An Efficient Simulation Algorithm for Resource-Constrained Project Scheduling Problem

L. Peng and P. Wuliang*

Shenyang Ligong University, Shenyang, 110159, China

Abstract: Since Resource-Constrained Project Scheduling Problem (RCPSP) is a well-known NP hard problem, it is difficult to solve large-scale practical cases by using traditional exact algorithms. Genetic algorithm (GA) is a kind of intelligent algorithm for approximate optimization, and it can find out the global optimization or the suboptimal solution within a reasonable time. This article presented a new simulation algorithm by using GA for solving Resource-Constrained Project Scheduling Problem. In the algorithm, the activity adjacency matrix and priority-based preemptive resource conflict resolution are used to prevent chromosome from generating infeasible schedules. Finally, the method is tested with an actual machine and electricity project case, and the results show that the presented method is efficient and practical for practical project cases.

Keywords: Genetic algorithm, intelligent algorithm, project management, project scheduling, simulation algorithm.

1. INTRODUCTION

A project is a group of discernible tasks or activities that are conducted in a coordinated effort to accomplish one or more objectives. Projects require varying levels of cost, time and other resources. Project scheduling is the core problem of the project management. It is used by the project manager to commit people to the project and show the organization how the work will be performed. No matter the size or scope of a project, the schedule is the primary concern of project management. The schedule describes when activities should be done, what has already been completed, and the sequence in which things need to be finished. The requirement and complexity of project scheduling increase along with the increase of the number of project activities. Since RCPSP is a well-known NP hard problem, finding feasible schedules which efficiently use scarce resources is a meaningful and challenging task in project management. In this context, the well-known Resource Constrained Project Scheduling Problem (RCPSP) has been studied during the last decades, see e.g. [1-6]. Nevertheless, all the existing studies are still centralized on the conceptual model as well as algorithms by using generated project instances, and the practical research is scarce up to now.

As RCPSP is an NP-hard problem, traditional exact solutions can’t solve it with a reasonable time. For this, this article presents an improved GA as the simulation algorithm for solving Resource-Constrained Project Scheduling Problem. In the algorithm, the activity adjacency matrix and priority-based preemptive resource conflict resolution are used to prevent chromosome from generating infeasible schedules. Finally, the method is tested and verified by an actual machine and electricity project case.

2. SIMULATION MODEL

We assume that a project is represented by an activity-on-the-node network $G=(V,E)$ in which $V$ denotes the set of vertices (nodes) representing the activities and $E$ is the set of edges (arcs) representing the finish-start precedence relationships with zero time-lag. The activities are numbered from 1 to $n$, where the dummy activities 1 and the end activity $n$ mark the beginning and the end of the project. The activities are to be performed without preemption. The fixed integer duration of an activity is denoted by $d_i (1 \leq i \leq n)$, its integer starting time by $s_i (1 \leq i \leq n)$ and its integer finishing time by $f_i (1 \leq i \leq n)$. There are $K$ renewable resource types with $r_k (1 \leq i \leq n, 1 \leq k \leq K)$ where the constant resource requirement of activity $i$ for resource type $k$ and $a_k$, the constant availability of resource type $k$. Conceptually, the simulation model of RCPSP can be formulated as follows [7,8]:

$$
\begin{align*}
\text{Min} & \quad f_n \\
\text{s.t.} & \quad f_i = 0 \\
& \quad f_i - t_j \geq f_i \quad \forall (i,j) \in E \\
& \quad \sum_{j \in A_k} r_{kj} \leq a_k \quad t = 1,2,3,\ldots,f_i; k = 1,2,\ldots,K
\end{align*}
$$

where $E$ denotes the set of pairs of activities indicating precedence constraints and $A(t)$ denotes the set of activities in progress in time interval $[t-1,t]$: $A(t) = \{i | f_i - d_i < t \leq f_i \}$. 

1874-155X/14 2014 Bentham Open
(2) Assigns a completion time of 0 to the dummy start activity 1. The precedence constraints given by (3) indicate that activity $j$ can only be started if all predecessor activities $i$ are completed. The resource constraints given in (4) indicate that for each time period $[t-1, t]$ and for each resource type $k$, the renewable resource amounts required by the activities in progress cannot exceed the resource availability. The objective function is given as (1). The project duration is minimized by minimizing the finishing time of the unique dummy ending activity $n$.

### 3. PROPOSED SIMULATION ALGORITHM

In the computer science field of artificial intelligence, the genetic algorithm (GA) is a search heuristic that simulates the process of natural evolution. This heuristic (also sometimes called a meta-heuristic) is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. Genetic Algorithm will express the excellent ability to quickly solve very difficult problem and is very easy to be merged with other technologies. Therefore, this article design the simulation algorithm based on genetic algorithm to solve RCPSP.

#### 3.1. The Structure of the Algorithm

We can realize the optimization of resource-constrained scheduling plan of the project, the flow chart of the whole project is shown in Fig. (1), and the group scale is M and the evolutionary generation is $T$.

#### 3.2. Encoding Schema

In order to apply the PSO, it is necessary to find a suitable encoding schema mapping between the RCPSP and the PSO particle. There are two main encoding schemes for RCPSP in the current research: one is priority-based encoding, the other is a permutation encoding [1]. In the priority-based encoding, all the priority values of activities are set in the priority list, and the position of the list is denoted by an activity ID. When the resource conflicts occur, the activity having the highest priority value is scheduled firstly. In permutation encoding, all the activities are permuted in a precedence-feasible activity list, where the activity closer to the head of the list will be scheduled earlier. The advantage of priority-based encoding is that the generation and evolution of the population are independent of the schedule generation schemes, while its drawback is that the search space may be too large due to the fact that the priority value can be set at all the real numbers, as shown in Fig. (2b). As the values in the precedence-feasible activity list must be integers which are activity IDs and different from

---

**Fig. (1).** The basic process of the simulation algorithm based on GA.
each other, the search space in the permutation encoding is remarkably reduced. The drawback of the permutation encoding is that during the generation and evolution of the population the precedence relationships of activities must be considered, so the search algorithm and the schedule generation scheme are tightly-coupled.

(a) New encoding

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

Values: Activity priority

(b) Priority-based encoding

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>5.5</td>
<td>2.2</td>
<td>3.0</td>
<td>1.2</td>
</tr>
</tbody>
</table>

Values: Activity priority

Equivalent to

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5</td>
<td>4.9</td>
<td>2.0</td>
<td>2.9</td>
<td>1.1</td>
</tr>
</tbody>
</table>

Values: Activity priority

Fig. (2). New encoding and priority-based encoding.

In this paper, we presented a new permutation of priority-based encoding scheme, which combines the merits of both the permutation encoding scheme and the priority-based encoding scheme. As shown in Fig. (2a), a particle is represented by a list, where the positions denote activity IDs and the values denote the priorities associated with the activities. The priority value is an integer exclusively within [1, n]. The larger the integer, the higher the priority is. When the resource conflicts occur, the activity that has a higher priority should be scheduled firstly. Therefore, all the particles represented by the permutation of priority-based encoding scheme can be translated into an active project plan, and the new encoding scheme keeps all the merits of priority-based encoding due to the precedence relationships need not be considered during the generation and evolution of the population. At the same time, just like the permutation encoding, all the values in the permutation of priority-based encoding are integers different from each other, so the search space is also reduced too.

3.3. Fitness Function

The fitness function is also known as evaluation function, it is the standard which is based on Objective Function that is used to distinguish the individual in the group, it is the driving force of the change of algorithm and the only basis for the nature selection.

As we need the shortest activity duration, this is a minimum problem, so we should convert the primal objective value into fit value to make sure that the excellent individuals have the maximum data to design fitness function.

\[
f(i) = \frac{D_{\text{Max}} - D_i + \gamma}{D_{\text{Max}} - D_{\text{Min}} + \gamma}
\]  

3.4. Selection Operator

This article is selected by roulette selection operation and based on the fitness value which is calculated in upside steps. The roulette selection operation is also called random selection, the probability of each personality to be in the next generation is equal to the proportion of its fitness value to the sum fitness of all the chromosomes. The higher the fitness value is, the probability to be selected is larger and the probability to be in the next generation is larger, this is a common selection method in genetic algorithms. In this method, the random number \( r \) will be generated at first and selected in the method as follows:

\[
\sum_{j=1}^{k} f(x^{j-1}) \leq r \leq \sum_{j=1}^{k} f(x^{j}), j = 1, 2, \ldots, k
\]

3.5. Crossover Operator

Crossover operation imitates the process that the two homologous chromosomes mate and resultantly a recombinant new chromosome generates new individuals and species. This article uses the two-point crossover pattern. However, in the traditional two-point crossover as in Fig. (3), if we choose the 3 loci and the 4 loci as cross position, we can find that there are two priority rules of 7 activities list of the newly generated chromosome 1, as a result, it cannot fit for the standard coding rules in this article.

Fig. (3). Traditional two-points crossover.

Therefore, this article presents a two-point crossover which is based on position mapping. The first step of the crossing operation is same as the traditional two-point crossover, exchange the 3 and 4 alleles from parent 1 and parent 2, the second step is establishing the mapping relationship to the alleles as shown in Fig. (4). The alleles of parent 1 and the alleles of parent 2 have connected mapping, and when its uncrossing part met with 7, it will be replaced as 4, using the same method to map and cross. Finally, the
children chromosomes with non-repeated priority can be generated, both of them have inherited the gene character of parent 1 and parent 2.

![Two-point crossover mutation operator](image)

**Fig. (4).** Two-point crossover mutation operator.

Mutation operation simulates the process of biological heredity and natural evolution, some replication mistakes might appear because of some accidental factors during the cell division, as a result, some genes may be mutated, and new chromosome appear and different character appear. This article mainly adopts the mutation method based on centric positions as follows: 1) Compute the number of mutagenic genes $U$ according to: $U = \text{the total number of genes} \times P_m$, where $P_m$ is mutation probability; 2) Generate $U$ (from 1 to the total number of genes) random number as the mutations of genes; 3) Locate related chromosomes; 4) Generate new chromosome based centric positions, where the positions of genes in a chromosome are switched randomly.

### 3.6. Parameters Setting

The selection of control parameters in genetic algorithm is very important, different selection of the control parameter has different influence on performance of genetic algorithms. Especially, it has an influence on the convergence of the algorithm. In the optimization procedure, the crossover probability consistently controls the crossover operator which is on the leading status in the mid-term of genetic algorithms. Undesirable results could be caused by unsuitable crossover probability. The frequency of use of the crossover operation is controlled by crossover probability. Larger crossover probability can make each generation fully crossed, but the possibility of the excellent schema to be destroyed will increase, as a result, a big generation gap will appear and the research will be randomized. The lower the crossover probability, the smaller the generation gap, so it could keep a continuous solution space, and increase the possibility of the global optimum solution, but the evolution speed will be slower. If the crossover probability is so low that more individuals will replicate their generation directly, and the genetic search will be logjam. Mutation operation is the improvement of the genetic algorithm, it makes some repairing and supplement of the gene which has been lost in the process of systematic, it also can prevent the genetic algorithm from converging to a locally optimal solution. The frequency of use was controlled by the mutation probability. When the mutation probability is larger, even though it can generate more individuals and increase the varieties of the population, it can destroy many good individuals, or make the performance of genetic algorithms equal to the performance of random searching algorithm. If the mutation probability is small, the ability of the mutation operation in generating new individual and inhibition to early-maturing will be poor.

The main method to generate new individuals has the global research ability, and the mutation operation is only the subsidiary method to generate new individuals. But it deals with local search ability. Crossover operator should be combined with mutation operator to finish the local search to search space together, then to make a genetic algorithm to finish the searching process of the optimization problem with a good searching performance.

### 4. CASE STUDY

As listed in Table 1, there are 21 activities in a company’s development project, the first one and the last one are represented as the starting-activity and the fishing-activity, the available key resource numbers are 6, 4, 2. Fig. (4) has listed out the relationships among activities, the duration of activities and resource consumption.

<table>
<thead>
<tr>
<th>ID</th>
<th>Duration</th>
<th>Res.1</th>
<th>Res.2</th>
<th>Res.3</th>
<th>Preceding Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>15</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>5</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>16</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>7, 14, 15</td>
</tr>
<tr>
<td>17</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>9, 10, 11</td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>8, 16</td>
</tr>
<tr>
<td>19</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>12, 13, 17, 18</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td>20</td>
</tr>
</tbody>
</table>

The results are shown in Fig. (5), by using the traditional project scheduling method in manual ways, this make-span of the project is 720 days, while GA algorithm made it
Algorithm for Resource-Constrained Project Scheduling Problem

The Open Mechanical Engineering Journal, 2014, Volume 8 13

finished in 600 days. 100 days have been shortened, then it can be pulled into the market. Therefore, GA has an obvious effect on utilizing scarce resources in project scheduling, and thus it is suitable for generalizing the optimal project schedule for enterprise project management.

Fig. (5). The running process of the algorithm.

5. CONCLUSION

Under the environment of market economy, project scheduling problem for enterprise management is of great meaning to make full use of resources, accelerate product development and put the new products on sales as soon as possible to occupy the market share. This article presents a simulation algorithm based GA for solving Resource-Constrained Project Scheduling Problem. In the algorithm, the activity adjacency matrix and priority-based preemptive resource conflict resolution are used to prevent chromosome from generating infeasible schedules. Finally, the simulation algorithm is tested by an actual machine and electricity project case, and it is verified that the improved genetic algorithm is suitable in the real-world project scheduling.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This work was funded by the National Natural Science Foundation of China under Grant No.71071100.

REFERENCES


Received: November 11, 2013 Revised: February 11, 2014 Accepted: March 3, 2014

© Peng and Wuliang; Licensee Bentham Open.

This is an open access article licensed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0/) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.