Smart GA-Based Cost-Scheduling Model for Resource-Constrained Projects

H. J. Weng [1]and P. H. Chen[2]

Abstract—**In construction scheduling, problems can arise when each activity can be started at different time points and the resources needed by the activities are limited. Moreover, activities have required conditions to be met, such as precedence relationships, resource demands, etc. To resolve these problems, a two-phase GA (genetic algorithms) model is proposed in this paper, in which both the effects of time-cost trade-off and resource scheduling are taken into account. A GA-based time-cost trade-off analysis is adopted to select the execution mode of each activity through the balance of time and cost, followed by utilization of a GA-based resource scheduling method to generate a feasible schedule which may satisfy all the project constraints. Finally, the model is demonstrated using an example project.**

Index Terms—**Resource-constrained project scheduling; genetic algorithms; time-cost trade-off; resource scheduling**

I. Introduction

Construction projects consist of complex facets of work. Improper decisions on selecting construction methods and allocating resources, such as crew size and equipment, could lead to problems like cost overrun or project delay.

Conventional critical path method (CPM) techniques are widely used planning and scheduling tools in the construction industry. They assume in scheduling unlimited availability of resources for each activity. However, in practice, resources are available only in limited quantities and the resource demands of concurrent activities may not be satisfied. These techniques also assume non-interruption and non-overlap for construction activities. In real-world projects, an activity in construction may be interrupted, i.e., its resources may be assigned to other more important or urgent activities first, and then return to the activity itself. Normally, an activity cannot start until its predecessors have completed. However, overlap between an activity and its predecessor is possible in practice. For instance, an activity may start when its predecessor is partially, say 80%, completed. Furthermore, each activity could be performed in several ways depending on how resources needed are arranged and allocated. For example, if two excavators are assigned to an excavation work, it could be completed in half of the time required as it is done by one excavator. However, CPM techniques do not provide a method to decide which execution mode should be selected. Understanding of the abovementioned drawbacks of CPM directed the research towards more general and practical resource-constrained project scheduling problems.

In resource-constrained project scheduling problems, each activity could be executed in more than one mode, and each mode might have different resource requirements, provided limited resource quantities. The general steps of handling resource-constrained project scheduling problems are as follows:

- Creation of a project plan. The plan refers to the proposal of different execution ways for each activity, the estimates of duration, cost and resource requirements for each execution mode, as well as the precedence relationships between activities. Moreover, other constraints, such as availability of resources, the interruption and/or overlap of activities, should be listed in the project plan.
- Generation of schedules. Because selection of different execution modes for each activity can result in different schedules, different feasible schedules are produced for the completion of a project within the project constraints.
- Evaluation of schedule performance. A set of measurements, such as total project duration, total project cost and resource utilization, are provided to measure the performance of different feasible schedules.
- Selection of an optimal schedule. The optimal schedule not only satisfies all the constraints, but also is optimal in given conditions, such as the shortest duration with fixed project cost, or the lowest project cost with fixed project duration.

Construction time and cost are intricately related, e.g., leading to an increase in labor and plant costs (i.e., direct costs) when project duration is compressed; and project overhead (indirect costs) increasing with the project duration.

To minimize the costs associated with schedule compression, construction planners are urged to examine the time-cost optimization of construction activities before a decision is made. The first step for time-cost optimization is to find the trade-off curve between the direct cost and the project duration. Subsequently, construction planners can determine the total cost by summing up the estimated indirect coat and direct cost from the trade-off curve. As shown in Fig. 1, the optimal choice to perform the project would be the lowest total cost. By

[1] H. J. Weng is the research student with the School of Civil & Environmental Engineering, Nanyang Technological University, Singapore (e-mail: hjweng@ntu.edu.sg).

[2] P. H. Chen is the assistant professor with the School of Civil & Environmental Engineering, Nanyang Technological University, Singapore (e-mail: CPHChen@ntu.edu.sg).

adjusting the time for the project schedule to coincide with the optimal time obtained from the combined graphs, it is possible to carry out the work in the optimal overall duration.

The resource-constrained project scheduling problems (RCPSP) arise when each activity can be carried out in several possible ways. The trade-off between project direct cost and duration is the dominant consideration in selecting activity execution modes. For example, an execution mode which uses more productive equipment or hires more workers may give a shorter duration, but the corresponding cost may increase. Selecting a proper execution mode for each activity such that the project can be completed in the most cost-effective way within a specific completion time is desirable.

The most popular approaches to time-cost trade-off scheduling problems are analytical and heuristic methods. Analytical methods use mathematical programming, such as linear programming or dynamic programming, to solve problems [1]-[7]. Generally, mathematical models are difficult to create and require a great deal of computation effort. Thus, they are only suitable for small-sized projects [8]. Some heuristic methods have been developed to solve time-cost trade-off problems due to their simplicity and ease for applications. These methods provide good solutions, but do not guarantee optimality and are proven to be problem dependent. Fondahl's [9] method, effective cost slope model [10], and structural stiffness model [11] are examples of heuristic approaches. Recently, genetic algorithms (GA) have become popular in solving time-cost trade-off problems. Feng et al. [12] proposed a model using genetic algorithms and the Pareto front approach to solve construction time-cost trade-off problems. Leu et al. [13] proposed a GA-based fuzzy construction time-cost trade-off model, in which the effects of both uncertain activity duration and time-cost trade-off are taken into account. The abovementioned time-cost trade-off models did not address the problems regarding activity-relevant constraints, e.g., precedence relationships, resource demand and availability, interruption and overlapping of activities, etc.

Time-cost trade-off analysis is to find the optimality of a schedule, i.e., minimizing the project cost while maintaining the desired project duration. However, the resource constraints determine the feasibility of a schedule and often affect the optimality of the schedule. For example, it may be easy to shorten a project's duration by assigning more resources to the tasks, but the availability of the limited resources may not satisfy the increased resource demands. Thus, allocation of resources is the key issue for generation of feasible schedules in resource-constrained project scheduling problems.

Early attempts to solve resource allocation problems were made to find an exact optimal solution by using mathematical programming approaches [14]-[16]. However, such optimization approaches remain computationally impractical for most large projects because of an enormous number of variables and constraints. Thus, the problem is most amenable to heuristic approaches, which allow selection between activities that are competing for the use of limited resources. Heuristic methods perform well over a variety of problems, however, they still experience the drawbacks described above. Davis [17] advocated hybridizing GA with the existing problem-solving algorithms so that the domain expertise could be preserved. Many successful applications have been reported to solve resource-constrained problems by hybridizing GA with heuristic methods [18]-[19].

The aforementioned resource-constrained scheduling models mainly focused on the impact of resource constraints on project duration and assumed only one execution mode for each activity. Those models also dealt with duration-minimization problems in which activities could not be interrupted and/or overlapped.

In this paper, the authors integrated the ideas from the aforementioned research (time-cost trade-off analyses and resource scheduling) to form a new approach using a two-phase genetic algorithm (GA) to tackle resource-constrained project scheduling problems (RCPSP). In this model, a GA-based time-cost trade-off analysis is used to determine which execution mode should be selected for each activity. A set of constraints (e.g., precedence relationships, resource demands and availability, interruption and overlapping of activities, etc.) are provided as the criteria for determining the scheduling of each activity and a GA-based resource scheduling process will check the satisfaction of the constraints after the execution modes for all the activities are selected through the time-cost trade-off analysis. This model is implemented into a computer system with user-friendly interfaces. The scheduling methods will be addressed later in this paper.

II. Characteristics of Resource-Constrained Project Scheduling Problems

In general, a resource-constrained scheduling problem is defined as follows: A project consists of a set of interrelated activities. Each activity could be executed in more than one way, which can be represented using a continuous duration-resource function. Each activity could also be interrupted and/or overlapped. Resources are limited in quantity. Under these conditions, a solution which could find the optimal execution mode for each activity and properly assign the resources to the activities is to be researched to satisfy all the required constraints and produce the best time-cost combination. The following sections list the characteristics of resource-constrained project scheduling problems from two perspectives: activities and resources.

A. Activities

Shi and Deng [20] developed a generic object-oriented data structure to represent an activity. This object-oriented expression of activity can represent the information of an activity at different levels and planning stages. In this paper, the object-oriented expression of activity is adopted to show the characteristics of a general resource-constrained project-scheduling problem as shown in Eq. (1).

$$\text{Activity } \{E, D, C, P, R, O, I, S\} \qquad (1)$$

where,

E : Execution modes. Activities may have more than one execution mode. Each execution mode has its own set of resource requirements and estimated duration. For example, an excavation work may require 12 days if an average excavator is used or it may require 10 days if a more powerful excavator is used.

D : Activity duration.

C : Activity cost.

P: Precedence. Each activity has its precedence relationship. Usually an activity cannot start until all its predecessors have completed.

R : Resource demands. Resources required by each activity may be renewable or non-renewable. The costs of renewable resources are usually measured based on their hourly or daily rates. Labor and equipment are two examples of renewable resources. Non-renewable resources are mainly referred to raw materials.

O: Overlap. Although precedence relationships should be observed, overlapping is sometimes possible. Some activities can be defined to allow their successors to start, even when they are only partially completed. For example, the installation of temporary structure may begin when the excavation work is 80% completed.

I : Interruption. In a construction project, some activities have to be continuously executed once they are started, whereas some activities may be interrupted, i.e., their resources may be used by other activities for a while and then return to them. Interruption of an activity by the other activity is due to the relative importance between these two activities. In view of limited resources, preempting resources from non-essential or non-urgent ongoing activities for essential or urgent ones may shorten duration and/or smooth resource requirements. Thus, it is reasonable to assume that only ongoing activities are eligible for interruption. It is also assumed that when an interrupted activity is resumed, the working mode is kept the same for simplicity.

S : State of the activity.

A construction activity could be in one of the three states: scheduled but not started (SC), ongoing (ON) and completed (CO), i.e.,

$$S = \{SC, ON, CO\} \tag{2}$$

The precedence relationships within a construction project can be classified into four categories: FS (finish-to-start), SS (start-to-start), SF (start-to-finish) and FF (finish-to-finish). Thus,

$$T = \{FS, SS, SF, FF\} \tag{3}$$

For simplicity, only the FS (finish-to-start) relationship between activities is demonstrated in the paper. Shortly in the future, all the other three types of relationships will also be incorporated into the computer system.

Resources can be further broken down to renewable (RE) and nonrenewable (NR) resources. Renewable resources can be classified as labor (L) and equipment (E), as shown in Eq. (4).

$$R = \{RE, NR\}; RE = \{L, E\} \tag{4}$$

Furthermore, activity cost consists of resource cost (RC) and interruption cost (IC). An ongoing activity may be interrupted and restarted later, with some cost or increase in estimated time for completion. For example, the excavation work on site A can be interrupted with the excavators being moved to site B for more urgent excavation work. A cost will be incurred to reflect the cost of interrupting the current work, such as the cost for workers to familiarize the new work condition, the cost for moving the excavators to site B and so on. Correspondingly, a similar cost will be incurred to reflect the cost of resuming the interrupted work. While such cost is realistically tied to the interrupted activity, the use of interruption cost in a lump-sum value may be considered to simplify the problem. Hence the activity cost is expressed as follows:

$$C = \{RC, IC\} \tag{5}$$

Activity objects can be instantiated through object-oriented representations by supplying the information to the corresponding attributes. The information can be updated with project progress. Subsequently, the work schedule can be modified or updated according to the updated information. An example activity object is illustrated in Fig. 2.

B. Resources

In practice, resources are limited in quantity. As allocation and arrangement of renewable resources are the concern of this paper, non-renewable resources (like raw materials, etc.), which could usually be estimated as a fixed amount based on the quantities needed, are not discussed in detail here. The resource constraint pattern for renewable resources can be defined mathematically as follows:

$$(Ra_t - Rr_t) \geq 0,\ 1 \leq t \leq T \tag{6}$$

$$Rr_t = \sum_{j=1}^{A} Rr_{tj} \tag{7}$$

where t is the current time; T is the total duration; Ra_t is the resource available at time t; Rr_t is the resource required at time t; Rr_{tj} is the resource required at time t by activity j; $j \in A$, the set of activities scheduled at time t.

III. Genetic Algorithms

Genetic algorithms (GAs) are stochastic search algorithms based on the mechanisms of natural selection and genetics to search through decision space for optimal solutions [21]. In GAs, potential solutions to a problem are represented as a population of chromosomes and each chromosome stands for a possible solution. GA starts with an initial population representing a set of random solutions. The chromosomes in the population evolve through successive iterations, which are called generations. During each generation, the chromosomes are evaluated based on their performances by using some measures of fitness. To create the next generation, new chromosomes, or usually called offsprings, are formed by either merging two chromosomes from the current generation using a crossover operator or modifying a chromosome using a mutation operator. A new generation is formed by selecting some of the parents and offspring according to the fitness values and rejecting the others so as to keep the population size constant. Fitter chromosomes have higher probabilities of being selected. After several generations, the final chromosomes could usually represent the optimum or sub-optimal solutions to the problem [19]. A typical optimization procedure involving the genetic algorithm is summarized below:

```
Begin
  Generate a new population of solutions;
  While (terminating condition not met) do {
    Evaluate solutions through fitness assignment;
      Select better solutions based on fitness value;
      Recombine solutions using crossover operator;
    Mutation;
  }
End
```

The types of crossover and mutation operators used in this model are one-point crossover and uniform mutation. Crossover operates on two chromosomes at a time and generates offsprings by combining both chromosomes' feature. The one-point crossover chooses a random cut-point and generates the offsprings by exchanging the right parts of the two parent chromosomes (see Fig. 3). Besides crossover, mutation is a background operator which produces spontaneous random change in various chromosomes. Uniform mutation alters one gene in a chromosome, depending on the defined mutation rate (see Fig. 3).

IV. Two-Phase GA Model for Resource-Constrained Project Scheduling Problems

The operational architecture of the two-phase GA model for resource-constrained project scheduling problems is shown in Fig. 4. The model consists of four subsystems: the input subsystem, the time-cost trade-off subsystem, the resource scheduling subsystem and the output subsystem. All the four subsystems are described in detail in the following sections.

A. Input Subsystem

To initialize the resource-constrained project scheduling, two types of information are required: GA-related input information and project input information. The GA-related input information includes population size, crossover rate, mutation rate, maximum generation, etc. The project-related input information includes estimates of direct costs, duration and resource demands for each corresponding execution mode, state of activity, the precedence relationships between activities and other constraints and indirect cost rate for the project. The input subsystem provides interfaces for users to input or update information and stores the information data for further scheduling.

B. Time-Cost Trade-Off Subsystem

Feng et al. [12] treated construction time-cost trade-off problems as multiobjective optimization problems, which try to minimize cost and duration simultaneously, and proposed a model using genetic algorithms and the Pareto front approach to solve the problems. In this research, a GA-based multiobjective optimization technique using the Pareto approach (see Fig. 5) was adopted for the time-cost trade-off analysis. A pool of chromosomes is created to represent possible execution modes. Each gene in a chromosome represents the execution mode of its corresponding activity. Once the execution modes for each activity are decided, the corresponding activity cost, duration and resource demands will be determined. Afterwards, the execution modes will be input to the resource scheduling subsystem, which will checks the satisfactions of the constraints and produce a feasible schedule. As a result, the total project duration and cost for each execution-mode chromosome are fed back to the time-cost trade-off subsystem for evaluation. The subsystem uses one-point crossover and random mutation operators to generate feasible child chromosomes. Note that the execution modes for ongoing activities do not participate in mutation, i.e., the execution mode for an ongoing activity can not be changed. According to the objective functions described below, the fitness for each chromosome is calculated. The surviving chromosomes for the next generation are selected according to the roulette wheel principle. This means that the selection possibility for a chromosome i is proportional to the ratio of $\frac{1}{f_i} / \sum_{j=1}^{pop\_size} \frac{1}{f_j}$, where f_i is the fitness value of chromosome i. Note that fitness is to be minimized here, i.e., small fitness values correspond to high selection probabilities. The elitist selection method is combined with the selection procedure to preserve the best chromosomes for the next generation, thus all the nondominated chromosomes are automatically selected into the next population to overcome the stochastic error of sampling. In the final step of time-cost trade-off subsystem, the optimal or sub-optimal solution is exported to the output subsystem.

To calculate the fitness values of the chromosomes in the population, the following steps were used:

- According to the concept of the Pareto optimum, given any two chromosomes A, B of the population, A is said to dominate B if and only if the cost of A is less than or equal to the cost of B and the duration of A is less than or equal to the duration of B; and A is better in one objective, i.e., either the cost is less or the duration is less [see Fig. 5(b)].
- A chromosome A is denoted as nondominated regarding a given population if and only if no chromosome of the population dominates A. The set of all nondominated chromosomes are so-called Pareto front [see Fig 5(a)]. A trade-off curve (the Pareto front) can therefore be determined since there are no members in the population that have better objective values in both time and cost than the members in the nondominated set.

- For those nondominated chromosomes, the fitness is calculated as $f(i) = \dfrac{m}{N+1}$, where *m* is the number of chromosomes that chromosome *i* dominates.
- For other chromosomes, the fitness is calculated as $f(i) = 1 + n$, where *n* is the sum of all the fitness values of the nondominated chromosomes that dominate chromosome *i*.

C. Resource Scheduling Subsystem

Davis [17] advocates hybridizing the GA with the existing problem-solving algorithms so that the domain expertise will be preserved. Many successful applications have been reported in the domain of resource scheduling [18]-[19]. The GA performs the basic GA processes of selection, recombination and mutation on succeeding populations of solutions, while the evaluation function for the resource allocation problem is supplemented with conventional heuristic methods. In Gen and Cheng's [19] approach, the topological-based heuristics, i.e., the heuristics based on the precedence between activities in the schedule, are used. Interested readers can refer to it for more detailed information. In this approach, priority for each activity is firstly generated by GA. The activities are then sorted using precedence relationships as the criterion. Activities of the same topological rank are then mapped to a contiguous segment of the chromosome. In the segment where genes have the same topological rank, the genes are sorted from high priority to low priority based on the priorities assigned to the activities. Thus, the precedence constraint is met. Because a topological sort gives a feasible order of activities, a schedule can be constructed by selecting activities in the order of their appearance in the topological sort and scheduling them one at a time as early as resource availability allows.

In this research, the Gen and Cheng's [19] approach was employed for resource allocation, but with some modifications as follows:

- Activities which have started are updated with remaining duration, cost, and nonrenewable resource requirements (raw materials). It is reasonable to assume that renewable resources (labor and equipment) demands remain unchanged.
- Activities which have started and required continuity must be assigned with highest priorities so that resources can be allocated to such activities first. For example, if there are *m* ongoing activities that can not be interrupted among total *n* activities, then priority from *n* to *n*-*m*+1 will be randomly assigned to those activities, where *n* represents the top priority. Activities which have started and allow interruption will be treated as non-started activities.
- Activities whose predecessors allow overlapping can start earlier according to the overlap time allowed.

The flow of the procedure is shown in Fig. 6. In the resource scheduling subsystem, each gene in a chromosome represents the priority of its corresponding activity. The subsystem uses one-point crossover and uniform mutation operators. The scheduled project duration is taken as the fitness value of each chromosome and the roulette wheel approach is adopted as the selection procedure. The elitist selection is incorporated as well.

D. Output Subsystem

In the output subsystem, all the project total costs and their corresponding durations are gathered for further plotting. Subsequently, the trade-off curve between direct cost and duration is extracted from the final generation. After finding the trade-off curve, construction planners can determine the total cost by summing up the estimated indirect cost and the direct cost from the trade-off curve. Indirect cost is usually assumed to be proportional to the project duration. The optimal choice to perform the project would be the lowest total cost. Using trade-off curve as the objective function allows for much more efficient evaluations of various indirect cost rates without performing another GA run. This is an improvement over treating the total cost as the objective in the GA. What's more, the feasible schedule and resource profile for corresponding combination of project duration and cost can be obtained as well.

V. Example Project

In this section, a simple project is planned with the network shown in Fig. 7 and activity data in Table 1. The GAs parameters, such as crossover rate and mutation rate, are tuned through experimental tests to obtain good performance using GAs search, as tabulated in Table 2.

To illustrate the influences of constraints (e.g., resources constraints, interruption or/and overlap of activities) over the time-cost optimization, final generation of the GA calculation, optimal time-cost trade-off curve, total cost curve and the schedule for the optimal choice were plotted for the three types of conditions described below respectively (see Figs. 8-19).

A. Planning Project with Precedence Relationships Only

If only precedence relationships are active, the project duration and cost generated by resource scheduling subsystem is comparable to the one obtained from CPM computation. The optimal project total cost is $227,700 with project duration of 49 days (see Figs.8-11).

B. Adding Resources Capacities into Active Constraints

If both precedence relationships and resource capacities are selected as the active constraints, the resource-constrained time-cost trade-off and the total cost curve were plotted in Fig.13 and 14 respectively. Due to the limitation of the resource availability, it can be seen that optimal project total cost is increased to $244,000 with optimum project duration postponed to 56 days (see Figs. 12-15).

C. Allowing Interruption for Ongoing Activities and Overlaps of Selected Activities

For the example project described above, we assume that only activities 1 and 3 are started when the project has been commenced for one day. Activities 4, 8 and 9 allow overlap by their successors. The resulting resource-constrained time-cost trade-off and total cost curve were plotted in Fig. 17 and 18 respectively. As the result of the interruption and/or overlap of activities, it can be seen that optimal project total cost is reduced to $237,300 though the optimum project duration remains at 56 days. In Fig. 19, it can be seen that the ongoing activity 1 was interrupted to make more critical activity 2 be executed earlier and activity 10 overlapped its predecessor (activity 9) by 1 day.

VI. CONCLUSIONS

This paper has presented a two-phase GAs model for resource-constrained project scheduling, which incorporated GA-based computational techniques for time-cost trade-off and resource scheduling. A set of constraints are provided in the model for project scheduling problems. If only precedence relationships are selected, the time-cost trade-off result is comparable to the work done by Feng et al. [12]. On the other hand, if more project constraints are applied, a corresponding practical time-cost trade-off result can be generated using this model. Allocations of limited resources, interruption and/or overlap of activities are allowed in the scheduling process.

It is fair to point out that GAs share one drawback with heuristic methods: it is not possible to know if an optimal result has been obtained. Some existing algorithms (e.g., mathematical methods) can solve some forms of scheduling problems, but fail to do so when the activity number increases or more constraints are added. Hence, using GAs to solve project scheduling problems is a good way to find optimal or near-optimal solutions. An example is shown in this paper to demonstrate the implementation of the two-phase GA model in a computer system.

REFERENCES

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

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

[3] S. E. Elmagrhraby, "Resource allocation via dynamic programming in activity networks." *Euro. J. Oper. Res.,* 64, pp. 199-215, 1993.

[4] C. Henderickson and T. Au, *Project management for Construction*, Prentice-Hall, Inc., Englewood Cliffs, N.J, 1989.

[5] J. E. Kelly, "Critical path planning and scheduling: mathematical basis." *Oper. Res.,* 9(3), pp. 167-179, 1961.

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

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

[8] D. Panagiotakopouols, "Cost-time model for large CPM project networks." *J. Constr. Engrg. and Mgmt.,* ASCE, 103(C02), pp. 201-211, 1977.

[9] J. W. Fondahl, "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.

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

[11] O. Moselhi, "Schedule compression using the direct stiffness method." *Can. J. Civ. Engrg.,Ottawa,* 20(1), pp. 65-72, 1993.

[12] C. Feng, L. Liu, and S. Burns, "Using genetic algorithms to solve construction time-cost trade-off problems." *J. Comp. Civ. Engrg.* , ASCE, 11(3), pp. 184-189, 1997.

[13] S. S. Leu, A. T. Chen, and C. H. Yang, "A GA-based fuzzy optimal model for construction time-cost trade-off." *Int. J. Pro. Mgmt.,* 19(1), pp. 47-58, 2001.

[14] H. Adeli and A. Karim, "Scheduling/cost optimization and neural dynamics model for construction." *J. Constr. Engrg. and Mgmt.*, ASCE, 123(4), pp. 450-458, 1997.

[15] E. Davis and G. Heidorn, "An algorithm for optimal project scheduling under multiple resource constraints." *Mgmt. Sci.,* 17, pp. 803-816, 1971.

[16] A. Pritsker, L. Waters, and P. Wolfe, "Multiproject scheduling with limited resources: a 0-1 approach." *Mgmt. Sci.,* 16, pp. 93-108, 1969.

[17] L. Davis, *A handbook of genetic algorithms.* Van Nostrand Reinhold, New York, N.Y, 1991.

[18] W. T. Chan, D. K. H. Chua, and G. Kannan, "Construction resource scheduling with genetic algorithms." *J. Constr. Engrg. and Mgmt*, ASCE, 122(2), pp. 125–132, 1996.

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

[20] J. Shi and Z. M. Deng, "Object-oriented resource-based planning method (ORPM) for construction." *Int. J. Pro. Mgmt.*, 18(3), pp. 179-188, 2000.

[21] D. E. Goldberg, *Genetic Algorithm in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Co., Inc., Reading, Mass, 1989.

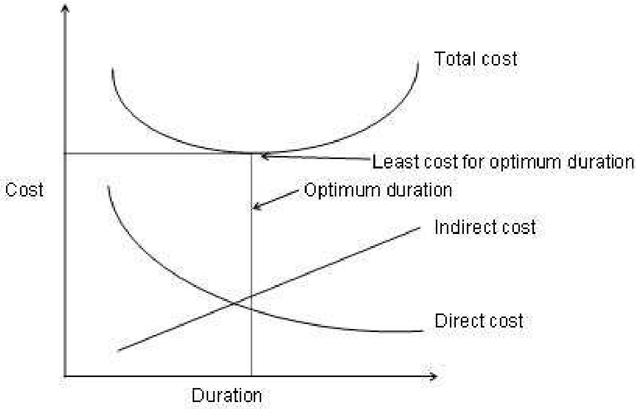

Fig. 1. Optimal choice of a construction project.

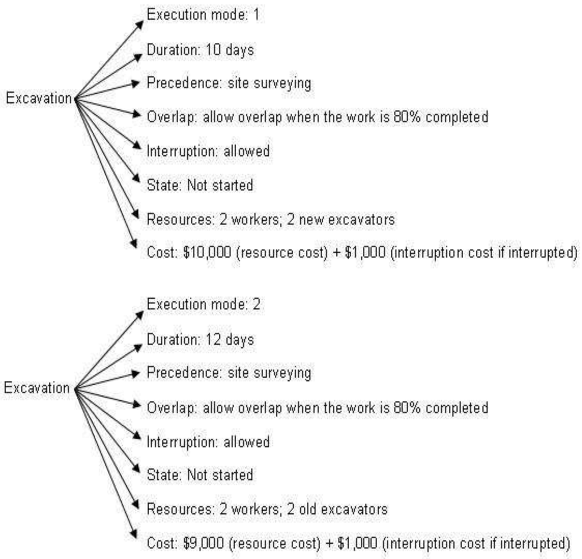
Execution mode: 1
Duration: 10 days
Excavation
Precedence: site surveying
Overlap: allow overlap when the work is 80% completed
Interruption: allowed
State: Not started
Resources: 2 workers; 2 new excavators
Cost: $10,000 (resource cost) + $1,000 (interruption cost if interrupted)
Execution mode: 2
Duration: 12 days
Excavation
Precedence: site surveying
Overlap: allow overlap when the work is 80% completed
Interruption: allowed
State: Not started
Resources: 2 workers; 2 old excavators
Cost: $9,000 (resource cost) + $1,000 (interruption cost if interrupted)

Fig. 2. Example activity object.

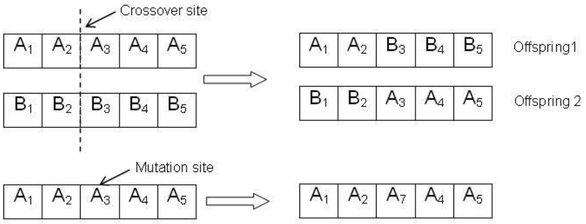

Fig. 3. One-point crossover and uniform

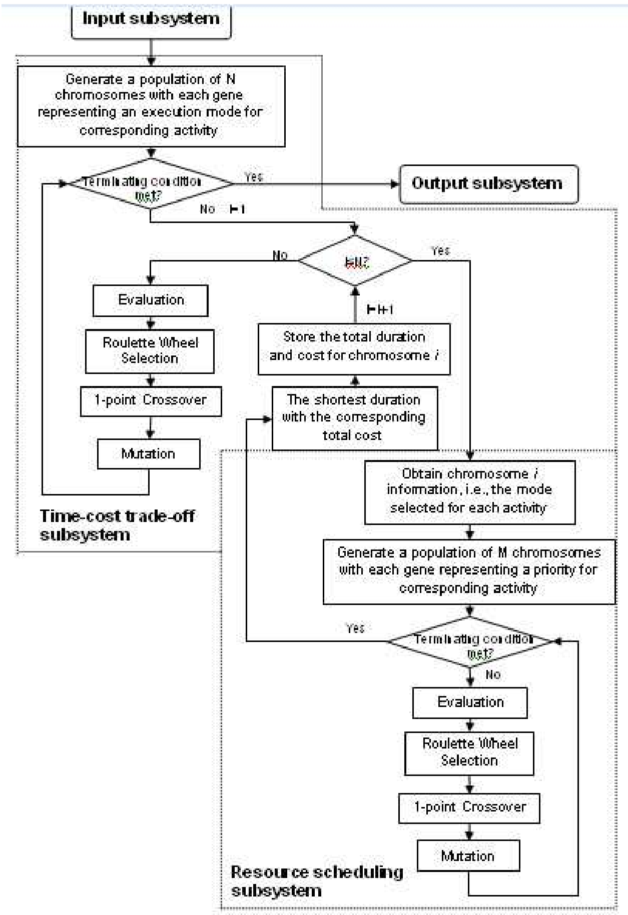
Input subsystem
Generate a population of N chromosomes with each gene representing an execution mode for corresponding activity
Terminating condition met?
Yes
No
Output subsystem
No
i≤N?
Yes
Evaluation
Roulette Wheel Selection
1-point Crossover
Mutation
i=i+1
Store the total duration and cost for chromosome i
The shortest duration with the corresponding total cost
Obtain chromosome i information, i.e., the mode selected for each activity
Time-cost trade-off subsystem
Generate a population of M chromosomes with each gene representing a priority for corresponding activity
Yes
Terminating condition met?
No
Evaluation
Roulette Wheel Selection
1-point Crossover
Mutation
Resource scheduling subsystem

Fig. 4. Operation of two-phase GAs model.

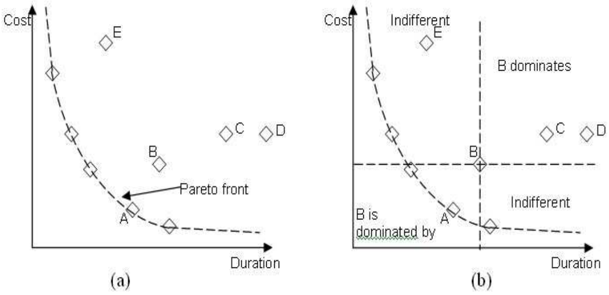

Fig. 5. (a) Pareto optimality in objective space; (b) Possible relations of solutions in objective space

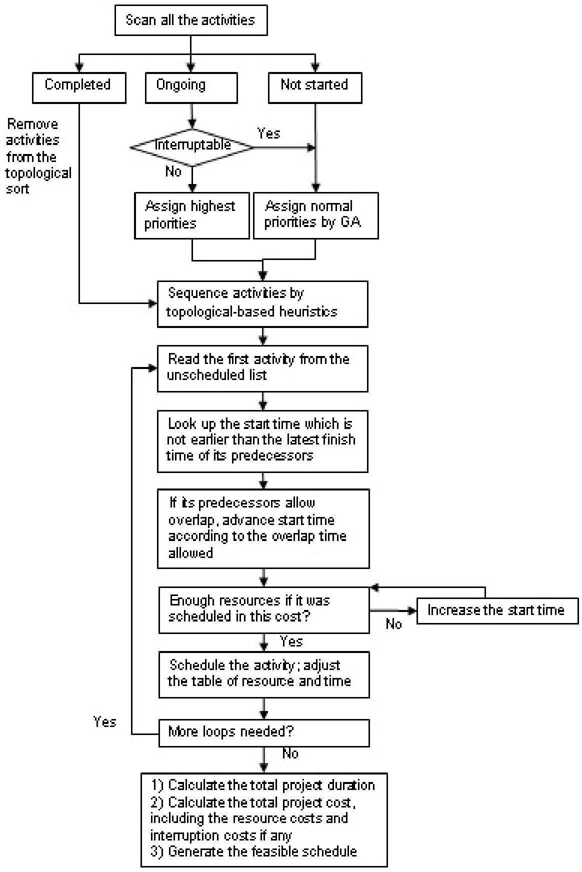
Scan all the activities
Completed
Ongoing
Not started
Remove activities from the topological sort
Interruptable
Yes
No
Assign highest priorities
Assign normal priorities by GA
Sequence activities by topological-based heuristics
Read the first activity from the unscheduled list
Look up the start time which is not earlier than the latest finish time of its predecessors
If its predecessors allow overlap, advance start time according to the overlap time allowed
Enough resources if it was scheduled in this cost?
No
Increase the start time
Yes
Schedule the activity; adjust the table of resource and time
Yes
More loops needed?
No
1) Calculate the total project duration
2) Calculate the total project cost, including the resource costs and interruption costs if any
3) Generate the feasible schedule

Fig. 6. The flow of procedure in resource scheduling subsystem

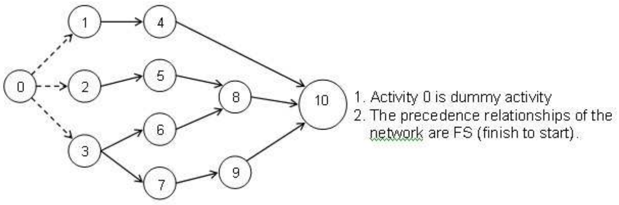
0
1
2
3
4
5
6
7
8
9
10
1. Activity 0 is dummy activity
2. The precedence relationships of the network are FS (finish to start).

Fig. 7. The network of example project.

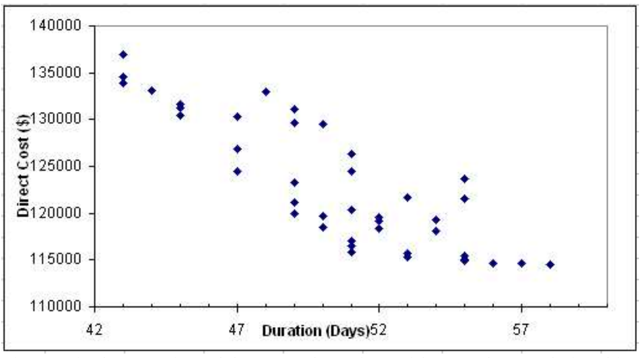

Fig. 8. Final generation of the example project with precedence relationships

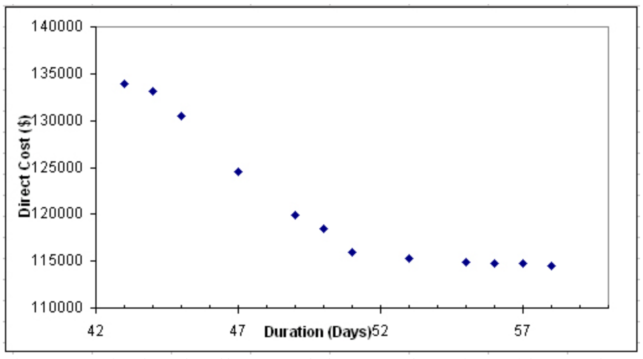

Fig. 9. Optimal trade-off curve of the example project with precedence relationships only

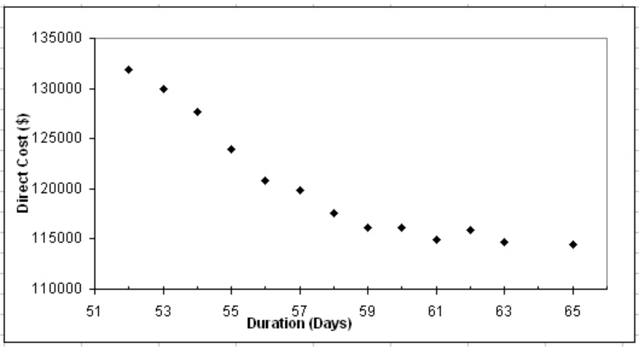

Fig. 13. Optimal trade-off curve of the example project under resource constraint

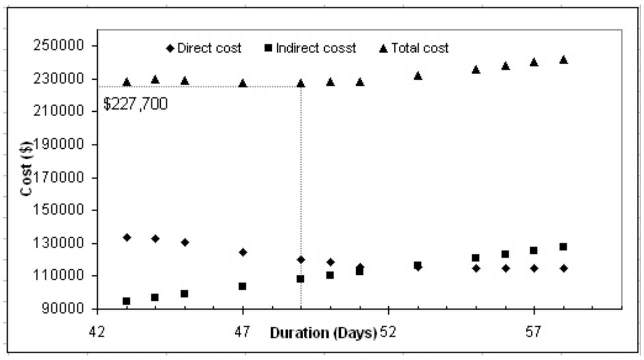

Fig. 10. Optimal choice of the example project with precedence relationships only

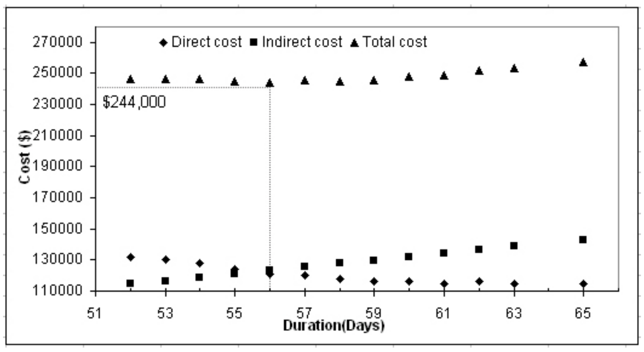

Fig. 14. Optimal choice of the example project under resource constraint

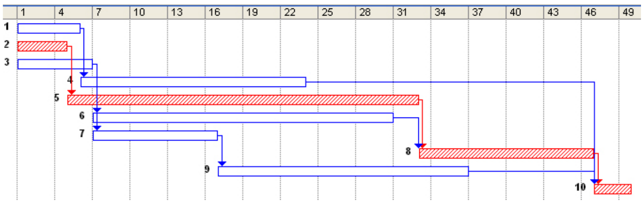

Fig. 11. Schedule for duration of 49 days if planning project with precedence relationships only (hatched blocks stand for critical path)

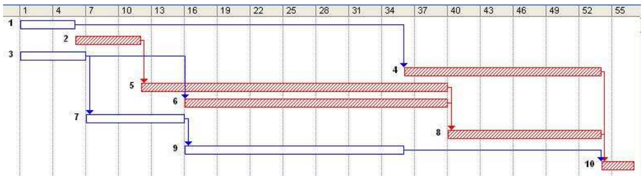

Fig. 15. Schedule for duration of 56 days if adding resource capacities into active constraints (hatched blocks stand for critical path)

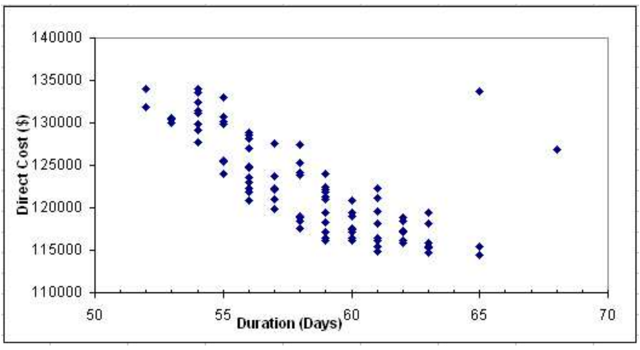

Fig. 12. Final generation of the example project under resource constraint

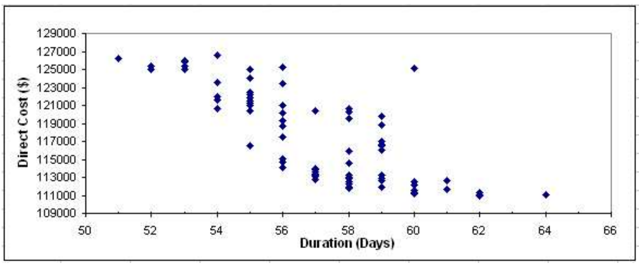

Fig. 16. Final generation of the example project with all the constraints activated

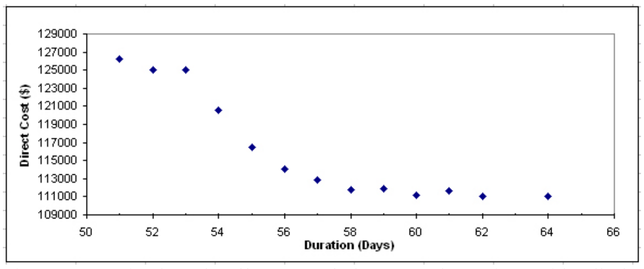

Fig. 17. Optimal trade-off curve of the example project with all the constraints activated

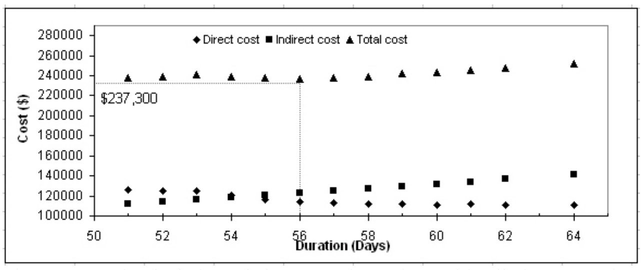

Fig. 18. Optimal choice of the example project with all the constraints activated

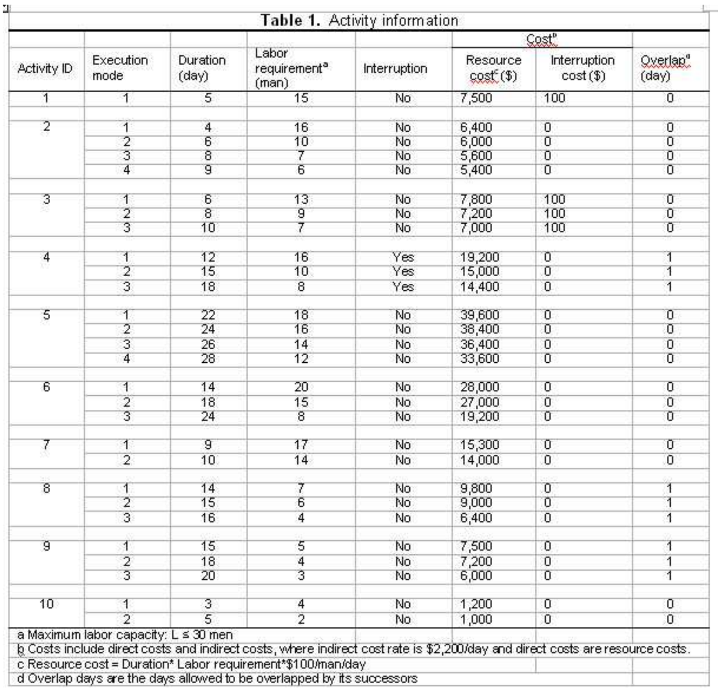

Table 1. Activity information							
					Cost^b		
Activity ID	Execution mode	Duration (day)	Labor requirement^a (man)	Interruption	Resource cost^c ($)	Interruption cost ($)	Overlap^d (day)
1	1	5	15	No	7,500	100	0
2	1	4	16	No	6,400	0	0
	2	6	10	No	6,000	0	0
	3	8	7	No	5,600	0	0
	4	9	6	No	5,400	0	0
3	1	6	13	No	7,800	100	0
	2	8	9	No	7,200	100	0
	3	10	7	No	7,000	100	0
4	1	12	16	Yes	19,200	0	1
	2	15	10	Yes	15,000	0	1
	3	18	8	Yes	14,400	0	1
5	1	22	18	No	39,600	0	0
	2	24	16	No	38,400	0	0
	3	26	14	No	36,400	0	0
	4	28	12	No	33,600	0	0
6	1	14	20	No	28,000	0	0
	2	18	15	No	27,000	0	0
	3	24	8	No	19,200	0	0
7	1	9	17	No	15,300	0	0
	2	10	14	No	14,000	0	0
8	1	14	7	No	9,800	0	1
	2	15	6	No	9,000	0	1
	3	16	4	No	6,400	0	1
9	1	15	5	No	7,500	0	1
	2	18	4	No	7,200	0	1
	3	20	3	No	6,000	0	1
10	1	3	4	No	1,200	0	0
	2	5	2	No	1,000	0	0

a Maximum labor capacity: L ≤ 30 men

b Costs include direct costs and indirect costs, where indirect cost rate is $2,200/day and direct costs are resource costs.

c Resource cost = Duration* Labor requirement*$100/man/day

d Overlap days are the days allowed to be overlapped by its successors

Table. 1. Activity information

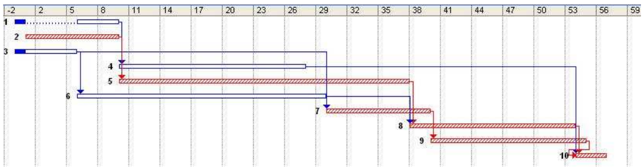

Fig. 19. Schedule for duration of 56 days if activating all the constraints (hatched blocks stand for critical path and solid blocks stand for completed part of the activities)

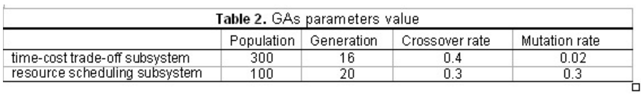

Table 2. GAs parameters value				
	Population	Generation	Crossover rate	Mutation rate
time-cost trade-off subsystem	300	16	0.4	0.02
resource scheduling subsystem	100	20	0.3	0.3

Table. 2. Gas parameter value