is explained in details in section 5. While, numerical simulation is presented in section 6. Finally, conclusion of the paper is presented in section 7.

2 Mathematical formulation of MORAP

The general form of the MORAP is as follows [23]:

$$
\begin{aligned}
&Max \quad Z_1(x_1,x_2,\ldots,x_n) = \sum_{k=1}^{n} z_k^1(x_k)\\
&Max \quad Z_2(x_1,x_2,\ldots,x_n) = \sum_{k=1}^{n} z_k^2(x_k)\\
&\quad\ldots\ldots\ldots\ldots\\
&Max \quad Z_q(x_1,x_2,\ldots,x_n) = \sum_{k=1}^{n} z_k^q(x_k)\\
&\text{Subject to}\ : \quad \sum_{k=1}^{n} g_k(x_k) \le S,\\
&\qquad\qquad g_k(x_k) \ge 0,\\
&\qquad\qquad x_k \ge 0;
\end{aligned} \tag{1}
$$

wherex_k define (decision variables), S donates the (resources), $g_k(x_k)$ represents the activities stages and $Z_q(x_1,x_2,\ldots,x_n)$ is the q-objective functions. The process of MORAP seeks to optimize the q-objective functions which subjected to some constraints of resource and finds an optimal allocation (decision variables) of limited amount of resource to a number of tasks (activities stages) [23].

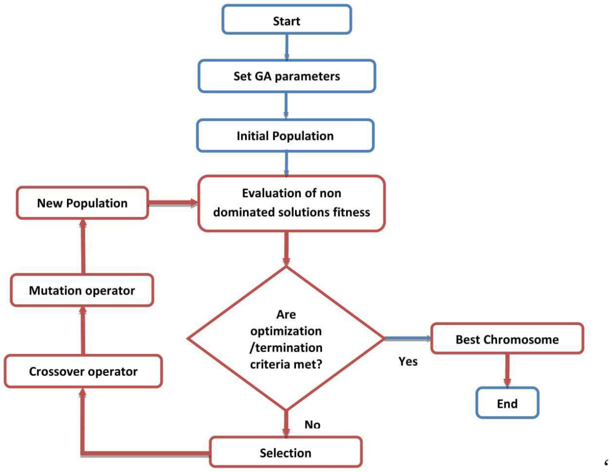


Fig. 1: Main flowchart of GA

3 GA procedures

GAs operateson a population of candidate solutions called chromosome. To obtain optimality, all chromosomes (solutions) exchanges information with each other by using operators Inspired from natural genetic to produce an improved solution [43,44]. The steps of GA is described below [24]:
Step 0: Generate randomly the initial population.
Step 1: Evaluate the values of the q-objective functions for every chromosome.
Step 2: Select the best chromosomes (parents).
Step 3: Crossover is implemented on every pair selected of chromosomes (parents) to generate two offspring.
Step 4: Mutation is applied to every two offspring obtained after the crossover.
Step 5: Check the stopping criterion, if it is not reached, go to Step 1, else, continue.
Step 6: The most preferred solutions among the alternatives are selected by the DM.

The basic step of GA is illustrated in flowchart as shown in figure 1 (taken from [45]).

4 Clustering Technique

Clustering is the process to find groups of objects (clusters); where the objects in any group is similar to each other and not-similar to the objects in other groups [35]. K-means-clustering approach [37] is one of the simplest algorithms that using to solve clustering problems. The procedures of K-means-clustering are simple and easy to classify a given data set to a certain number of clusters (k); which is already known. The main idea of K-means-clustering is to define a centroid k to each cluster. The better choice, for the centroids, is to place them as far as possible be spaced out. The next step is to associate each point in the given data set with the nearest centroid. The first step is completed when no point is pending i.e. an early groups was generated. Then, k new centroids are re-calculated as centers of the resulting clusters. Every point in the same data set is associated with the nearest new centroid. These steps are repeated i.e. the location of the centroids is changed step by step until it do not move any more. The coordinates of the centroid point is defined as the average of the coordinates of the points in the cluster. The K-means-clustering algorithm can be described by the following steps [46]:

Step 1: the number of desired clusters (k) is defined. Step 2: the initial centroid of the cluster is chosen randomly.

Step 3: the sum of square error (the squared Euclidean distance) between each object and the centers of clusters is computed as follows:

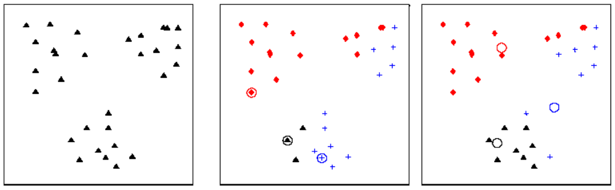
$$\text{Sum of squre error}\ (SSE) = \sum_{i=1}^{k}\sum_{x\in C_i} dist^2(m_i,x) \tag{2}$$

where x is a point in cluster C_i and m_i is the centroid of cluster C.

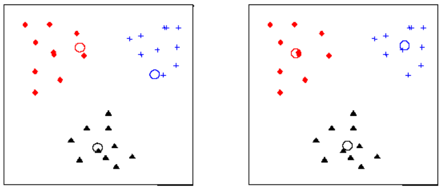
Step 4: each object is assigned to the nearest cluster.

Step 5: for each cluster, the new centroid is computed.

Step 6: steps 3 to 5 are repeated until the location of the centroids are not change. Figure 2 illustrates the steps of the K-means procedure on dataset with three clusters in R^2 [47].



(a) Input data (b) Seed point selection (c) Iteration 2



(d) Iteration 3 (e) Final clustering

Fig. 2: K-means procedure on dataset with three clusters in R^2.

5 K-means-Clustering Based Genetic Algorithm

In this paper, K-means-clustering based GA is proposed to solve MORAP; where the population of GA is divided to a specific number of sub-populations by using K-means-clustering technique. Each sub-population has common features. In addition, different GA operators are implemented to each sub-population instead of applying the same GA operator to all population. The steps of K-means-clustering based GA are described below:

5.1 Initialization

MORAP is reformulated as network model as in figure 3.It is considered as multi-objective; where it aims to maximize the total efficiency of the human resource allocation problem and minimize the total cost. It is required to obtain the path between the two nodes source node (s) and terminal node (t) that has the minimum total cost and maximum efficiency [23].The path from S to T is a sequence of paths (s,x_{1m}), (x_{1m},x_{2m}), $\cdots$ (x_{N-1m},t). A path can be described as a sequence of nodes $(s,x_{1m},x_{2m},\cdots,x_{N-1m},t)$; where $m=0,1,2,\cdots,m$ and N is the number of stages. So, each chromosome is initialized as sequence of nodes (path).

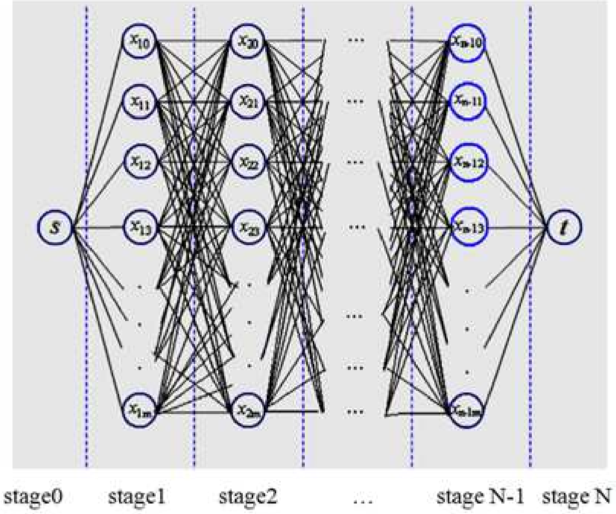


Fig. 3: Representation MORAP as network model

In this step, the population is initialized by generating N_{pop} strings; where the algorithm chose a random element from available number in each stage. For example, in the MORAP to allocate 10 workers in a set of 4 jobs (4 stages) as in figure 4. MORAP aims to determine a path in the 11 states (0→10 workers) and the 4 stages to achieve the minimum costs and the maximum efficiency. Figure 5 shows the structure of the chromosome for 4 stages allocation path.

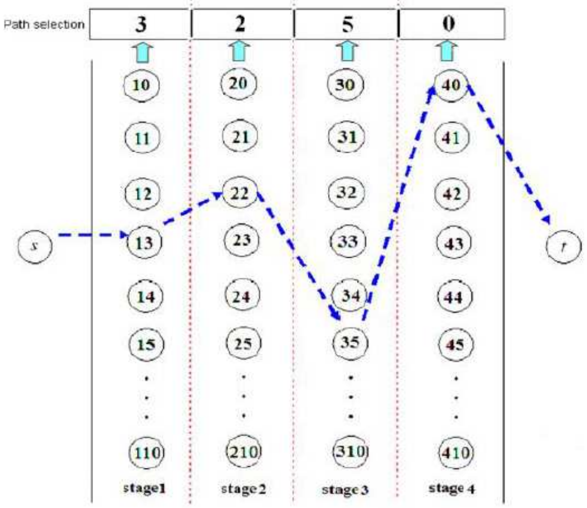
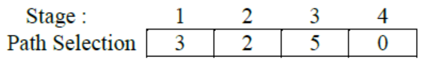


Fig. 4: Allocation path of 10 workers (states) in 4 jobs (stages) to minimize costs and maximize efficiency



Stage :	1	2	3	4
Path Selection	3	2	5	0

Fig. 5: The structure of the chromosome for 4-stages allocation path

5.2 Rejection of illegal individuals

The initialization step may be produced illegal individuals, which do not satisfy the problem feasible region, i.e., if there are N workers needed to allocate in M jobs, the number of worker must equal to N [23]. So, we should delete these individuals and replace them by generating legal individuals randomly.

5.3 Evaluation of non-dominated solutions

In multi-objective optimization problem (MOOP), there is no single solution optimizes the all objectives. But, there exists a number of solutions (may be infinite). These solutions are called non-dominated solutions, Pareto optimal solutions, Pareto efficient or noninferior [48]. So, to find these solutions, the population must be classified according to the definition of Pareto optimal solution.

Definition (Pareto optimal solution): the solution x^* is said to be a Pareto optimal solution for MOOP if there is no other feasible solution x such that: $f_j(x) \leq f_j(x^*)$ for all $j = 1,2,...,q$ and $f_j(x) < f_j(x^*)$ for at least one objective function f_j.

The image of the Pareto set (Pareto solutions) is called Pareto front. A population can be evaluated according to non-domination criteria. Consider two solutions x_1 and x_2 for a problem having more than one objective function $q\,(q > 1)$. These solutions may have one of these two possibilities: one of these solutions can be dominates the other or non-dominates the other. The solution x_1 is said to dominate other solution x_2 if the two following condition are achieved (say the operator $\prec$ denotes worse and $\succ$ denotes better) [49]:

1. $f_q(x_1) \not\prec f_q(x_2)$ for all $q = 1, \ldots, q$ objectives.
2. $f_q(x_1) \succ f_q(x_2)$ for at least one $q \in \{1,2, \ldots, q\}$.

The following procedures is used to find the non-dominated set of solutions from a set of population each having q objective function values $(q > 1)$ [50].

Step 0: Begin with $m = 1$.

Step 1: For all $n = 1,2,\cdots,N_{POP}$ and $m \neq n$, compare solutions x_m and x_n for domination using the previous two conditions for all q objectives.

Step 2: If for any n, x_m is dominated by x_n, mark x_m as ‘dominated’, and it is inefficient.

Step 3: If $m = N_{POP}$, go to Step 4, else $m = m+1$and go to Step 1.

Step 4: All set of solutions that are not marked by ‘dominated’ are the set of non-dominated solutions. All non-dominated solutions are constitute the Pareto (non-dominated) front on the population in a specific generation.

5.4 Selection

The aim of selection (parent selection stage) is to determine the best individuals that are suited to be parent and pass their chromosomes to the next generation. By other words, Better the fitness, the bigger chance to be selected [45]. It is intended to improve the quality of the population; where it gives the high quality individuals a better chance to copy in the next generation. Also, it directs the search of GA in the direction of promising regions in the search domain.

In this paper, the multiple objective functions is combined into a scalar fitness solution as [51]:

$$f(x) = w_1\, f_1(x) + \ldots + w_i\, f_i(x) + \ldots + w_q\, f_q(x); \qquad (3)$$

where x is an individual, $f_i(x)$ is the *ith* objective function and w_i is a weighting-vector for $f_i(x)$ such that: $w_i \geq 0$ for all $i = 1, \ldots, q$ and $\sum_{i=1}^{q} w_i = 1$. In general, each weight can be determine randomly. For a MOOP with q objective functions, we can assign a random real number to each weight as follows:

$$w_i = \frac{rand_i}{\sum_{i=1}^{q} rand_i}, \qquad i = 1,2,...,q; \qquad (4)$$

where$rand_i \;\forall\, i = 1,...,q$ are non-negative random real numbers.

Finally, binary tournament selection [52] is used; where two chromosomes are chosen randomly and the better of the two chromosomes according to the scalar fitness is copied in the mating pool

5.5 K-means-Clustering Technique

In this step, as shown in figure 6, K separated sub-populations with dynamic size is generated by dividing the population using K-means-Clustering Technique.

5.6 Genetic Algorithm (GA) Operators

In this step, different GA operators are applied to sub-populations instead of one GA operator applied to the all population; where the population is divided to many of sub-populations with a dynamic size.

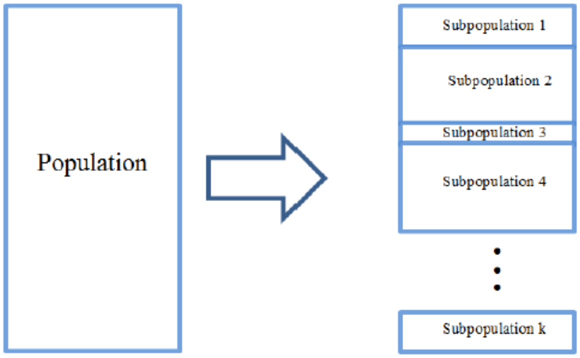


Fig. 6: Division of population into K sub-populations with dynamic size

5.6.1 Crossover operator

The crossover aims to exchange the information between two individuals to produce two new offspring [53]. In our study, we used some of crossover techniques; which are explained briefly below.

A. One-point crossover

In one-point-crossover operator, a crossover point is selected randomly within an individual then the two parent individuals at this point are interchanged to produce two new offspring [54].

B. Two-point crossover

In this crossover, two different cut-off points were randomly selected. New offspring were obtained by transmitting the zones, in the two parents, between the two cut points [54].

C. Uniform crossover

In this method, a random binary string is generated with the same-size of chromosome. Then relative genes under this binary string between parents is exchanged, where parent strings exchange their bit at the position where the corresponding position in random binary string is 1. Otherwise, no exchange of bit is performed [55].

D. Cross crossover

In such crossover, the new offspring are selected from different parts of parent. Often the produced generation are very different from their parent. A cut point is selected in the middle point in the two parent. The left side gene of the cut point of the first parent are copied directly to right side of the new first offspring. While, the remaining portion of offspring array are selected from left side gene of the cut point of the second parent. In order to produce the second offspring the parents are swapped in this process as in this pervious manner [56].

5.6.2 Mutation operator

Mutation operator is applied to all offspring in the new generation and used to modify some features of chromosomes depending on a predetermined small probability value to produce new chromosomes [53]. In the following, a brief explanation of mutation techniques that used in our study:

A. Twors mutation

In Twors mutation [57], two genes are randomly chosen to exchange of their position, as shown in figure 7.

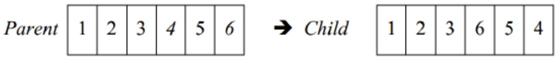


Fig. 7: Twors mutation

B. One point mutation

In one point mutation, data at a particular point is mutated (particular gene was randomly selected and then it was replaced with a random state from the available set) [57].

C. Reverse Sequence Mutation

In this mutation, a sequence S is taken; which is bounded by two randomly chosen positions i and j, such that $i < j$. Then, The arrange of the gene in this sequence is reversed by the same method as in the previous operation. Figure 8 shows the implementation of this mutation operator.

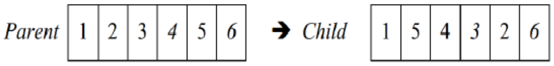


Fig. 8: Reverse Sequence Mutation

D. Centre inverse mutation

As shown in Figure 9, this mutation divided the parent into two sections. All genes in every section are copied and then inversely placed in the same section of a child [57].

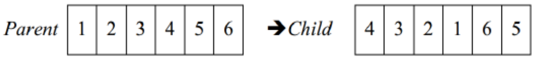


Fig. 9: Centre inverse mutation

5.7 Combination stage

In this stage, to create a new population, all sub-populations are combined together, as shown in Figure 10.

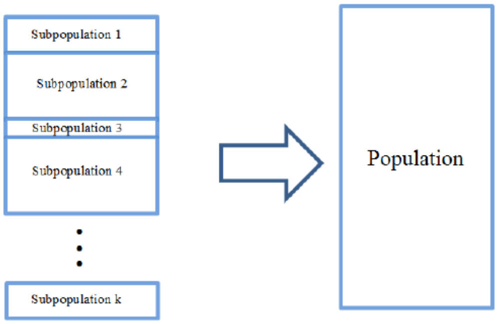


Fig. 10: Combination stage

5.8 Archive Update (Update the archive of non-dominated solution).

The algorithm has an external archive of non-dominated solutions which is updated iteratively based on the concept of non-domination in the state of presence new solutions. The main objective of this archive is to store a historical record of the non-dominated solutions found through the process of search [58]. During the process of search, a set of non-dominated solutions is added to the archive which updated iteratively every generation. Algorithm 1 show the procedure which is used to update the archive [59]. The main idea of Algorithm 1 is to generate a new archive set $A^{(k)}$ in each iteration k, using the contents of the old archive set $A^{(k-1)}$ and the current population $P^{(t)}$.

Algorithm 1: Update the archive of non-dominated solution

```
$k \triangleq 0$
$A^{(0)} = 0$
While terminate $(A^{(k)}, k) = false$  do
    $k \triangleq k+1$
    $p^{(k)} \triangleq generate(\,)$        {generate new search point}
    $A^{(k)} \triangleq update\ (A^{(k-1)}, p^{(k)})$    {update archive }
End while
Output : $A^{(k)}$
```

6 Experimental results

In this section, the proposed algorithm is evaluated by two multi-objective resource allocation test problems taken from the literature [23,24]. Moreover, the proposed algorithm is applied to an engineering application [60] to verify the performance of it for solving MORAPs and illustrate its ability to handle engineering problems. All test problems have been solved on an Intel core I5, 2.6 GHz processor. The proposed approach is coded using MATLAB programming language. The parameter setting used for all runs are depicted in Table 1.

Table 1:parameters setting of the proposed algorithm

Parameter	**Values**		
	Prolem1	**Prolem1**	**Application**
Population size	100	500	500
Crossover rate	0.65	0.95	0.95
Mutation rate	0.02	0.02	0.02
Iteration	30	200	1000
Number of cluster (k)	k =1 or k=4		

6.1 Test problem 1

Test problem 1 [23] of allocate 6 workers in a set of 4 jobs is tested by our approach. Table 2 provides the expected cost and efficiency.

Table 2: The expected cost and efficiency of test problem 1

Number of worker	JOB							
	1		2		3		4	
	Cost	Efficiency	Cost	Efficiency	Cost	Efficiency	Cost	Efficiency
0	70	0	90	0	85	0	130	0
1	60	25	60	20	60	33	115	13
2	50	42	50	38	50	43	100	24
3	40	55	40	54	55	47	100	32
4	40	63	30	65	40	50	90	39
5	45	69	20	73	30	52	80	45
6	50	74	25	80	25	53	80	50

Tables 3 and 4 present the efficient solution obtained by our algorithm at k=1 and k=4, respectively. While, table 5 gives the results obtained by other approach (the effective genetic algorithm [23]). Furthermore, figure 11 shows the efficient solution by our algorithm at k=1, k=4 and the effective genetic algorithm [23].

Table 3: Efficient solution obtained at k=1

Efficient solution				Overall Cost	Overall efficiency
X_1	X_2	X_3	X_4		
1	2	1	2	270	120
2	2	1	1	275	126
2	3	1	0	280	129

Table 4: Efficient solution obtained at k=4

Efficient solution				Overall Cost	Overall efficiency
X_1	X_2	X_3	X_4		
1	2	1	2	270	120
2	2	1	1	275	126
2	3	1	0	280	129

Table 5: Efficient solution of the effective genetic algorithm approach [23]

Efficient solution				Overall Cost	Overall efficiency
X_1	X_2	X_3	X_4		
1	2	1	2	270	120
2	2	1	1	275	126
2	3	1	0	280	129

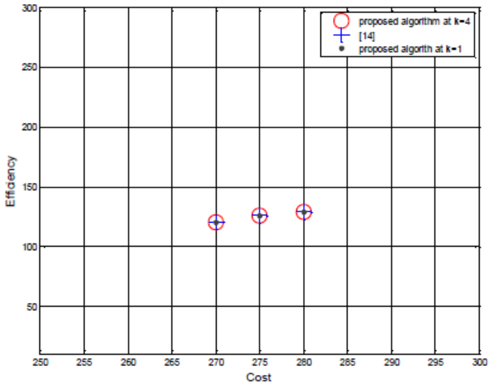


Fig. 11: Efficient solution of test problem 1 obtained by the proposed algorithm at k=1, k=4and the effective genetic algorithm approach [23].

It is clear from previous comparison in test problem 1 that the results obtained by the introduced algorithm at k=1, k=4 and the results obtained by effective genetic algorithm is the same values.

6.2 Test problem 2

Test problem 2 24 of allocate 10 workers in a set of 4 jobs is solved by our approach. Table 6 provides the expected cost and efficiency. Table 7 and table 8 presents the efficient solutions obtained by the introduced algorithm at k=1 and k=4, respectively. While, Table 9 presents the results obtained by multistage decision-based GA [24]. Furthermore, figure 12 shows the Pareto results of the proposed algorithm at different values of k and multistage decision-based GA [24].

Table 6: The expected cost and efficiency of test problem 2

Number of worker	JOB							
	1		2		3		4	
	Cost	Efficiency	Cost	Efficiency	Cost	Efficiency	Cost	Efficiency
0	41	0	45	0	36	0	46	0
1	38	37	54	49	43	45	78	60
2	46	42	36	55	68	49	88	67
3	32	50	55	59	56	57	64	72
4	78	54	87	62	72	64	90	79
5	76	56	82	67	59	77	80	83
6	72	58	90	73	32	88	120	88
7	84	65	132	80	67	92	104	97
8	80	72	97	87	86	100	96	102
9	92	80	21	95	188	105	86	110
10	96	95	134	102	100	110	120	120

Table 7: Efficient solutions of the proposed algorithm at k=1

Efficient solution				Overall Cost	Overall Efficiency
X_1	X_2	X_3	X_4		
2	3	1	0	190	146
3	1	1	0	175	144
3	3	0	1	201	169
3	3	0	0	169	109
3	1	1	1	207	204
3	0	0	0	159	50
3	3	1	1	208	214
3	1	0	0	168	99
3	1	0	1	200	159
3	0	1	1	198	155

Table 8: Efficient solutions of the proposed algorithm at k=4

Efficient solution				Overall Cost	Overall Efficiency
X_1	X_2	X_3	X_4		
2	3	1	0	190	146
3	1	1	0	175	144
3	3	0	1	201	169
3	3	0	0	169	109
3	1	1	1	207	204
3	0	0	0	159	50
3	3	1	1	208	214
3	1	0	0	168	99
3	1	0	1	200	159
3	0	1	1	198	155

Table 9: Efficient solutions of the multistage decision-based genetic algorithm [24]

Efficient solution				Overall Cost	Overall Efficiency
X_1	X_2	X_3	X_4		
2	3	1	0	190	146
3	1	1	0	175	144
3	3	0	1	201	169
3	3	0	0	169	109
3	1	1	1	207	204
3	0	0	0	159	50
3	3	1	1	208	214
3	1	0	0	168	99
3	1	0	1	200	159
3	0	1	1	198	155

As shown in figure 12, the results obtained by proposed algorithm with k=4 dominate the most of that obtained by multistage decision-based GA. In addition, the result obtained by the proposed algorithm with clustering technique (i.e. k=4) dominate the results obtained by the algorithm without clustering data (i.e., k=1). So, the proposed algorithm is more appropriate for using to solve MORAPs.

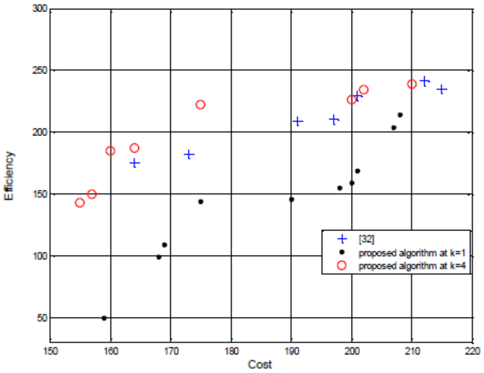


Fig. 12: Simulation results of test problem 2 obtained by our algorithm at k=1, k=4 and the multistage decision-based genetic algorithm [24].

6.3 Application

In this subsection, the proposed approach is applied to solve an application of reclamation of derelict land in Toshka-Egypt [60]. The aim is to illustrate the ability of proposed approach to solve engineering applications. In this application 4 stages is considered; which affect the optimization of the agriculture reclamation, including land settlement, land planning, digging of canals and plant cultivation. Also, two objectives are considered: maximizing benefit of reclamation and minimizing the costs of agriculture reclamation. The annually data of 10 staff are given in Table 10, where for each staff the minimum cost and maximum profit for the4 stages are provided. Figure 13 shows the Pareto solutions obtained by proposed approach at k=1, k=4 and that obtained by multi-objective multipheromone ant colony optimization approach [60].

Table 10: The average profit and cost of staff needed for reclamation

Staff Index	Land settlement		Land planning		Digging of canals		Plant cultivation	
	Cost	Profit	Cost	Profit	Cost	Profit	Cost	Profit
1	790	138	770	147	930	131	910	140
2	800	129	710	141	870	127	820	134
3	720	121	630	134	810	123	840	129
4	680	106	660	127	730	112	700	118
5	610	114	550	121	600	101	630	103
6	530	97	510	113	590	99	510	106
7	470	91	470	101	480	96	460	91
8	350	83	410	97	510	91	420	87
9	280	65	650	91	360	82	310	77
10	310	66	310	85	310	79	210	71

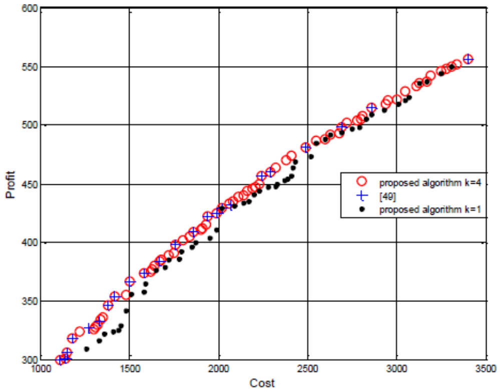


Fig. 13: Simulation results of test problem 2 obtained by our algorithm at k=1, k=4 and the multistage decision-based genetic algorithm [24].

It is clear that the Pareto solutions set obtained by multi-objective multipheromone ant colony optimization approach is part of the Pareto solutions set obtained by the algorithm with clustering data (i.e. k=4). In addition, the result obtained by the proposed algorithm with clustering technique (i.e. k=4) is dominant to the results obtained by the algorithm without clustering data (i.e. k=1).As shown in figure 13 the proposed algorithm outperformed the multi-objective multipheromone ant colony optimization approach in both distribution and spread and able to find points that the other method failed to reach. So, we can say that the proposed algorithm is a suitable method to solve to solve real life MORAPs.

7 Conclusion

This paper presents K-means-clustering approach based on one of the evolutionary algorithm, genetic algorithm (GA), to solve MORAP. K-means-clustering algorithm is used to divide the population to a specific number of sub-populations, each of them with a dynamic size. Therefore, we can implement different operators of GA (crossover and mutation) to each subpopulation instead of applied the same operator to the all population. The results of different test problems have showed the superiority of our algorithm to solve MORAP.Finally, the proposed algorithm has the following benefits :

1. It integrates the powerful searching of GA anddiversity of dynamic clustering.
2. Incorporating the evolutionary algorithm with dynamic clustering, preserves the diversity of the solutions and prevents it to be similar with each other
3. The results of different test problems have showed the superiority of it to solve MORAP
4. Due to its simplicity, it was demonstrated to be a good tool to solve MORAPs.

In our future works, the following will be researched:

1. Solving larger scale examples to demonstrate the efficiency of our approach.
2. Updating our approach to can be applied to solve many applications of MORAP.
3. Using other clustering techniques to accelerate the convergence property of the proposed approach and improve the solution quality.

References

[1] J.L. Ponz-Tienda, V. Yepes, E. Pellicer, and J. Moreno-Flores, The Resource Leveling Problem with multiple resources using an adaptive genetic algorithm, Automation in Construction 29 (2013) 161-172.

[2] J.C. Whittaker, C. Cannings, A Resource Allocation Problem, Journal of Theoretical Biology, 167(4) (1994) 397-405.

[3] G.L. Lilien, A. Rangaswamy, G.H.V. Bruggen, and K. Stark, DSS Effectiveness in Marketing Resource Allocation Decisions: Reality vs. Perception, Information Systems Research 15(3) (2004) 216-235.

[4] X. Wang, and G.B. Ginannakies, Resource Allocation for Wireless Multiuser OFDM Networks, IEEE 57(7) (2011) 4359-4372.

[5] S. Tan, W. Weng, and S. Fujimura, Scheduling of Worker Allocation in the Manual Labor Environment with Genetic Algorithm, International Multi Conference of Engineers and Computer Scientists (IMECS) I (2009).

[6] M. Ehrgott, K. Klamroth, and C. Schwehm, An MCDM approach to portfolio optimization, European Journal of Operational Research 155(3) (2004) 752-770.

[7] K. Claxton, M. Paulden, H. Gravelle, W. Brouwer and A.J. Culyer, Discounting and decision making in the economic evaluation of health-care technologies, Health Economics 20(1) (2011) 2-15.

[8] M. Fogue, P. Garrido, F.J. Martinez, J.-C. Cano, C.T. Calafate, and P. Manzoni, A novel approach for traffic accidents sanitary resource allocation based on multi-objective genetic algorithms, Expert Systems with Applications 40 (2013) 323-336

[9] R. Alikhani, and A. Azar, A hybrid fuzzy satisfying optimization model for sustainable gas resources allocation, Journal of Cleaner Production 107 (2015) 353-365.

[10] A. Kazemi, M.R. Mehregana, G.H. Shakouri, and M. Hosseinzadeh, Energy Resource Allocation in Iran: A Fuzzy Multi-Objective Analysis, Procedia - Social and Behavioral Sciences 41 (2012) 334-341.

[11] A. Azaron, and R. T.-Moghaddam, A multi-objective resource allocation problem in dynamic PERT networks, Applied Mathematics and Computation 181 (2006) 163-174.

[12] T. Ibaraki, and N. Katoh, Resource Allocation Problems: Algorithmic Approaches,MIT Press, Boston , USA, 1988.

[13] M. Patriksson, and C. Strömberg, Algorithms for the continuous nonlinear resource allocation problem—New implementations and numerical studies, European Journal of Operational Research 243 (2015) 703-722.

[14] Z.J. Zhao, H.C. Lau, and S.S. Ge, Integrated Resource Allocation and Scheduling in a Bidirectional Flowshop with Multimachine and COS Constraints, Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews 39(2) (2009) 190-200.

[15] E. Demeulemeester, and W. Herroelen, A branch-and-bound procedure for the multiple resource constrained project scheduling problem, Management Science 38 (12) (1992) 1803-1818.

[16] K.K. Lai, and L. Li, A dynamic approach to multiple-objective resource allocation problem European Journal of Operational Research 117 (1999) 293-309.

[17] K.M. Bretthauer, and B. Shetty, A pegging algorithm for the nonlinear resource allocation problem, Computers & Operations Research 29 (2002) 505-527.

[18] V. Chankong, and Y.Y. Haimes, Multiobjective Decision Making Theory and Methodology, Series in System Science and Engineering, North-Holland, Amsterdam, 1983.

[19] S.E. Wrighta, and J.J. Rohal, Solving the continuous nonlinear resource allocation problem with an interior point method, Operations Research Letters 42 (2014) 404–408

[20] A. Pla, B. Lpez, and J. Murillo, Multi-dimensional fairness for auction-based resource allocation, Knowledge-Based Systems 73 (2015) 134-148.

[21] O. Moselhi, and P. Lorterapong, Least impact algorithm for resource allocation, Canadian Journal of Civil Engineering 20 (2) (1993) 180-188.

[22] M.A. El-Shorbagy, A.A. Mousa, and S.M. Nasr, A chaos-based evolutionary algorithm for general nonlinear programming problems, Chaos, Solitons and Fractals 85 (2016) 8-21.

[23] M.S. Osman, M.A. Abo-Sinna, and A.A. Mousa, An effective genetic algorithm approach to multi-objective resource allocation problems, Applied Mathematics and Computation 163 (2005) 755-68.

[24] C.-M. Lin, and M. Gen, Multiobjective resource allocation problem by multistage decision- based hybrid genetic algorithm, Applied Mathematics and Computation 187 (2007) 574-583.

[25] K.J. Wang, and Y.S. Lin, Resource allocation by genetic algorithm with fuzzy inference, Expert Systems with Applications 33 (2007) 1025-1035.

[26] S. Ghasemi-Falavarjani, M. Nematbakhsh, and B.S. Ghahfarokhi, Context-aware multi-objective resource allocation in mobile cloud. Computers & Electrical Engineering 44 (2015) 218-240.

[27] T. Sousa, Z. Vale, J.P. Carvalho, T. Pinto, and H. Morais, A hybrid simulated annealing approach to handle energy resource management considering an intensive use of electric vehicles, Energy 67 (2014) 81-96.

[28] P. C. Pendharkar, An ant colony optimization heuristic for constrained task allocation problem, Journal of Computational Science 7 (2015) 37-47.

[29] S.K. Chaharsooghi, and A.H.M. Kermani, An effective ant colony optimization algorithm(ACO) for multi-objective resource allocation problem (MORAP), Applied Mathematics and Computation 200 (2008) 167-177.

[30] Q.Jia, and Y. Seo, An improved particle swarm optimization for the resource-constrained project scheduling problem, The International Journal of Advanced Manufacturing Technology 67 (2013) 2627-2638.

[31] P.-Y. Yin, and J.-Y. Wang, Optimal multiple-objective resource allocation using hybrid particle swarm optimization and adaptive resource bounds technique, Journal of Computational and Applied Mathematics 216 (2008) 73-86.

[32] K. Fan, W. You, and Y. 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.

[33] Y.-C. Liang, and C.-Y. Chuang, Variable neighborhood search for multi-objective resource allocation problems, Robotics and Computer-Integrated Manufacturing 29 (2013) 73-78.

[34] H. Garg, An approach for solving constrained reliability-redundancy allocation problems using cuckoo search algorithm, Beni-Suef University Journal of Basic and Applied Sciences 4(1) (2015) 14-25.

[35] M. Verma, M. Srivastava, N. Chack, A.K.Diswar, and N. Gupta, A Comparative Study of Various Clustering Algorithms in Data Mining, International Journal of Engineering Reserch and Applications (IJERA) 2(3) (2012)1379-1384.

[36] C.C. Aggarwal, and C.K. Reddy, DATA CLUSTERING Algorithms and Applications CRC Press Taylor & Francis Group, New York, 2014.

[37] J. MacQueen, Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press (1967) 281-297.

[38] K. Arai, and X.Q. Bu, ISODATA clustering with parameter (threshold for merge and split) estimation based on GA: Genetic Algorithm, Reports of the Faculty of Science and Engineering, Saga University 36 (1) (2007).

[39] R.T. Ng, and J. Han, CLARANS: A Method for Clustering Objects for Spatial Data Mining, IEEE14(5) (2002) 1003 - 1016.

[40] D. Judd, P.K. McKinley, and A.K. Jain, Large-scale parallel data clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 871-876.

[41] D.R. Edlaa, and P.K. Janaa, A prototype-based modified DBSCAN for gene clustering, Procedia Technology 6 (2012) 485-492.

[42] Y. Rani, Manju, and H. Rohil, Comparative Analysis of BIRCH and CURE Hierarchical Clustering Algorithm using WEKA 3.6.9, The SIJ Transactions on Computer Science Engineering & its Applications (CSEA) 2 (1) (2014) 25-29.

[43] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison Wesley Publishing Company, Boston, USA,1989.

[44] Z. Michalewiz, Genetic Algorithms+Data Structures=Evolution Programs, third ed., Springer-Verlag, 1996.

[45] R.N. Mohd, and J. Geraghty, Genetic Algorithm Performance with Different Selection Strategies in Solving TSP, Proceedings of the World Congress on Engineering II (2011).

[46] M. Sakthi, and A.S. Thanamani, An Effective Determination of Initial Centroids in K-means Clustering Using Kernel PCA ,International Journal of Computer Science and Information Technologies 2 (3) (2011) 955-959.

[47] A.K. Jain, Data clustering: 50 years beyond k-means, Pattern Recognition Letters 31(8) (2010) 651-666.

[48] K. Miettinen, Nonlinear Multi-objective Optimization, Kluwer Academic Publishers, Boston , USA, 1999.

[49] R.E. Steuer, Multiple Criteria Optimization: Theory, Computation, and Application, Wiley, New York, 1986.

[50] N. Srinivas, and K. Deb, Multi-objective optimization using non-dominated sorting in genetic algorithms, Evolutionary Computation 2 (3) (1999) 221-248.

[51] T. Murata, H. Ishibuchi, and H. Tanaka, Multiobjective genetic algorithm and its application toflowshop Scheduling, Computers and Industrial Engineering 30 (4) (1996) 957-968.

[52] R. Sivaraj, and T. Ravichandran, A review of selection methods In genetic algorithm, International Journal of Engineering Science and Technology 3(5) (2011) 3792-3797.

[53] G.-H.Tzeng, and J.-J. Huang, Fuzzy Multiple Objective Decision Making, Chapman and Hall/CRC, 2013.

[54] M. Kaya, The effects of two new crossover operators on genetic algorithm performance, Applied Soft Computing 11 (2011) 881-890.

[55] M. Gen, and R. Cheng, Genetic Algorithms and Engineering Optimization, John Wiley & Sons, Inc., New York, 2000.

[56] R. Faraji, and H.R. Naji, An efficient crossover architecture for hardware parallel implementation of genetic algorithm, Neurocomputing 128 (2014) 316– 327.

[57] O.Abdoun, J. Abouchabaka, and C. Tajani, Analyzing the Performance of Mutation Operators to Solve the Travelling Salesman Problem, International Journal of Emerging Sciences 2(1) (2012) 61-77.

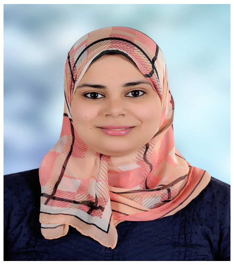
[58] C.-K. Goh, and Y.-S. Ong, Kay Chen Tan, Multi-Objective MemeticAlgorithms , springer, Berlin , Germany 2009.

[59] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, Archiving with Guaranteed Convergence And Diversity in Multi-objective Optimization. In GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference, Morgan Kaufmann Publishers, New York, USA, (2002) 439-447.

[60] A.A. Mousa, and I.M. El_Desoky, Stability of Pareto optimal allocation of land reclamation by multistage decision-based multipheromone ant colony optimization, Swarm and Evolutionary Computation, 13 (2013) 13-21.

A. A. Mosua received the B.Sc. degree in electrical engineering from Menoufia University, Shebin El-Kom, Egypt, in 1997, M.Sc. and Ph.D degrees in engineering mathematics from Menoufia university, in 2003 and 2006, respectively. He has been with the Basic Engineering sciences, Menoufia University, since 1998. He is currently an associate professor in the Basic Engineering Sciences, Faculty of Engineering, Menoufia University and at Taif university faculty of sciences. His research interests are evolutionary multiobjective optimization, multimodal optimization, and optimization techniques applied to power systems.

M. A. El-Shorbagy received the B.Sc. degree in electrical engineering from Menoufia University, Shebin El-Kom, Egypt, in 2004, M.Sc. and Ph.D degrees in engineering mathematics from Menoufia university, in 2010 and 2013, respectively. He is currently an assistant professor in Department of Basic Engineering Sciences, Faculty of Engineering, Menoufia University, Shebin El-Kom, Egypt. His current research interests include: multiobjective optimization, swarm intelligence, numerical optimization and optimization for engineering applications.



M. A. Farag received the B.Sc. degree in Electrical Engineering and the M.Sc. degree in engineering mathematics from Faculty of Engineering, Menoufia University, Shebin El-Kom, Egypt, in 2011 and 2016, respectively. She is working towards the Ph.D. degree in the Department of Basic Engineering Sciences at the same college. She is currently an associate Lecturer in Department of Basic Engineering Sciences, Faculty of Engineering, Menoufia University, Shebin El-Kom, Egypt. Her current research interests include: evolutionary multiobjective optimization, Swarm Intelligence, Numerical optimization and evolutionary algorithms in general.