Send Orders for Reprints to reprints@benthamscience.ae

The Open Mechanical Engineering Journal, 2014, *8*, 9-13

1Open Access

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, which can ascertain global optimization or 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 was 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 with coordinated efforts to accomplish one or more objectives. Projects require varying levels of cost, time and other resources. Project scheduling is the core problem of 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. Irrespective of 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 tasks need to be finished. The requirement and complexity of project scheduling increase along with the increase in 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, as shown in [1-6]. Nevertheless, all the existing studies are still centralized on the conceptual model as well as algorithms by using instances of generated project , and practical research is limited up till now.

As RCPSP is an NP-hard problem, traditional exact solutions cannot solve it in a reasonable time. Therefore, 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.

*Address correspondence to this author at the Shenyang Ligong University, Shenyang, 110159, China; Tel: 0086-024-24682231;
Fax: 0086-024-24682231; E-mail: pengwuliang@126.com

2. SIMULATION MODEL

It was assumed 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 respectively. The activities are performed without preemption. The fixed integer duration of an activity is denoted by $d_i(1 \le i \le n)$. The stating time of its integer is represented by $s_i(1 \le i \le n)$ and the finishing time is represented by $f_i(1 \le i \le n)$. There are K renewable resource types with $r_{ik}(1 \le i \le n, 1 \le k \le K)$ where the constant resource requirement of activity is i for resource type k and a_k is the constant availability of resource type k. Conceptually, the simulation model of RCPSP can be formulated as follows [7,8]:

$$Min \qquad f_n \tag{1}$$

$$s.t. \qquad f_1 = 0 \tag{2}$$

$$f_j - t_j \ge f_i \qquad \forall (i,j) \in E \tag{3}$$

$$\sum_{j \in A_t} r_{jk} \le a_k \quad t = 1,2,3......f_n; k = 1,2,......K \tag{4}$$

Where E denotes the set of pairs of activities indicating precedence constraints and $A(t)$ denotes the set of activities

1874-155X /14 2014 Bentham Open

in progress in time interval $[t-1,t]$: $A(t)=\{i \mid f_i - d_i < t \le f_i\}$. (2) A completion time of 0 to the dummy is assigned to begin 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 expresses an excellent ability to quickly solve very difficult problem and is very easy to be merged with other technologies. Therefore, this article designed the simulation algorithm based on genetic algorithm to solve RCPSP.

3.1. The Structure of the Algorithm

The optimization of resource-constrained scheduling plan of the project, with the flow chart of the whole project is shown in Fig. (**1**), in which the group scale is represented by M and the evolutionary generation is represented by *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 and 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 are identified, the activity having the highest priority value is scheduled firstly. In permutation encoding, all the activities are permutated 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 domain of search 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 preceding-feasible activity list must be based on integers which are included in activity IDs and different from each other, the search space

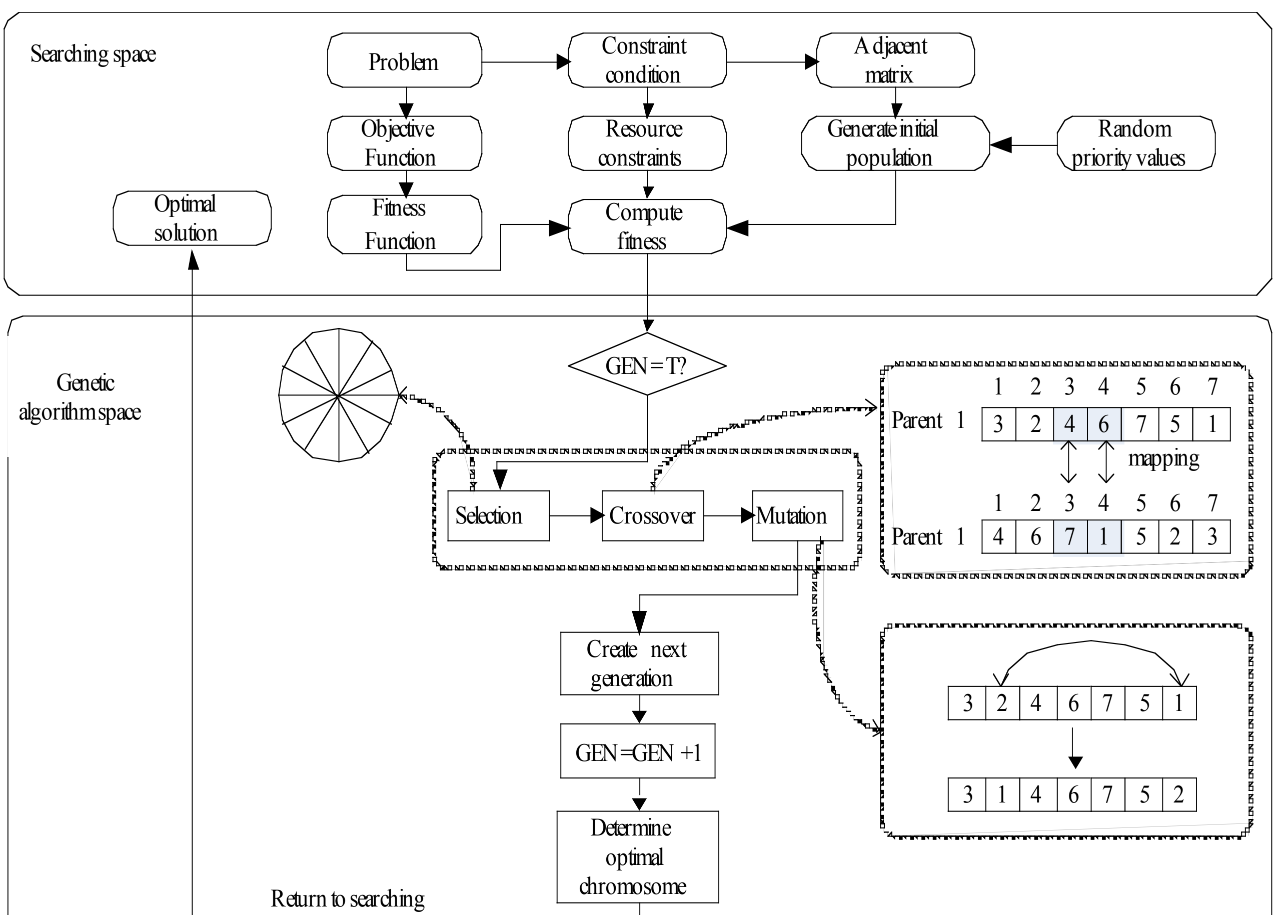

Fig. (1). The basic process of the simulation algorithm based on GA.

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, therefore, the search algorithm and the schedule generation scheme are tightly-coupled.

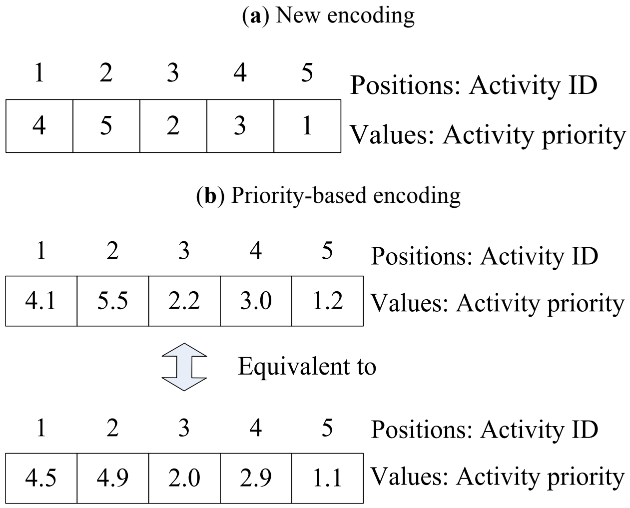

Fig. (2). New encoding and priority-based encoding.

In this paper, a new permutation of priority-based encoding scheme was presented, 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 is the priority. When the resource conflicts occur, the activity that has a higher priority should be scheduled first. Therefore, all the particles represented by the permutation of priority-based encoding scheme were translated into an active project plan, and the new encoding scheme demonstrated all the merits of priority-based encoding due to the reason that precedence relationships did not need to 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 were integers different from each other, as a result, the search space was also reduced .

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 bringing a change in the algorithm and is the only basis for natural selection.

Since the requirement was to obtain the shortest activity duration, this was a minimum problem, therefore, the primal objective value was converted into fit value to make sure that the excellent individuals have the maximum data to design fitness function.

$$f(i) = \frac{D_{Max} - D_i + \gamma}{D_{Max} - D_{Min} + \gamma} \qquad (1)$$

i is the ith chromosome, $f(i)$is the fitness function, D_{Max} is the maximum target value in the group, D_{Min} is the minimum target value in the group, and D_i is the target value which is to be exchanged. γ is the positive real number in (0,1). There are two advantages of using γ: 1) preventing (1) exact division and 2) adjusting the selection behavior from adapt value proportion of selection to pure random value proportion of selection. However, if the gap in the chromosome is relatively large, then the adapt value proportion of selection should be used. While, if the difference is relatively small, random value proportion of selection should be preferred in the competitive chromosome.

3.4. Selection Operator

This article is selected by roulette selection operation and is based on the fitness value which is calculated in reverse steps. The roulette selection operation is also called random selection. In this method, 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 , the greater the probabilityto be selected and the probability in the next generation is also larger. This is a common selection method in genetic algorithms. In this method, random number r is generated at first and selected in the method as follows:

$$\sum_{j=1}^{k} \frac{f(x^{j-1})}{\sum_{k=1}^{popsize} f(x^{j-1})} \le r \le \sum_{j=1}^{k} \frac{f(x^{j})}{\sum_{k=1}^{popsize} f(x^{j})}, j = 1,2,\ldots k \qquad (2)$$

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 used the two-point crossover pattern. However, in the traditional two-point crossover as shown in Fig. (**3**), if the 3 loci and the 4 loci were selected as cross positions, it can be observed that there were two priority rules of 7 activities list of the newly generated chromosome 1. As a result, it could not fit for the standard coding rules employed in this article.

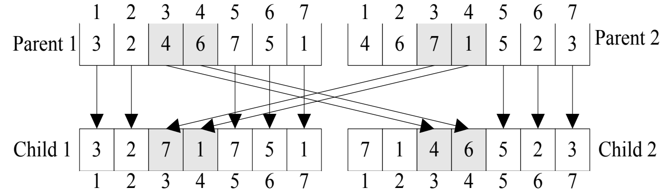

Fig. (3). Traditional two-points crossover.

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, exchanging the 3 and 4 alleles from parent 1 and parent 2. The second step involves 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 reaches 7, it is replaced as 4, using the same method to map and cross. Finally, the children chromosomes with non-repeated priority are generated, and both of them inherited the gene character of parent 1 and parent 2.

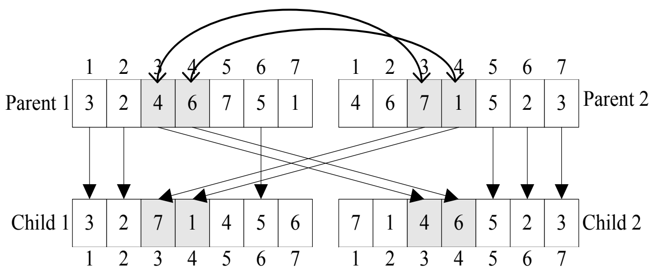

Fig. (4). Two-point crossover mutation operator.

Mutation operation simulates the process of biological heredity and natural evolution. However, 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 may appear exhibiting different character. This article mainly adopted the mutation method based on centric positions as follows. 1) Computing the number of mutagenic genes U according to: U=the total number of genes× P_m , where P_m is the mutation probability. 2) Generating U (from 1 to the total number of genes) random number as the mutations of genes. 3) Locating related chromosomes. 4) Generating 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 as different selection of the control parameter has different influence on the 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 of the leading status in the mid-term of genetic algorithms. Undesirable results could be caused by unsuitable crossover probability. The frequency of usage 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, which could keep a constant solution space, and increase the possibility of the global optimum solution, but the evolution speed will be slower. If the crossover probability is too low that more individuals replicate their generation directly, the genetic search will be logjam. Mutation operation is used for the improvement of the genetic algorithm. It performs repairing and can serve as supplement of the gene which has been lost in the systematic process. Moreover, it also can prevent the genetic algorithm from converging into a locally optimal solution. The frequency of use is 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 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, to make a genetic algorithm in order 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, and the available key resource numbers are 6, 4, 2. Fig. (**4**) lists out the relationships among activities, the duration of activities and resource consumption.

Table 1. Activities of the project.

ID	Duration	Res.1	Res.2	Res. 3	Preceding Activities
1	0	0	0	0	
2	10	3	0	0	1
3	10	3	0	0	1
4	12	3	0	0	1
5	8	2	2	1	2
6	5	2	3	1	2
7	4	2	3	1	2
8	2	2	3	1	2
9	4	3	3	1	3
10	3	3	3	1	3
11	2	1	2	2	3
12	15	3	0	0	4
13	5	3	0	0	4
14	3	2	2	1	5
15	3	2	2	1	6
16	5	1	3	1	7,14,15
17	4	1	3	1	9,10,11
18	2	1	3	1	8,16
19	6	3	4	2	12,13,17,18
20	2	3	2	1	19
21	0				20

The results are shown in Fig. (**5**), by using the traditional project scheduling method manually. The make-span of the project was 720 days, while GA algorithm completed it in 600 days, shortening the procedure by 100 days. 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.

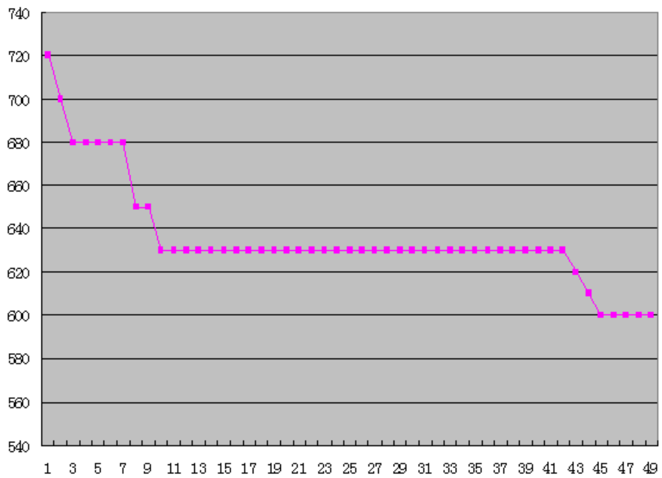

Fig. (5). The running process of the algorithm.

CONCLUSION

Under the environment of market economy, project scheduling problem for enterprise management is of great significance as it makes full use of resources, accelerates product development and puts the new products on sales as soon as possible to occupy the market share. This article presents a simulation algorithm based on GA for solving Resource-Constrained Project Scheduling Problem. In the algorithm, the activity adjacency matrix and priority-based preemptive resource conflict resolution were used to prevent the chromosome from generating infeasible schedules. Finally, the simulation algorithm was tested by an actual machine and electricity project case, and it was 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

[1] V. Valls, S. Quintanilla, and F. Ballesti'n, "Resource-constrained project scheduling: a critical activity reordering heuristic", *European Journal of Operational Research*, vol. 149, pp. 282-301, 2003.

[2] D. Debels, B. De Reyck, R. Leus, and M. Vanhoucke, "A hybrid scatter search/electromagnetism meta-heuristic for project scheduling", *European Journal of Operational Research*, vol. 169, pp. 638-653, 2006.

[3] J.J.M. Mendes, J.F. Goncalves, and M.G.C. Resende, "A random key based genetic algorithm for the resource constrained project scheduling problem", *Computers & Operations Research*, vol. 36, pp. 92-109, 2009.

[4] L. Ozdamar, and G. Ulusoy, "An iterative local constraints based analysis for solving the resource constrained project scheduling problem", *Journal of Operations Management*, vol. 14, pp. 193-208, 1996.

[5] P. Brucker, A. Drexl, R. Mohring, K. Neumann, and E. Pesch "Resource-constrained project scheduling: Notation, classification, models, and methods", *European Journal of Operational Research*, vol. 2, pp. 3-41, 1999.

[6] R. Kolisch, and S. Hartmann. "Experimental investigation of heuristics for resource-constrained project scheduling: An update", *European Journal of Operational Research*, vol. 174, pp. 23-27, 2006.

[7] P. Brucker, A. Drexl, R. Mohring, and E. Pesch, "Resource-constrained project scheduling: notation, classification, models, and methods", *European Journal of Operational Research*, vol. 112, pp. 3-41, 1999.

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/4.0/) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.