A Multi-objective Scheduling Model for Solving the Resource-constrained Project Scheduling and Resource Leveling Problems

Jia Hu\textsuperscript{1} and Ian Flood\textsuperscript{2}

\textsuperscript{1} Ph.D. student, Rinker School of Building Construction, University of Florida, Gainesville, FL 32611-5703; PH (352) 273-1175; email: hujia@ufl.edu
\textsuperscript{2} Professor, Rinker School of Building Construction, University of Florida, Gainesville, FL 32611-5703; PH (352) 273-1159; email: flood@ufl.edu

ABSTRACT

The project resource leveling problem was proposed to smooth resource usage and reduce resource fluctuation, while the resource-constrained project scheduling problem focuses on minimizing the project duration with limited resources. Solving the two interrelated problems separately is unlikely to result in a globally optimum schedule. The few integrated optimization methods that have been developed favor a predetermined resource profile that may be difficult to achieve. This paper proposes an integrated scheduling method to minimize the project duration and resource fluctuation by using the strength Pareto evolutionary approach II (SPEA II) which outperformed several other multi-objective optimization techniques in solving the resource-constrained project scheduling problem. An innovative chromosome representation scheme for SPEA II was proposed. A set of case studies were tested to compare the optimization performance of the proposed method with the best of the existing techniques. The results showed that the method yields better results than popular methods presented in the literature.

INTRODUCTION

Project scheduling is the basis of decision-making in project planning and management. Resource-constrained project scheduling problems (RCPSPs) were proposed to optimize scheduling under resource constraints. The objective functions of many RCPSPs are to minimize the project duration or to minimize resource cost within a predetermined project duration.

The RCPSPs can be categorized into single-mode and multi-mode approaches in terms of the number of modes in which resources are used or consumed. When activity durations are variable in different modes, the combination of different modes of activities generates varied costs. In this case, the RCPSP may be evolved into a discrete time-cost tradeoff problem (Brucker et al. 1999).

Metaheuristic algorithms, especially genetic algorithms (GAs), have been used widely to solve RCPSPs and its sub-problems. The objective functions of the GAs were usually set to minimize the project duration (Chen and Shahandashti 2009; Goncalves et al. 2008; Hartmann 1998). The chromosomes of GAs have been represented as permutation based, priority rule based and priority value based. Serial and parallel scheduling techniques have been adopted to convert chromosomes into the actual schedule (Kolisch 1996).

Resource fluctuation has been studied mainly within the scope of the resource leveling problem which describes the process of reducing the fluctuations in resource
usage over the project duration. Undesired resource fluctuations may cause inefficient and costly implementation of construction, for example, frequently rehiring and releasing workers, lowering production levels and interruption of learning curve effects (El-Rayes and Jun 2009). These implicit costs incurred by undesired resource fluctuations can account for a large portion of resource costs. Heuristic algorithms, such as particle swarm optimization (PSO) and GAs have been adopted to solve resource leveling problems (Chan et al. 1996; El-Rayes and Jun 2009; Senouci and Eldin 2004).

Resource leveling and resource-constrained project scheduling problems are inherently interrelated. A certain schedule having a higher resource cost may have a lower resource fluctuation. However, the two problems have usually been studied independently. Only a few integrated models have been developed to solve the two problems simultaneously (Liao et al. 2011). Resource leveling has usually been addressed in these models by adding a constraint (Senouci and Eldin 2004). These methods favored a predetermined resource profile that may be difficult to achieve (Leu and Yang 1999). In addition, PCPSPs were not fully explored in the integrated models. These studies mainly used a single objective function to minimize project duration, cost, or deviation between resource usage and the defined value (Chan et al. 1996). Only the single-mode RCPSP has been integrated with the resource leveling problem. Therefore, this research aimed to develop a model solving the resource leveling and multi-mode resource-constrained discrete time-cost tradeoff problems simultaneously by using a multi-objective optimization.

PROBLEM DESCRIPTION

The mathematical description of the problem is shown in Eqs.(1)-(4). The problem can be solved by using a multi-objective optimization technique whose objective functions are to minimize the resource cost (i.e., Eq. (1)) and the total project duration (i.e., Eq. (2)). The cost is split into two parts: resource usage cost ($RUC$) and resource fluctuation cost ($RFC$). The constraints include the resource feasibility constraint (Eq.(3)) and the precedence constraint (Eq.(4)).

\[
\begin{align*}
\text{Min} \quad C_{\text{total}} &= RUC + RFC \\
\text{Min} \quad T &= \max(LF_1, LF_2, ..., LF_n) \\
\text{s.t.} \quad RU_{m,t} &\leq R_{m,t} \\
FT_i &\leq ST_j
\end{align*}
\]

Where $R_{m,t}$ = the capacity of a resource type $k_m$ at a time period $t$; $RU_{m,t}$ = the calculated amount of resource $m$ used at a particular time $t$. $FT_i$ = finish time of activity $i$ whose successor is activity $j$. $ST_j$ = start time of activity $j$.

The resource usage cost ($RUC$) is defined as the total cost of the resources used by all the activities in the given activity modes. Resource fluctuation costs ($RFC$) denotes additional costs incurred when resources are newly added or dismissed. Such costs may include training, transportation and bidding costs. El-Rayes and Jun (2009) proposed two metrics: release and re-hire ($RRH$) and resource idle days ($RID$), to measure the level of the resource fluctuation. The $RRH$ was calculated from Eq. (5) (El-Rayes and Jun 2009).
\[ RRH = H - MRD = \frac{1}{2} \times \left[ r_t + \sum_{i=2}^{n} \left| r_i - r_{i-1} \right| + r_T \right] - \text{Max}(r_1, r_2, ..., r_n) \]  

(5)

Where \( H = \text{total increases in the daily resource demand; } T = \text{total project duration; } r_t = \text{resource demand on day } t; \) MRD=maximum resource demand during the entire project duration.

In this research, the metric of measuring the resource leveling cost RFC was calculated based on the metric RRH. RFC is calculated by Eq. (6). \( UHC_m \) denotes the unit cost of hiring and releasing a resource type \( m \) and \( RRH_m \) denotes the amount of a hired or increased resource \( m \) calculated by Eq. (5). The cost of MRD is also added to RFC to minimize the resource demand.

\[ RFC = UHC_m \cdot RRH_m + C_m \cdot MRD \]  

(6)

MODEL DEVELOPMENT

The model consists of three modules: the evolutionary multi-objective optimization (EMO), the project scheduling, and the resource leveling (see Figure 1). The fitness values of the EMO are calculated based on time and cost fed by the project scheduling and resource leveling modules.

The EMO module

The EMO module uses the strength Pareto evolutionary approach II (SPEA II) to find a Pareto optimal solution. SPEA II was developed based on the natural evolutionary principle (Zitzler et al. 2001). The previous study showed that SPEA II performed well in RCPSPs (Ballestín and Blanc 2011). SPEA II conducts crossover and mutation as shown in Figure 1. SPEA II uses an external archive containing non-dominated solutions previously found. Non-dominated individuals are copied to the external non-dominated set (see Steps 4-6 of the EMO module in Figure 1). SPEA II uses an enhanced archive truncation method that preserves the boundary solutions. The fitness assignment strategy (see Step 2 in Figure 1) and GA operations (see Step 7 in Figure 1) considers both closeness to the true Pareto front and even distribution of solutions. The non-dominated points are also preserved based on the fitness values.

The encoding approach is illustrated in Figure 1. The first section of the chromosome denotes the priority values which are unique integers in the range from 1 to \( n \) (i.e., the total number of activities). The second section denotes the mode of activities. When the two sections are specified, chromosome gene values would be fed into the resource leveling module to generate a genotype (i.e., schedule).

SPEA II calculated fitness using \( R(i) + D(i) \) (Zitzler et al. 2001) where \( R(i) \) is the function of “strength” which can be derived from cost and project duration and \( D(i) \) is density estimator calculated as the inverse of the distance to the k-th nearest neighbor. \( D(i) \) is calculated to ensure that the points are evenly distributed along the known Pareto front and to avoid cluster. The fitness values of the non-dominated individuals are less than 1.

The crossover method developed by Hartmann (1998) was adopted to implement the crossover operation in the first section of the chromosome. For instance, the left part of the gene values in child 1 comes from parent 1 and the right part in child 1 is from the parent 2 by the left-to-right scan. The scan makes the right part of child 1
take the gene values in parent 2 which are different from the gene values in the left part of child 1.

The mutation of the first section of genes is achieved by swapping two randomly selected genes in a chromosome. For the second section of the chromosome, the mutation operator modifies modes at each position with an equal probability. If a position is selected, a random value within the range of the total number of modes is selected.

**Figure 1. Flow chart of the model**

**The project scheduling module**

The activity and mode lists stored in the chromosome are fed into the project scheduling module to formulate the schedule. Activities are scheduled according to
the serial scheme scheduling approach (Kolisch 1996). Three activity sets are established, namely the scheduled set, eligible set, and decision set (see step 1 in Figure 1). At first, all the activities are placed in the decision set. The decision set contains all the activities to be scheduled. The scheduled activity set contains activities that have been scheduled (i.e., starting and finishing time is determined). The activities in the decision set whose precedent activities are scheduled and included in the scheduled set are then moved to the eligible set. Several activities may be moved to the eligible set simultaneously. Only the activity in the eligible set having the largest priority value is selected to be moved to the scheduled set. This activity is added to the project schedule chart and the starting time is adjusted to meet the resource constraint. This process is repeated until all the activities in the decision set are moved to the schedule set. This method is called the backward pass.

The last task in this module is to calculate the total float (TF) for each activity. The following method is used: the backward pass is used to determine the late activity finish times. Then, by deducting the latest finish time by the earliest finish time, the TF of each activity can be calculated.

**The resource leveling module**

The resource leveling module calculates the project duration and cost as measured by the fitness calculation of the EMO module. The first task is to adjust the noncritical activities (i.e., activities whose TF is larger than zero) to reduce the RFC. Where to locate these noncritical activities in the schedule is critical to reducing the RFC. Reducing RFC can be achieved by reducing the sum $RRH + MDR$ (see Eq. (6)). First of all, the noncritical activities are removed from the schedule and form a noncritical activity list (see Step 1 of the resource leveling module in Figure 1). In the subsequent steps, in terms of the priority values of the noncritical activities, each activity will be added into a location where the maximum reduction of $RRH + MDR$ is achieved (see step 2 in Figure 1). After a noncritical activity is added to the schedule, the start and finish time of other noncritical activities are updated. Following this process, all the noncritical activities are added to the schedule (see step 4 in Figure 1).

In step 3, the noncritical activities are placed by aligning a newly added noncritical activity to existing activities in the schedule. An example is shown in Figure 2. The grey highlighted bar denotes the resource usage of the existing activities in the schedule. Two new activities with oblique lines are added to the schedule. To achieve the maximum reduction of RFC, the two activities should be aligned with the edge of existing activities. Thus, according to Eq. (5), if the first activity may be located at $P_1$, the reduction of $RRH$ is $R_1$ and if the second activity is added at the position $P_2$, the reduction is $R_2 + R_3$.

After an activity has been added to the existing schedule, it will be merged into the existing schedule. The next noncritical activity is added based on the newly generated schedule where the maximum reduction of $RFC$ can be achieved. This approach uses a greedy algorithm which makes the locally optimal choice at each stage with the intent of finding the global optimum. The sequence of adding noncritical activities depends on the priority values of these noncritical activities. An
activity with a highest priority value will be chosen first to add the existing schedule to achieve the maximum reduction of $RRH + MDR$.

![Figure 2. An example of the RRH reduction method](image)

**COMPUTATIONAL EXPERIMENT**

The developed model was implemented in MATLAB Version 7.9 using a laptop with a 2.0 GHz CPU and 2G RAM. A similar dataset of test cases originally created for the multi-mode RCPSP were selected from the well-known PSPLIB (Kolisch and Sprecher 1997). Ten cases from the dataset “j10.mm” (available in PSPLIB: http://129.187.106.231/psplib/) were selected (selected case numbers: 1, 50, 100, 150,…450). A baseline model was proposed for comparison with the developed model.

**Baseline model**

The baseline model has two phases each of which uses the GA algorithm. (a) In Phase I, a number of the minimum project durations were generated. The objective was to minimize the project duration only. This was regarded as a standard multi-mode RCPSP that could be solved by a number of algorithms (Hartmann 2001; Wuliang and Chengen 2009). An archive was included to store the 30 best individuals in the GA simulation. (b) In Phase II, the method developed by El-Rayes and Jun (2009) was used to calculate the minimum cost of $RRH$ (corresponding to $RFC$) for each schedule (or individual) preserved in Phase I. Then the minimal total cost (i.e., $RUC + RFC$) is calculated.

Referring to Figure 3, assume the black solid point $P_0$ denotes $(T, C)$ ($T =$ time, $C =$ cost) calculated by the baseline model and was located in the Pareto front line. The point $(T,C)$ can be compared with the white points $P_i$ calculated by the tested model (i.e., the model developed in this research). Four cases may occur after the comparison (see Figure 3): “dominated”, “non-dominated”, “dominate” and “equal”. $P_1$ dominates the point $(T, C)$ and $P_2$ is dominated by $(T, C)$. $P_0$ may dominate many points $P_i$, thus, the average point is calculated. The percentage error $E$ in regard to $P_0$ is calculated as $\left(\frac{\sum P_i/n-P_0}{P_0}\right)/P_0$ where the point $P_i$ is calculated as the tested model and dominated by $P_0$; $n$ is the number of points dominated by $P_0$. The grey point in Figure 3 denotes the average position of the two points dominated by $P_0$.

**Experimental results**

Two groups of resource unit cost were tested. The resource unit cost was set to 10 for both groups and the unit hiring/releasing cost (UHC) was set to 10 for one group...
and 20 for another. Preliminary tests were conducted to determine the simulation parameters. The test results showed that the mutation rate was set to be 0.05, which was the same as that recommended by Hartmann (1998), the population and archive sizes were set to 30. The percentage errors $E$ were calculated for each case. It was found that only cases #1 and #2 had errors (see Table 1). Further analysis showed that the reason why most of the errors were zero was that for all the ten cases, more than 80% of the points generated by the tested model dominated the points generated by the baseline model, that is to say, most of the calculated points by the test model were located in the lower left corner of the coordinator system in Figure 3. The reason was that the baseline model did not find the true global optimum individuals or only found the individuals that were inferior to those calculated by the test model. Though the experiment could not guarantee the points generated by the tested model were located on the Pareto front line, the experiment still showed that the tested model had the potential to generate relatively satisfactory results and also had better performance than the baseline model.

<table>
<thead>
<tr>
<th>Cases with errors</th>
<th>Group 1(UHC = 10)</th>
<th>Group 2(UHC = 20)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time</td>
<td>Cost</td>
</tr>
<tr>
<td>#1</td>
<td>0.0%</td>
<td>1.1%</td>
</tr>
<tr>
<td>#2</td>
<td>0.0%</td>
<td>0.0%</td>
</tr>
</tbody>
</table>

The time and cost of case #2 are shown in Figure 4. When UHC = 10, the resource usage cost had a larger effect on the total cost and the increase of duration led to a larger decrease in the total cost. The results indicated that the RFC accounted for about 20% of the total cost (UHC = 10) and 25% (UHC = 20). When the duration increased by about 12%, the total cost decreased by 12% (UHC = 10) and 15% (UHC = 20).

![Figure 4. Time and cost for case #2](image)

**CONCLUSIONS**

This research proposed a time-cost tradeoff model based on SPEAII to optimize the project duration and resource costs by integrated consideration of the resource-constrained project scheduling and resource leveling problems. Ten test cases were selected to evaluate the accuracy of the model by comparison with the baseline model that used the popular methods presented in the literature. The test showed that in two out of the ten cases the errors of the model were less than 5%, and in the other test
cases the model generated better solutions than the baseline model. Future work may focus on improvement of the model (e.g., using dynamic unit costs of resources to evaluate the fluctuation cost) and developing models to solve the resource-constraint and resource leveling problems for multiple projects using shared resources.

REFERENCES


