DEVELOPMENT OF A RESOURCE SCHEDULING MODEL USING OPTIMIZATION

Ralph D. Ellis¹, and Jin-Lee Kim²

ABSTRACT

The CPM has proven to be helpful only when the project deadline is not fixed and the resources are not constrained by either availability or time. Resource-constrained scheduling arises as a result of problems with the availability of resources. It is necessary to apply the availability of resources into a scheduling process for this reason. The objective of this paper is to present an optimal algorithm for a resource allocation model, which would be implemented into a framework for the development of an integration model. This framework for the development of an integration model for the resource-constrained scheduling is discussed.

An optimal resource allocation model is presented within the framework of an integration model. Unlike present heuristic-based resource allocation models, the model does not depend on any set of heuristic rules, but adopts the concept of future float to set the order of priority when activities compete for resources. The model determines the shortest duration by allocating available resources to a set of activities simultaneously. Genetic algorithms (GAs) are adopted to search near-optimal solutions. A case example is solved to illustrate the performance and effectiveness of the resource allocation model. The results indicate that the model is capable of producing near-optimal scheduling alternatives, compared to a single solution that is produced by either the total float model or the least impact model.

KEY WORDS

Resource, Scheduling, Resource allocation, Project management, Optimization.

INTRODUCTION

The Critical Path Method (CPM) has been used widely in the construction industry. The CPM generates useful information for the project manager to plan and control the project more actively and efficiently. Critical and non-critical activities are particularly important for the purpose of project management on large, complex projects. However, the CPM has proven to be helpful only when the project deadline is not fixed and the resources are not constrained by either availability or time. Therefore, resource-constrained scheduling arises as a result of problems with the availability of resources. It is essential to apply the availability of resources into a scheduling process for this reason.

¹ Associate Professor, Department of Civil and Coastal Engineering, 460-B Weil Hall, University of Florida, Gainesville, FL 32611, Phone +1 352/392-9537 x1485, FAX 352/392-7983, relli@ce.ufl.edu
² Graduate Research Assistant, Ph.D. Candidate, Dept. of Civil & Coastal Engineering, 460 Weil Hall, Univ. of Florida, Gainesville, FL 32611, Phone +1 352/392-9537 x1531, FAX 352/392-7983, jin5176@ufl.edu
The CPM technique makes use of a project graph to schedule large projects with the assumption that there is an unlimited availability of resources. When the availability of resources is limited, the concepts of critical path and total float to the CPM technique lose their original meaning. If some activities on a critical path are delayed due to insufficient resources, then the critical path that has a zero total float no longer exists. If there is a systematic way to consider both resource dependencies and precedence relationships, resource-constrained scheduling for large projects can be resolved.

Traditionally, resource-constrained scheduling has been classified into three categories: (1) resource allocation, (2) time-cost trade-off analysis, and (3) resource leveling (Weist and Levy 1969; Moder et al. 1983; Moselhi and Lorterapong 1993; Hegazy 1999). These problems have been solved with either heuristic or optimal procedures. Heuristic procedures employ some rule of thumb or experience to determine priorities among activities competing for available resources. Optimal procedures, also called analytical procedures, employ some form of mathematical programming or other analytical procedure to search for the best possible solutions.

The heuristic procedures generally produce solutions for the resource-constrained scheduling problems in a reasonable amount of time, even though the size of the project network is large. However, they have proven to be inconsistent with regard to the quality of results produced on project networks (Easa 1989; Boctor 1990; Hegazy 1999).

Relative to the vast amount of research that has been conducted on heuristic procedures, optimal procedures have rarely been the focus of such extensive research. Considerable progress has been made to produce optimal results by depending on strong assumptions for small-sized project networks. However, no optimal procedures have proven to be computationally feasible for large, complex projects that can occur in practice. Therefore, it is important to develop a more efficient algorithm that searches near-optimal solutions for the resource-constrained scheduling problems for large-sized project networks in a reasonable amount of time. A developed algorithm should consider both the availability of resources and precedence relationships.

Various models have been generated to solve problems within each class of the three resource-constrained scheduling problems. Each class of problem has its own algorithm for solutions because of the different objectives and constraints. Thus, algorithms designed for resource allocation are not applicable to resource leveling (Hegazy 1999). In addition, a project manager desires to satisfy both objectives, causing a conflict between the availability of resources and project durations.

On account of the inherent complexities of construction projects and the difficulties with applying the resource-constrained scheduling problems into a model, little research has been done to integrate solutions to these problems. Therefore, it is necessary to develop an efficient, optimal model that integrates solutions for the resource-constrained scheduling problems in order to provide a powerful tool for a project manager.

The objective of this paper is to present an optimal algorithm for a resource allocation model, which would be implemented into a framework for the development of an integration model. This framework for the development of an integration model for the resource-constrained scheduling is discussed. An application of the model using a case study is illustrated.
INTEGRATION OF RESOURCE SCHEDULING USING GAS

Genetic algorithms (GAs) have been used to solve construction management problems, including resource scheduling, time-cost trade-off optimization, and combined resource allocation and leveling. Unfortunately, the research on these topics focuses on a single objective. Also, the research does not present a method to obtain near-optimal solutions within a reasonable amount of time, which is the main disadvantage of GAs. Therefore, it is necessary to develop a near-optimal model that integrates resource scheduling problems, by simultaneously considering multiple objectives as well as reducing computation time.

Some research has been conducted to integrate the resource scheduling problems. Chan et al. (1996) combined resource allocation and leveling by setting a single equation with the objective of minimizing the difference between resource availability and utilization. In the string representation of GAs, they used the concept of current float to set the scheduling priority. The current float concept (Shamuganayagam 1989) was introduced to eliminate network recalculation, which is the main disadvantage of the total float concept in the CPM analysis. The model establishes the order of priority for the relevant activities based on their current floats and allocates resources to the activity that has the smallest current float under the availability of resources.

Hegazy (1999) combined resource allocation and leveling using GAs. The research incorporated the concept of minimum total slack as the decision variable for resource allocation, and minimum moment method for resource leveling, respectively. Although an effective improvement was made in the combined model, it has been stated clearly that it is necessary to integrate a time-cost trade-off analysis into the GA approach.

Leu and Yang (1999) proposed a GA-based multicriteria optimal model for construction scheduling. Researchers unified time-cost trade-off and resource allocation. They used activity duration obtained from the time-cost trade-off model as basic input data for the computation of minimum project duration under resource constraints. However, they mentioned that improvements in resource leveling need to be made because resource conflicts still occurred.

In the area of optimization, there is a need for ongoing research concerning the integration of resource scheduling problems, so that the integration can be applied in practice. There are recent trends to combine these three problems, even though this integration is neither easy nor intuitive. Therefore, this research team proposed a model that integrates resource scheduling problems. The framework of the model is described in the following sections.

PROPOSED INTEGRATION MODEL

The model proposed for integrating resource scheduling problems is a stepwise model as Figure 1 illustrates. The integration model starts with the problem formulation that establishes the objective function and constraints for each problem. Users may provide network information, such as the activity duration, activity cost options, precedence relationships, resource availability, and activity resource requirements. This information is required because they are project specific. The GA parameters, such as the number of
experiments and trials to conduct, population size, and probabilities of genetic operators are also provided by the user.

Figure 1: A framework for integration model

Module A is the resource-constrained allocation model, which is the focus of the following section. In order to develop the model, this research team chose the concept of future float to set the scheduling priority, which has yet to be accomplished in the area of GA-based scheduling. The objective of the resource-constrained allocation model is to allocate the available resources to activities so as to find the shortest duration of a project. Precedence relationships and the availability of resources are constrained in this model. Near-optimal project duration is obtained. Schedule and resource requirements are sent to Module B, time-cost trade-off model.

Module B is the time-cost trade-off model, the objective of which is to find an optimal point between the normal activity time-cost point and the crash activity time-cost point. Thus, the project duration is shortened with minimum costs. A time-cost trade-off curve that depicts the relationships between project duration and cost is created for the economic feasibility analysis of a project network plan. In Module B, the GA-Crashing simulator is to be developed in the following phase of the ongoing research.

Module C is the resource leveling model, the objective of which is to minimize fluctuation in patterns of resource usage within the required duration. A histogram showing the variation in the resource profile is created as an output of Module C. In Module C, the GA-Profile generator is to be developed in the following phase of the ongoing research.

Multiobjective optimization problems (MOP) arise in many real-world decision making situations. Solutions to an MOP need to be optimized simultaneously in GAs like in other optimization techniques. However, no clear definition of an optimum exists in an MOP, even
though an optimal solution can be obtained in the case of single objective optimization problems, such as resource allocation, time-cost trade-off, or resource leveling.

The common difficulty with an MOP is that the objectives of project cost, project duration, and resource utilization may conflict with each other when the proposed model integrates the resource-constrained scheduling problems. If there is at least one optimal solution in the resource-constrained scheduling problems, then the concept of non-dominated solutions arises in the other objectives. Therefore, it is important to search non-dominated solutions in order to satisfy the criteria of multiobjective resource-constrained scheduling problems.

**RESOURCE-CONSTRAINED ALLOCATION MODEL (MODULE A IN FIG. 1)**

The resource-constrained allocation problem aims to allocate the available resources to activities so as to find the shortest duration of a project within the constraints of precedence relationships. The assumptions underlying this problem are that the availability of resources is constrained to some maximum value, and that the project has to be completed using the given resources. As a result of the resource-constrained allocation, a schedule that shows the shortest duration with resource limits is created for a project network.

The additional costs directly associated with utilization of allocated resources are undesirable for the contractor as they occur in the following problems (Popescu and Charoenngam 1995):

- Cost of hiring or requiring additional resources,
- Cost of idle labor or equipment, and
- Cost of delay to other activities not having enough resources to schedule the work.

**PROBLEM FORMULATION**

The resource-constrained allocation problem has gained importance from the fact that it is a “combinatorial” problem, which includes job-shop sequencing and assembly-line balancing problems (Davis 1973). This is also well known as one type of Nondeterministic Polynomial (NP)-hard problem. This type of problem is characterized by a factorial growth in the amount of computation required to consider all possible solutions as problem size increases (Davis 1973).

Let $L$ and $k$ be the number of resource combinations and the number of activities, respectively. The size of the search space is $L^k$. It takes $L^k$ tries to search optimal solutions by exhaustive enumeration. For instance, a project has 30 activities and each activity has 2 resource combinations. It will take $2^{30}$ (equal to 1,073,741,824, that is up to $10^9$) tries to find the optimal solution. In the end, computational efforts increase exponentially as either the number of resource combinations or the number of activities increases.

Davis (1973) describes resource allocation as “the method of scheduling activities within fixed amounts of resources available during each time period of project duration so as to minimize the increase in project duration.”

The resource-constrained allocation problem solved in this paper may be stated as follows, as summarized by Moselhi and Lorterapong (1991):
A finite set of activities is given,

Each activity has a fixed duration, requires a fixed amount of one or more different types of resources, is subjected to a set of precedence relationships,

An activity can start only when its precedence constraints are met, and

All of its required resources are available throughout its duration, or, if desired, could be split. In other words, splitting could be achieved by breaking down the activity into a group of sequential subactivities.

The objective of the resource-constrained allocation model is to minimize the project duration when constrained by precedence relationships among project activities and the availability of resources. It is assumed that daily resource requirements of any activity are predetermined. The sum of all available resource units allocated to different activities on the same day should not exceed the total number of units available on that day.

**GENETIC ALGORITHMS PROCEDURE FOR RESOURCE ALLOCATION MODEL**

The simplified procedure of GAs that is used in this paper is as follows; (1) Define a solution representation, (2) Set variables, objective functions, and constraints, (3) Generate initial population, (4) Evaluate fitness of possible solution, (5) Apply genetic operators, and (6) Test termination conditions. This procedure was modified from Sakawa’s (2002) fundamental procedures of GAs.

**Step 1: Defining a solution representation**

A solution is represented by the set of values associated with the problem variable. In the resource-constrained allocation model, the activities that have a scheduling priority are the set of values, and the number of variables is the number of activities to be scheduled. GAs arrange that solution in a string as shown in Figure 2.

![Figure 2: Genetic algorithm representation of solution](image)

The priorities of activities are assigned to each feasible activity set based on its minimal negative impact on the remaining total float of the entire network of the project (Moselhi and Lorterapong 1993). They have proposed the least impact model and used two terms: future float (FF) and next time frame (NTF). The future float for an activity is defined as “the finish
float (Shanmuganayagam 1989) available to the activity if it will be postponed to the next time frame.” The next time frame coincides with the earliest finish time of all activities within the set being considered.

**Step 2: Setting the variable, objective functions, and constraints**

Variables are activities that have the priority of getting resources, and they are the values inside each feature, as shown in the string structure of Figure 2. A single objective function to evaluate each string is to minimize the project duration of a project network.

The sum of all available resource units allocated to different activities on the same day should not exceed the total number of units available on that day.

**Step 3: Generating initial population of solutions (strings)**

An initial population is generally created at random to apply GAs, as is done in most research (Bean 1992; Chan et al. 1996; Leu and Yang 1999; Hegazy 1999).

**Step 4: Evaluating fitness of possible solutions**

The fitness of each string in the population obtained in Step 3 is evaluated by the value calculated when the fitness of string \( i \) is divided by sum of total fitness of all strings (Whitley 1993).

**Step 5: Applying genetic operators**

First, reproduction measures the fitness of strings in a generation. Some of the strings are reproduced in proportion to their objective function values. The aim of reproduction is to provide good solutions with a higher chance of passing their feature to the next generation than bad ones. Reproduction also increases or decreases the number of offspring for each string in the population according to the fitness values.

![Figure 3: Genetic operators – Crossover and Mutation](image)

Second, crossover selects two distinct strings from the population at random. It exchanges some portion of the strings with a probability equal to the crossover rate in order to create a new offspring. One of the simplest ways is to choose some crossover point randomly. Everything before this point is copied from a first parent and everything after a crossover point is copied from a second parent (See Figure 3).
Finally, mutation takes place after a crossover is performed. Mutation is a random change of features in a string to reintroduce lost bit values into a population. It alters one or more features of a selected string with a probability equal to the mutation rate. Mutation changes the new offspring at random (See Figure 3).

**Step 6: Testing termination conditions**

An algorithm for the resource-constrained allocation model is terminated when it meets conditions. The shortest duration under the availability of resources is the termination condition. Also, a solution that has the shortest duration is considered a near-optimal solution.

**Cast Study**

This research team chose a simple six-activity network that was derived from Moselhi and Lorterapong (1993), as shown in Figure 4. For the sake of simplicity, the network is assumed to have two types of resources and each activity can consume only one type of resources. Durations and required resources are indicated directly on the network.

![Figure 4: Arrow diagram for case study (After Moselhi and Lorterapong 1993)](image)

**Comparison with Heuristic Method**

If the availability of resources were not considered, the project duration would be 10 days. The results for the resource allocation model using the total float model, the least impact model, and the GA-based model are shown graphically in Figure 5, respectively.

First, the total float model assigns the priority to activity 3-5 with a total float value of zero, as opposed to activity 1-2 with a total float value of 2 days. As a result, the project duration obtained using the total float model is 20 days, which is 8 days longer than that obtained using both the least impact model and the GA-based model (See Figure 5 (a)).

Second, the least impact model assigns the priority to activity 1-2 because it has a larger future float value than that of activity 3-5 (See Figure 5 (b)). The schedule obtained using this model is considered the optimal duration in this paper. The project duration for Figure 5 (b) and (c) is 12 days.

Finally, the GA-based model shows two different schedules in comparison to the single schedule generated by either the total float model or the least impact model. One is shown in Figure 5 (b), which is same as the schedule obtained using the least impact model, and
another is shown in Figure 5 (c) with variation in the start time of activity 2-4 and activity 2-5. The alternate start times of activity 2-4 and activity 2-5 are insignificant because they require the same number and type of resources under the assumption that there is no technological constraint.

Figure 5: Comparison of resource allocation histograms using (a) the total float model, (b) the least impact model (After Moselhi and Longterapong 1993), and (c) the GA-based model. Therefore, it is notable that the GA-based model is able to provide near-optimal scheduling alternatives, compared to a single solution by either the total float model or the least impact model. The GA-based model has the potential to reduce computation time because of faster programming and central processing unit (CPU) improvements.

CONCLUSIONS
This paper presents an optimal algorithm for a resource allocation model, which would be implemented into a framework for the development of an integration model. This framework for the development of an integration model for the resource-constrained scheduling is discussed. Its application using a case study was illustrated.
By making use of the findings of the integration model, this research can benefit both a project manager making decisions with regard to their resource management, and researchers applying the availability of resources into a scheduling process. More work is being carried out to expand the model by integrating both time-cost trade-off and resource leveling. Future work could examine the performance of the integration model by reducing computational time.

REFERENCES


