A Multi-Mode Resource-Constrained Optimization of Time-Cost Trade-off Problems in Project Scheduling Using a Genetic Algorithm

Sedigheh Nader Abadi[a,*], Emad Roghanian[b], Hadi Aghassi[c]

[a] Msc, Department of Industrial Engineering, Arak Branch, Islamic Azad university, Arak, Iran

[b] Assistant Professor, Department of Industrial Engineering, K.N.Toosi University of Technology, tehran, Iran

[c] instructor, Department of computer Engineering, Iran University of Science and Technology, tehran, Iran

Received 19 May, 2010; Revised 25 August, 2010; Accepted 15 September, 2010

Abstract

In this paper, we present a genetic algorithm (GA) for optimization of a multi-mode resource constrained time cost trade off (MRCTCT) problem. The proposed GA, each activity has several operational modes and each mode identifies a possible executive time and cost of the activity. Beyond earlier studies on time-cost trade-off problem, in MRCTCT problem, resource requirements of each execution mode are also allocated and the highest quantities of these resources are limited. In the MRCTCT, the goal is to reduce the total project cost with respect to the resource restrictions .The gene value is encoded as the mode index which is selected from among modes of the activity randomly. For indicating construction mode of the activity, integer encoding is applied instead of binary encoding. Additionally, the selection of genes for mutation is based on chromosome value, as solution convergence rate is high. The crossover operator of GA is based on a two-point method. This paper also offers a multi-attribute fitness function for the problem. This function can vary by decision maker (DM) preferences (time or cost). In this paper, a two-phase algorithm is proposed in which both the effects of time-cost trade-off and resource-constrained allocation are taken into account. A GA-based time-cost trade-off analysis is improved for choosing the execution mode of every activity through the trade-off of time and cost, followed by proposing a resource constrained allocation algorithm to generate an optimum schedule without overriding the project constraints. Lastly, the model is verified by means of a case study and a real project.

Keywords: A multi-mode resource constrained; Project scheduling; Time-cost trade-off; Resource constrained allocation; Multi- attribute fitness function.

1. Introduction

Scheduling of projects has many complicated factors. Making inappropriate decisions on assigning resources or choosing execution activities, like crew size and equipment, might cause serious problems such as project postponement or cost overflow.

Costs of a project are categorized into two main groups. First group is direct costs, which are necessary costs and properties that are used directly in the project such as resources, employees, material, energy , etc. Beyond direct costs which are involved in the projects directly, there is another type of tasks that would cost during the project indirectly, e.g. department of management, engineering, accounting, personnel funding, etc.

Generally, spending long duration for running an activity often needs to a less direct cost for it. There are two types of times in the time-cost scheduling problem [1]: (1) Normal Time, which is duration that the activity can be executed by its minimum direct cost. (2) Forced time, which is the minimum time that the activity can be executed. Generally, trying to decrease time length of a given project will increase direct costs of activities of the project. In addition, by decreasing execution time of activities and sequentially by decreasing execution time of the entire project, indirect costs of the project would decrease. If maximum and minimum times of performing an activity are prepared, we can select a favorable or economic time between these minimum and maximum times as the project has minimum sum of direct and indirect costs. Fig 1. illustrates trends of direct and indirect costs and sum of these costs. Obviously, the best economical time length of executing projects is when sum costs curve be minimum with respect to time. Thus, in most of the economical cost mixtures, our goal is to decrease the projects' activities

*Corresponding author’s E-mail: g.naderabadi@gmail.com

duration according to existing budget, as the project is executable in its shortest time.
One of the popular techniques for project scheduling is CPM (CPM stands for Critical Path Method) method [2].
The critical path method (CPM) is widely used as a planning and scheduling tool for construction projects [1,2,33]. In CPM, resource limitation is not considered and an activity can always start as long as all its predecessors are completed. This, however, is not practical, as resources are limited and the availability of resources would affect resource allocation and project scheduling. Furthermore, based on resource availability, the duration of an activity might vary, which results in various execution modes. Having these situations, we motivated to propose a new approach for above-mentioned resource-constrained project scheduling problems.
When a CPM network is prepared, each activity will be demonstrated by earliest start time (ES), earliest finish time (EF), latest start time (LS), latest finish time (FS), and activity duration (*t*).

In this paper, we used discrete points to relate cost and time of activities, because our GA technique would be more similar to real conditions.

In assigning chromosome values, an executive mode among the different modes of an executive activity is selected for its relevant gene haphazardly. For instance, a sample chromosome in 0represents a different feasible solution for the project [3] including the activities shown in Table 1. For activity A in the chromosome , the executive mode of 2 is assigned randomly involving the running time of 5 and the running cost of 290000 USD and the amounts of resource required of R1= 2&R2=1&R3=1.
Analytical and heuristic approaches are the most well known methods to time-cost trade-off scheduling problems. In analytical approaches, mathematical programming is utilized for solving problems, such as linear programming or dynamic programming, [4–10]. The formation of mathematical models is usually hard and it entails intense computation attempt. Therefore, they can only be utilized for small-sized projects [11]. Because heuristic algorithms have uncomplicated and convenient characteristics, they have been used to solve time-cost trade-off problems. These algorithms often introduce high quality solutions; however, they do not promise to find the optimum solution and their dependence to problem is proven. Fondahl's method [12], effective cost slope model [13], and structural stiffness model [14] are samples of heuristic methods. Currently, the genetic algorithm (GA) has become common in solving time-cost trade-off problems. Feng et al. proposed a model using the genetic algorithm and the Pareto front approach to solve construction time-cost tradeoff problems [15]. Leu et al. proposed a GA-based fuzzy construction time-cost trade-off model [16], in which the effects of both uncertain activity duration and time-cost trade-off are taken into account. These time-cost trade-off approaches did not deal with the problems concerning activity-relevant constrictions such as priority associations, resource requirement and accessibility, interruption and overlapping of activities, etc.

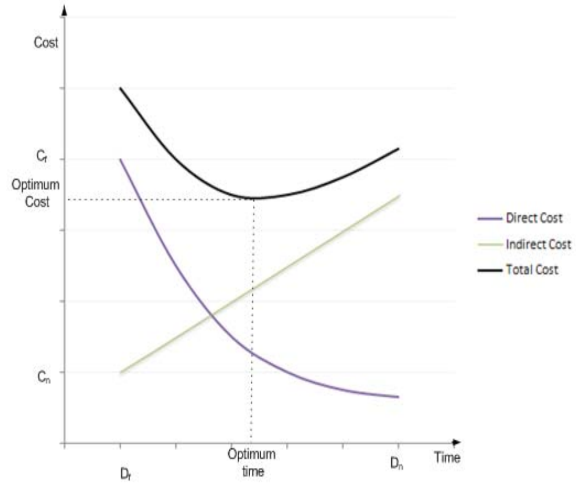

Fig. 1. Optimal project duration based on the least total cost

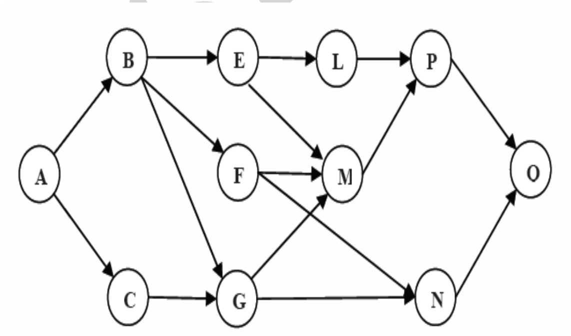

Fig. 2. Activity network of the example project

Ordering chromosome

A	B	E	C	L	G	F	M	N	P	Q

chromosome

4	3	3	2	3	1	2	3	1	2	3

Fig. 3. One candidate solutions for initial population

Table 1
Activity data of the example project

Activity name	Duration	Cost	Resource1	Resource2	Resource3
A	2	580	3	4	5
A	3	450	3	3	4
A	4	320	2	3	1
A	5	290	2	1	1
B	4	1700	6	3	4
B	5	1250	5	3	3
B	6	800	4	2	1
C	2	2800	5	5	3
C	3	2650	4	4	3
C	4	2500	3	4	2
C	5	2350	3	3	1
C	6	2200	3	2	1
C	7	2050	2	1	0
E	3	3780	3	2	4
E	4	3460	3	1	3
E	5	3140	3	1	2
F	4	6030	4	6	3
F	5	5395	4	4	2
F	6	4670	3	3	2
F	7	4125	2	2	1
G	4	6730	4	2	5
G	5	6320	3	1	4
G	6	5910	2	0	3
G	7	5500	1	1	2
L	2	3020	1	5	5
L	3	2870	1	4	5
L	4	2720	1	3	4
M	2	2030	5	1	1
M	3	1830	4	1	1
M	4	1630	3	1	0
P	2	1100	3	5	1
P	3	1000	2	4	1
P	4	900	1	3	0
N	5	1500	4	4	5
N	6	1300	4	3	4
N	7	1100	3	2	3
N	8	900	3	1	1
Q	1	620	2	0	1
Q	2	450	1	0	1
O	3	350	1	0	0

2. Related Works

Genetic algorithm concept was offered by John Holland for the first time [17]. GA not only is an issue of human existence, but it also simulates nature evolution [18]. Many researchers believe that this algorithm is a suitable approach for encountering optimization problems [19]. General structure of GA that is a random search technique is described as follows:
Entire initial population chromosomes and their children that are produced by mutation and crossover operators are evaluated by means of their fitness values (often in optimization problems, fitness is our goal function), and the best chromosomes can move to next generation. To choose chromosomes, we have to use selection mechanisms. After several generation iterations, algorithm will converge with optimum solution or a solution that is close to it. These stages are based on GA. There is a large variety of mutation and crossover operators and also there are many methods of applying them including selection methods, initial population generating methods, and selection parent mechanisms for mating. This large variety caused researchers to be interested in GA issues [20, 21].
Linear solutions of the TCTS problem are often far from optimum solution. In [24] a non-liner extension of optimized TCTS is described and the results are more optimum. In fact, to decrease the searches number, the search path counts are decreased. This approach is not using GA.
Authors in [25] implemented a GA for the TCTS problem, but still the preferences of DM is disregarded in which for finding the minimum costs of all activities, the applied search considers the deadline of activities, daily indirect cost, etc.
Except heuristic approaches, there are many researches on the mathematical scheduling models. Mathematical approaches are useful for numerical, dynamic, and linear models, but they are complex and time-consuming processes. Authors in [23] have shown that both the deadline problem and the budget problem are NP hard in the strong sense for the discrete time-cost trade-off model when the underlying project network is a general directed acyclic network. Thus, it is unlikely that there exists a fully polynomial time approximation scheme for either the deadline problem or the budget problem of our model. In fact, developing polynomial time approximation algorithms for the discrete time-cost tradeoff model is a challenging task.
Some new researches are done for modeling GA to resolve time-cost trade-off problem. Research [22,33] offered a multi-attribute fitness function which uses costs differences for computing the importance of chromosomes and also GA operators were designed as the convergence of the algorithm is high. This research for each activity used the forced and normal times (costs) and the time-costs were chosen between these values, so cost slope during all activity is a fixed number, hence it could change in real projects. We adopted the fitness function and GA technique presented in [22,33] to obtain a better convergence of the algorithm for piecewise time-cost scheduling problem.
Generally, resource constraints have a great influence on the possibility of a project schedule and being the optimal schedule. For instance, allocating extra resources to the project could reduce the length of a project. Nevertheless, the duration of reduced time may have dependency on the accessibility of the constrained resources. Consequently, in resource-constrained time cost trade of problems, resource accessibility and allocation is crucial for the creation of possible schedules.
Preceding researchers offered some approaches to discover the correct optimal solution to resource allocation problems by means of mathematical programming methods [26–28]. In practice, these optimization methods are very time-consuming, and they often have intense computations for large projects

because of a gigantic quantity of variables and constraints. For this reason, heuristic approaches can serve very important function of allowing the optimal project schedule to be discovered.

With the combination of GA with other methods, numerous useful applications have been introduced to solve resource-constrained problems [29, 30]. These resource-constrained scheduling approaches chiefly supposed only one execution mode for every activity and revolved around the impact of resource constraints on project length. In addition, solutions for decreasing project length in these methods were proposed in such a way that activities could be interrupted and overlapped.

In our proposed model, the authors combined the ideas of the aforementioned analytical research (time-cost trade-off analysis and resource scheduling) with the heuristic approaches using a two-phase algorithm to solve resource-constrained project scheduling problems (RCPSP). In this model, a GA-based time-cost trade-off analysis is used to find out which execution mode should be chosen for every activity. After selecting activities modes by means of GA, some lightweight analytical computations are applied in the next phase to identify the minimum execution time of entire project according to the selected modes with satisfaction of resource constraints. For proposed model, a collection of constraints is presented such as priority associations, resource requirements and accessibility of activities, etc. We developed our model with an easy-to-use user interface. In the next sections, proposed model is described and evaluated in details.

3. Multi-mode resource constrained time cost trade-off problem features.

Normally, in a multi-mode resource constrained time cost trade off problem, projects include a collection of pertained activities. In this section, we are going to explain features of activities in this type of problem. Activities have several modes that each activity can be executed in one of these modes in which a certain number of resources are needed, and the total number of in charge resources cannot exceed limited size of resources. For this purpose, after finding each solution candidate in which all optimal modes of activities are predicted and after computing the total project time according to resources limits, an investigation must be done to check if the above mentioned conditions on the solution are satisfied. The features of resource-constrained time cost trade-off problems from two point of view including activities and resources are described as follows.

3.1. Activities

A generic object-oriented data structure for indicating an activity is developed in [31]. This object-oriented form of activity can show the data of an activity at several levels and design steps. In proposed model, the object-oriented form of activity is chosen to demonstrate the features of a typical resource constrained time cost trade off problem as illustrated in Eq. (1).

Activity{E;D; C; P; R; S} (1)

where,

E execution modes. Execution modes of activities can be more than one. The collection of resource requirements and estimated duration and activity cost is distinctive for every mode of an activity.

D duration of activity.

C cost of activity.

P precedence.

Each activity has its precedence associations.

On a regular basis, after all predecessors of an activity are finished, it can start.

R required resource.

Requirements of resources of each activity are capable of being renewed. Generally, the costs of renewable resources are calculated according to their hourly or daily rates. For instance, equipment and labor are renewable resources.

S activity state.

The three states of an activity could be one of the following states:

scheduled but not started (SC), ongoing (ON), and completed (CO) That is:

S = {SC;ON; CO} (2)

The precedence associations within a construction project can be categorized into four clusters: FS (finish-to-start), SS (start-to-start),SF (start-to-finish) and FF (finish-to-finish). Consequently,

T = (FS; SS; SF; FF) (3)

In this paper, we just deal with the FS (finish-to-start) relation between activities for sensing purposes and plainness.

Resources are capable of being more divided into renewable (RE) and nonrenewable (NR) resources. As presented in Eq. (4), two model categories of renewable resources are labor (L) and equipment (E).

R = {RE;NR} ; RE = (L; E) (4)

By providing the data to the object-oriented properties, activity objects could be represented by instances through associated models. By the development of the project, the data could be edited, and then project schedule could be adapted or edited based on the edited data.

3.2. Resources

Generally, the number of resources is restricted. In this paper, we concentrate on assigning and sorting renewable resources. Therefore, non-renewable resources, such as raw materials, etc., which could generally be assumed as a constant number according to capacities required, are not explained in detail in this paper.

For sensing purposes and simplicity, three resources types (R1, R2, R3) is considered for all activities and the maximum amount of resource accessibility is shown in Table 2.

Table 2
The maximum amount of resource accessibility

Maximum Resource1	Maximum Resource2	Maximum Resource3
6	6	6

4. Implementation of the GA technique

Indirect cost of the project is often a constant value per time unit. We used 150 indirect cost units per day for all activities in our research. Also, by changing preferences for cost and time identities, DM can make the results ideal. In general, GA serves as a selection engine to screen out the construction alternative that produces bad system performances (i.e., longer project duration and high cost) in GA-simulation mechanism. The mechanism of creating chromosome structure, deciding fitness value, selection, crossover, and mutation operations are introduced in the current section.

A. Genetic algorithm

As summary, a standard multi-mode resource constrained time cost trade off optimization procedure concerning the genetic algorithm is shows as follows:

- Specify the initial population size (N) and specify the iterations (generations' number=M) and Specify α and β
- Generate initial population of solutions
- While (terminating condition not met or generations' number=M) repeat following steps:
  - Applying cross over operator on the first population and produceing offsprings (children)
  - Applying mutation operator on the first population and produceing offsprings
  - Checking if the children are under the conditions of the problem
  - For each chromosome call a procedure of Resource Constrained Allocation subsystem and calculate zt and zc (second phase)
  - Calling the procedure of time cost trade off subsystem which consists of the following steps:
  - Evaluate solutions through fitness assignment;
  - Evaluate offsprings through fitness assignment;
  - Sort chromosomes according to their fitness value;
  - Produce a new population using the top N chromosomes of end population.
  - Select better solutions based on fitness value as the best scheduling

B. Genetic Algotithm Initial Variables

At this stage of the algorithm, initial population size, iterations count (generations' number), crossover rate, and mutation rate are identified. In the proposed model, the process is executed for a fixed number of iterations. It is proved that the elitist model of GA will find the optimal solution as the number of iterations tends to infinity [32]. Notice that increasing the size of initial population, iterations count, and crossover rate could cause search space to be extended leading to the convergence of the algorithm. Moreover, mutation operator is an implicit operator, and it is better to assign a low mutation rate to prevent a purely random process.

We used 0.1 for mutation rate in our research. In addition, having experiences on several project's data, it seems that a value between 0.7 and 0.8 is good for crossover rate and deviating from this range exacerbates the result. To demonstrate this fact, we analyzed our case-study project, and after 11 times execution of the proposed approach with values of 0.75, 0.45, and 0.95 for the crossover rate, the results are shown in Table 0illustrating better project time and less CPU overhead time for the 0.75. Having these experiences, we considered 0.75 for crossover rate.

Table 2
Experiences of different crossover rates of the proposed approach

Crossover Rate	Avg CPU Time	Project Best Time
0.75	16.646	60
0.45	27.661	63
0.95	27.265	64

Large initial population size is often profitable, but it takes a long time for the algorithm to be processed. On the other hand, by choosing a small number for the initial population size, the search could be unfortunate or too time consuming. We had to choose a trade-off for size of initial population. We observed that selecting the size of 2.5 times as long as the chromosome size gives a good answer in a short time. For evaluating this, we examined the proposed approach for different population sizes of our case-study project. The average result of 11 times running of project is shown in 0By choosing 18 ($2.5 * 7 \approx 18$) for population size we have the best solution of the problem with the minimum CPU overhead.

Table 4
Experiences of different population sizes of the proposed approach.

Population Size	Avg CPU Time	Project Best Time
18	16.646	60
10	19.672	61
30	27.410	60

C. Initial Population

For the first step at this stage, we produce initial population at random. Initial population includes strings in which each string represents a schedule for the project and each cell of the string represents the execution mode of one activity in the project that includes the duration of activity. The maximum value of this time length of cells is equal to normal time of that activity and minimum time length is equal to forced time.

In producing new population strings, we should set a random value for each cell among the modes of concerning activity. Notice that in producing chromosomes, serial method is applied. Procedure of initial population production can be summarized as follows:

- Sort project activities according to their ES (observe prerequisite relations).
- Identify population size (number of chromosomes).
- Create strings with the length of activity count in the project.
- For each chromosome iterate following steps:
    - For each activity, choose a random number among the modes of that activity.
    - Put each random number into genes according to the order of first step.

D. *Crossover Operator And Mutation Operator*

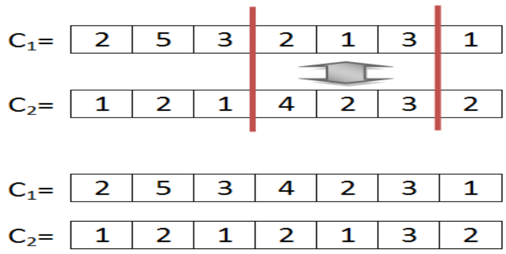

Fig. 4. Two-point crossover

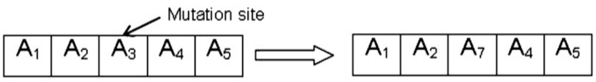

Fig. 5. Uniform mutation

In this GA technique, the types of crossover and mutation operators used in are two-point crossover and uniform mutation. Crossover operates on two chromosomes at a time and generates the offspring by combining both chromosomes' features.
There are several selection methods for choosing parents in the crossover operator. One of the methods in GA is Roulette Wheel selection in which parents are selected according to their fitness. Elitism selection method can enhance the performance of GA quickly because it prevents losing the best-found solutions. In two-point crossover for two chromosomes that are going to be parents, two points are selected randomly. Then two chromosomes break up in these two points (genes) and the portions between points are exchanged and generates the offspring.

Fig. 4. shows a sample crossover operation on our two example chromosomes C_1 and C_2. In this sample, two points 4 and 7 are selected randomly and gene values between these points in two chromosomes are exchanged and the new chromosomes are demonstrated in the figure.
Beyond crossover, mutation is a background operator that produces spontaneous random change in various chromosomes and tries to make some variations in the population. The purpose of this stage is to find new better chromosomes by random changes in the chromosomes.

Uniform mutation alters one gene in a chromosome, depending on the defined mutation rate (Fig 5.).

E. *Multi-Attribute Optimized Fitness Function*

After creating initial population, value of fitness function should be calculated for each chromosome. In multi-attribute problem for each chromosome, this function is achieved by normalizing time and cost values, and assigning certain weights to each parameter. While the given parameters are indicators of the time and cost importance, if DM prefers to have shorter time for the entire project, then the weight of time attribute can be considered larger than the weight of cost attribute (e.g. $2w_t=w_c$). And if DM wishes to have less cost for the entire project, then the weight of time attribute can be considered smaller than the weight of cost attribute (e.g. $w_t=2w_c$).

Fitness value of the chromosome will be computed as:

$$f(x) = \frac{\left[w_t\left(\frac{Z_t^{max}-Z_t+\delta}{Z_t^{max}-Z_t^{min}+\delta}\right)+w_c\left(\frac{Z_c^{max}-Z_c+\delta}{Z_c^{max}-Z_c^{min}+\delta}\right)\right]}{(w_t+w_c)} \qquad (5)$$

Where f(x) is the rate of fitness for chromosome x, Z_t^{min} is minimum time in chromosomes of one generation, Z_t^{max} is maximum time in chromosomes of one generation, Z_c^{min} is minimum cost in chromosomes of one generation, Z_c^{max} is maximum cost in chromosomes of one generation, δ is random number between zero and one, Z_t is the execution time of each chromosome based on the output of **Resource scheduling subsystem**, also Z_c is the execution cost of each chromosome based on the output of **Resource scheduling subsystem** for each chromosome, which is equal to sum of direct and indirect costs of that chromosome. Our approach relies on the fact that the importance of the cost or time is variable. This kind of variation is controlled by means of two w_t and w_c attributes representing the weight of time attribute and the weight of cost attribute.

F. *Selection Method and Termination Conditions*

At the last stage, the selection method should be applied. We used (μ+λ) mechanism for selection. In this mechanism, all earlier population chromosomes and all child chromosomes that were produced by mutation and crossover operators are gathered in a set and sorted according to their fitness values and based on problem objective function computations. Then, we select the best chromosomes in the size of the initial population.

Finally, iteration count is checked. If all iterations are done, the best chromosome of the population should be returned as the solution of the problem; or else, stage two should be executed again as next iteration.

5. Two-phase model for a multi-mode resource constrained time cost trade-off problem

Two phases of the two-phase model for resource-constrained time cost trade-off problems are: the time-cost trade-off subsystem, the resource constrained allocation subsystem. These subsystems are explained in detail as follows.

5.1. Time-cost trade-off subsystem

The initial population is created randomly that contains strings of modes of activities in which each string stand for a feasible solution. Every mode of an activity is represented by a gene value in the chromosome string. After the identification of the execution mode for each activity, the associated activity cost, duration, and resource requirements will be recognized. Subsequently, the resource scheduling subsystem will get activity mode data to produce a possible schedule according to the constraints of the project.

In response, the resource scheduling subsystem send back the total project duration and cost for per execution-mode string to the time-cost trade-off subsystem for evaluation. The time-cost trade-off subsystem uses two-point crossover and uniform mutation operators to create possible offspring strings.

The fitness value for every chromosome is computed using the fitness function explained in section D. The chromosomes that are allowed to go to next generation are chosen based on the (μ+λ) mechanism described in section E.

The elitist selection method is combined with the selection procedure to preserve the best chromosomes for the next generation.

In the last stage of time-cost trade-off subsystem, the best chromosome with the highest fitness value is returned to the output component. To guarantee an optimal chromosome could be achieved, a list with various mixtures of GA parameters is essential. In this proposed two-phase GA technique, the best parameter pattern recognized is shown in Table 5.

Table 5
Parameters of the GA technique time-cost trade-off subsystem

Population	Generation	Crossover rate	Mutation rate
100	50	0.75	0.1

5.2. Resource Constrained Allocation subsystem Computer Algorithm to specialize limited resources

Assumptions:

1- Phases of performance logic of activities and their time estimation should be specified.
2- Amount of required resources should be specified to execute each activity and fixed during the time.
3- For each resource, the maximum accessible number should be defined during the time.
4- Stopping the activities is not permissible up to their final completion.

Steps:

Step 1: determining ES_{ij} and LS_{ij} for each activity and current time of planning. In the current scheduled Time, T is considered as 1 as well as all of the accessible resources should be placed equal to R.

Step 2: We have to organize the eligible activity set (EAS) or set of activities, which their priority activities are executed.

Step 3: According to EAS set, for scheduling, the set of ordered scheduled set (OSS) is constructed, which is an arranged set of activities. Activities of this set are arranged based on ascending order of Ls_{ij}. In case that Ls_{ij} values are equal, the activities are arranged in ascending time order of D_{ij}.

Step 4: Available resources of R in order of OSS according to amounts of r_{ij} are allocated to activities as much as possible, and their final times are computed.

In this algorithm, the most important factor in the selection of order of activities executions, at the first stage, is the latest time for the beginning of activity (LS), and if LS values are equal at the second stage, is the minimum time of the activities execution (D). As mentioned above, in order to determine the LS values of activities, CPM backward calculation is used, and to determine ES values of activities, CPM forward calculation is used.

After determining values of LS and ES for each of the activities, we set up EAS set. At first, this set contains some of activities that do not have any prerequisite activity, and at the next stages, it consists of activities, which their prerequisite activities are executed. After selecting EAS set, subgroup of OSS is defined using EAS set.

Activities of this set are arranged based on Ls_{ij} in ascending order, or if are equal, are based on time of D_{ij} in ascending order. Then, in respect of the priority of doing activities, the rate of their needed resources, and the maximum of available resources for each activity, some activities have the possibility to find specialized resources. The remaining activities that do not have possibility to find specialized resources are postponed to the next day. Undoubtedly, after these stages, it is necessary to update EAS, remained resources, and ES.

The condition of completing algorithm is the emptiness of EAS set.

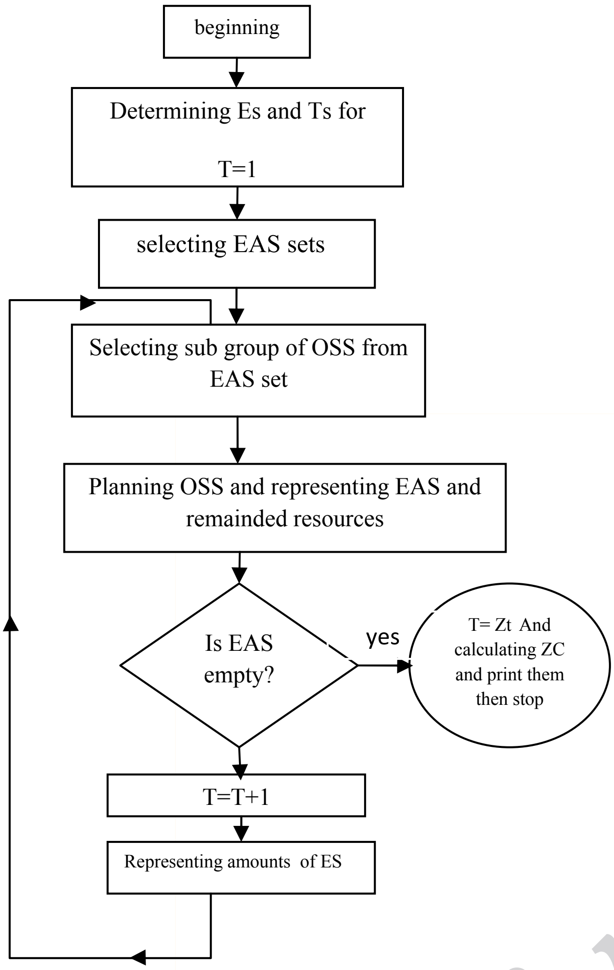

Fig. 6. Flow of the resource scheduling subsystem

6. Evaluation

Comparing proposed genetic algorithm of time-cost trade-off in terms of unlimited resources to one in terms of limited resources:
In this section, two proposed algorithms have executed over the project of example 1 provided section 1. Because of random features of genetic algorithms, these algorithms have performed over projects for 11 times, and the average of their performance duration has been compared. In performing these two algorithms, at first, the algorithm of time-cost trade-off is performed in terms of unlimited resources, and then its issued results are compared to the results of the time-cost trade-off algorithm in terms of limited resources.
To carry out these algorithms, the programming language of C#.2008 was used. These algorithms were performed over a computer with the processer 2.13 Intel Pentium, two Gigabyte RAM and over Windows 7.

To consider the efficiency of the proposed model, at first a cost-time trade-off issue in terms of unlimited resources and one in terms of limited resources were considered.

A- Considering the issue in case of unlimited resources

To solve the issue in case of unlimited resources, it is required to assume the maximum supply is very high. After very efforts and errors in selecting genetic algorithm parameters with below parameters we will be achieved to this ideal answer time 21, cost=23936.

Population size= 100
Type of selection= Elitism & $\lambda+\mu$
Generation Number= 50
Weight of time=1
Weight of cost=1
Type of fitness function: Normal
Pc =0.75 (exchange cost)
pm =0. 1(leaping cost)
In this case, the minimum and the maximum of project time and project cost, are considered as:
Maximum Time = 16
Minimum Time = 30
Maximum cost = 28920
Minimum cost = 22405

If the significance of attributes varies according to DM's view, the following answers are achieved.

Wt =1 , Wc=2	Time= 24 ,Cost = 23315
Wt =1 , Wc=3	Time=25 , Cost = 22801
Wt =1 , Wc=5	Time=27 , Cost = 22607
Wt =1 , Wc=10	Time=28 , Cost = 22622
Wt =2 , Wc=1	Time=18 , Cost = 26720
Wt =3 , Wc=1	Time=17 , Cost = 27637
Wt =10,Wc=1	Time=16 , Cost = 28920

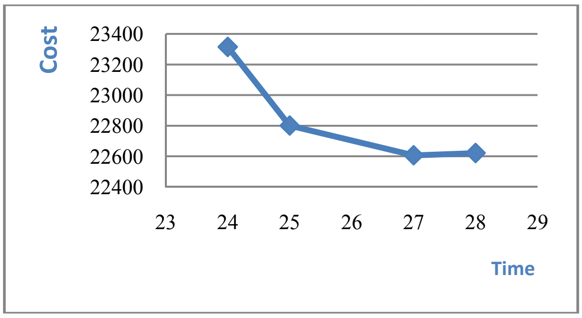

Fig. 7. The chart of time-cost of the project in case of unlimited resources

B- Considering the issue in the case of limited resources

In this case, the maximum of resources is considered as six. After very efforts and errors when selecting genetic algorithm parameters with below parameters, we will achieved to this ideal answer:

Population size= 100
Type of selection: Elitism & $\lambda+\mu$
Generation Number= 50
Weight of time=1
Weight of cost=1
Type of fitness function: Normal
P_c =0.75 (exchange cost)
p_m =0. 1(leaping cost)

If the significance of attributes varies according to DM's view, the following answers are achieved.

Wt=3 , Wc=1	Time= 26 , Cost = 22895
Wt=5 , Wc=1	Time= 25 , Cost = 23395
Wt=8 , Wc=1	Time= 24 , Cost = 24185
Wt=10 ,Wc=1	Time= 23 , Cost = 24965
Wt= ۱, Wc =10	Time= 30 , Cost = 22405
Wt= ۱, Wc =8	Time= 29 , Cost = 22435
Wt= ۱, Wc =5	Time= 28 , Cost = 22585

As observed, in this case, the minimum feasible time of the project is increased from 16 to 23 because of the lack of the resources.
The time-cost chart of the project concerning DM priorities is illustrated in 0.

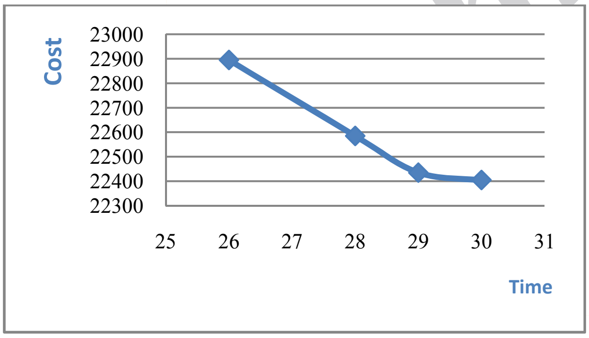

Fig. 8.The chart of time-cost of the project in case of limted resources

In general, having opposite criteria in time-cost trade-off problem, DM can select his/her own ideal answer according to the chart of the project time-cost.

7. Conclusions

In the proposed approach, a two-phase model is introduced to solve the resource-constrained time-cost trade-off problem. Both time-cost trade-off and resource-constrained allocation are concerned in the proposed approach.
Resource limits are considered in the second phase of our approach. These limits are used due to suggest the best time order of activities during the project while suggested modes of activities are determined by the first phase developed using GA algorithm. The GA technique, which is presented closely in this paper, has applied to solve time-cost trade-off problems in case of limited resources by means of a multi attribute, decision-making method.
Although, the proposed GA technique needs GA parameters to be arranged by many efforts, such as mutation and crossover possibility, it has the following advantages.

- It can consider objectives of time-cost trade-off problem and allocating resources in a time that lead to solve the problem of time-cost trade-off in case of limited resources. Undoubtedly, considering common mathematical and searching models for these objectives simultaneously is very difficult.
- Regarding DM priorities in terms of cost and time attributes, this model can present entire project cost-time chart.
- The proposed GA technique in this research has higher flexibility to solve problems compared to other methods, because using empirical rules to solve problems is not necessary and because limitations and objective functions are not necessary while formulating is required.

Because the presented GA technique in this research is multi attribute, the final answers are one pointed. For future works, to make this model in the form of multipurpose, DMs can use *pareto* front in order to make their decision.

8. References

[1] Chen, P.H., Weng, H.J. A Two-Phase GA Model for Resource-Constrained Project Scheduling. Automation in Construction, 18(4), (SCI, EI), 485-498. July 2009.

[2] Siemens, N. A Simple CPM Time-Cost Tradeoff Algorithm. Management Science, 17(6), Application Series Feb. 1971.

[3] M, D.X., Ng , S.T.and Kumaraswamy, M. M. Applying a Genetic Algorithm-Based Multi-objective Approach for Time-Cost Optimization.Journal of Construction Eng. and Management, ASCE, 130(2), 168-176. 2004.

[4] Burns, S. A., Liu, L. and Feng, C.W. The LP/IP hybrid method for construction time-cost trade-off analysis, Constr. Manag. Econ. 14 (3), 265–276. 1996.

[5] Butcher, W.S. Dynamic programming for project cost-time curve, J. Constr. Div.,ASCE 93 (C01), 59–73. 1967.

[6] Elmagrhraby, S.E. Resource allocation via dynamic programming in activity networks, Eur. J. Oper. Res. 64, 199–215. 1993.

[8] Kelly, J.E. Critical path planning and scheduling: mathematical basis, Oper. Res. 9(3), 167–179. 1961.

[9] Meyer, W.L., Shaffer, L.R. Extending CPM for multiform project time-cost curves, J. Constr. Div., ASCE 91 (C01) ,45–67. 1965.

[10] Pagnoni, A. Project engineering: Computer oriented planning and operational decision making, Springer, Berlin. 1990.

[11] Panagiotakopouols, D. Cost-time model for large CPM project networks, J. Constr.Eng. Manag., ASCE 103 (C02),201–211. 1977.

[12] Fondahl, J.W. A non-computer approach to the critical path method for the construction industry, Technical Rep. No. 9, The Construction Institute, Department of Civil Engineering, Stanford University, Stanford, Calif. 1961.

[13] Siemens, N. A simple CPM time-cost tradeoff algorithm, Manag. Sci. 17 (6), B-354-0B-363. 1971.

[14] Moselhi, O. Schedule compression using the direct stiffness method, Can. J. Civ.Eng., Ottawa 20 (1), 65–72. 1993.

[15] Feng, C., Liu, L. and Burns, S. using genetic algorithms to solve construction time-cost trade-off problems, J. Comput. Civ. Eng., ASCE 11 (3), 184–189. 1997.

[16] Leu, S.S., Chen, A.T. and Yang, C.H. A GA-based fuzzy optimal model for construction time-cost trade-off, Int. J. Proj. Manag. 19 (1), 47–58. 2001.

[17] Holland, J.H. Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, Michigan. (1992). re-issued by MIT Press. 1975.

[18] Gen, M., Cheng, R. Genetic Algorithms and Engineering Design, John Wiley & Sons. 1997.

[19] Goldberg, D. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley. 1989.

[20] Kea, H., Maa, W., Ni, Y. Optimization models and a GA-based algorithm for stochastic time-cost trade-off problem Applied Mathematics and Computation, 215 (1), 308-313. 2009.

[21] Wuliang, P., Chengen, W. A multi-mode resource-constrained discrete time-cost tradeoff problem and its genetic algorithm based solution. International Journal of Project Management, 27 (6), 600-609. 2009.

[22] Aghassi, H., Nader Abadi, S. and Roghanian, E. A Multi-Objective Genetic Algorithm for Optimization Time-Cost Trade-off Scheduling. Third Malaysian Joint Conference on Artificial Intelligence, Kuala Lumpur , Malaysia. 2011.

[23] De, P., Dunne, E.J., Ghosh, J.B., Wells, C.E. Complexity of the discrete time-cost tradeoff problem for project networks. Operations Research, 45, 302-306. 1997.

[24] Al-Haboubi, M., Duffuaa, S. Cost-time trade off for projects with nonlinear cost functions. Proccedings of the International Industrial Eng. Conf., Tehran,Iran, 2, 224-233. 2004.

[25] Hegazy, T. Optimization of Construction Time-Cost Trade-Off Analysis using Genetic Algorithms. Canadian Journal of Civil Eng., CSCE, Canada. Dec. 1999.

[26] Adeli, H., Karim, A. Scheduling/cost optimization and neural dynamics model for construction, J. Constr. Eng. Manag., ASCE 123 (4), 450–458. 1997.

[27] Davis, E., Heidorn, G. An algorithm for optimal project scheduling under multiple resource constraints, Manag. Sci. 17,803–816. 1971.

[28] Pritsker,A., Waters, L. and Wolfe, P. Multi project scheduling with limited resources: a 0-1 approach, Manag. Sci. 16, 93–108. 1969.

[29] Chan, W.T., Chua, D.K.H. and Kannan, G. Construction resource scheduling with genetic algorithms, J. Constr. Eng. Manag., ASCE 122 (2), 125–132. 1996.

[30] Gen, M., Cheng, R. Genetic Algorithms & Engineering Design, John Wiley & Sons, Inc, New York, N.Y. 1997.

[31] Shi, J., Deng, Z.M. Object-oriented resource-based planning method (ORPM) for construction, Int. J. Proj. Manag. 18 (3), 179–188. 2000.

[32] Bhandari, D. C., Murthy, A. and Pal, S. K. Genetic algorithm with elitist model and its convergence. Int. J. Pattern Recognit. Artif. Intell., 10, 731–747. 1996.

[33] Nader Abadi, S., Aghassi, H. and Roghanian, E. A Multi-Attribute GA for Piecewise Linear Time-Cost Trade-off Scheduling Optimization.Third Malaysian Joint Conference on Artificial Intelligence, Kuala Lumpur, Malaysia. 2011.