A new hybrid genetic algorithm and its application in the RCPSP

Li Zheng[a], Qin Jinlei[b]
[a]Department of Computer, North China Electric Power University, Baoding, China 071003;
[b]Information and Network Management Center, North China Electric Power University, Baoding, China 071003

ABSTRACT

A new hybrid genetic algorithm is generated in this paper, which is based on the simple genetic algorithm. In this algorithm, some genetic operators such as crossover operator are improved. In the crossover operator, the crossover method based on threshold and the two-points-crossover method are combined into a new hybrid crossover method. An example which is Resource-Constrained Project Scheduling Problem (RCPSP) is given, whose activity network, the execution time and the number of resource required for each activity, selection and crossover operator are also referred. In addition, there are examples to prove the superior of the new algorithm, which is benefit to speed up the evolution and get the optimal solution.

Keywords: hybrid genetic algorithm, RCPSP, hybrid crossover method, scheduling.

1. INTRODUCTION

Resource-Constrained Project Scheduling Problem (RCPSP) is a kind of broadly applied combinatorial optimization problem. It requires scheduling the tasks of a project reasonably to achieve the optimal object on the basis of meeting the resource constraint and the precedence relations. In theory, the problem belongs to NP-hard problems, and many scholars at home and abroad have proposed varies of optimal method, which can be summed up and divided into the followings: (1) exact algorithm which can be represented by branch and bound; (2) heuristic algorithm. Heuristic algorithm includes heuristic algorithm based on priority rules, sample algorithm and intelligent optimization method emerged recent years. Exact algorithm could find the optimal solution of the problem, but the scale of the problem can not be too big. Heuristic algorithm is just opposite, which can solve large scale problems in shorter time with stronger capability[1].

Resource-constrained and shortest duration problem means that under the resource constraints, not only the amount of resources required must meet the requirement of the amount of resource constraints in all time slices, but also the schedule duration of the project must be the shortest[2]. For example, the given project consists of a set of activities M={1,2,…,n}, where each of them must be processed without interruption. The activities 1 and n represent the beginning and end of the project. There are precedence relations between the activities, that is to say, the start time of a certain job depends on the completion of another job.

Therefore, the jobs can be sorted into a sequence j1j2…jn, where ji represent the number of the activity executed in the order of i, $j_i \in \{1,2,...,n\}$ [3].

Note the precedent job set of ji as P(ji)，and suppose the unit time is day. D(ji) and S(ji) represent the execution time and the start time of ji respectively. Al represents the executed job set in the day l. In the whole project, m is the kind number of the resource. Suppose that the number of resource k required while executing job ji is R(jik), and the total available number of resource k everyday is bk, then the final object is to find a sequence j1j2…jn, where the total duration of the project will be the shortest. Here $T(j_1 j_2 ... j_n) = \max_{i=1,2,...,n} \{S(j_i) + D(j_i)\}$, namely, the duration is the sum of the start time and the execution time of the final job.

Thus the mathematical model of such problems can be obtained as follows:

$$\min T(j_1 j_2 ... j_n) = \max_{i=1,2,...,n} \{S(j_i) + D(j_i)\} \quad (1)$$

s.t.

Fourth International Conference on Machine Vision (ICMV 2011): Machine Vision, Image Processing, and Pattern Analysis, edited by Zhu Zeng, Yuting Li, Proc. of SPIE Vol. 8349, 83490Z · © 2012 SPIE
CCC code: 0277-786X/12/$18 · doi: 10.1117/12.920126

$$S(j_k) - S(j_i) \geq D(j_l), \forall j_l \in P(j_k) \ , \quad k, l \in \{1,2,...,n\} \ , k \neq l, \tag{2}$$

$$\sum_{j_p \in A_l} R(j_{pk}) \leq b_k \ , \quad k = 1,2,...,m, \ \ l = 1,2,...,T \tag{3}$$

where the objective function(1) denotes the completion time of the final activity, which is represented by the sum of start time and execution time; Constraint set (2) stands for that only when the pre-ordered set of j_k completed can j_k start; (3) illustrates that the total consumption of resource k should not exceed the total amount of it.

2. AN EXAMPLE OF RCPSP

A typical example of RCPSP is shown in Fig.1, which includes 8 activities and only one kind of resource whose number is 7. A circle denotes an activity, or in other words, a job, while an arrowhead denotes the precedence relations between activities. The execution time and required resource of beginning and end activity1 and 8 are 0. In Table 1 the execution time and the number of required resource of every activity are displayed. A feasible schedule is shown in Fig.2.

A feasible schedule of the example L(1,2,4,3,6,5,7,8) is shown in Fig.2.

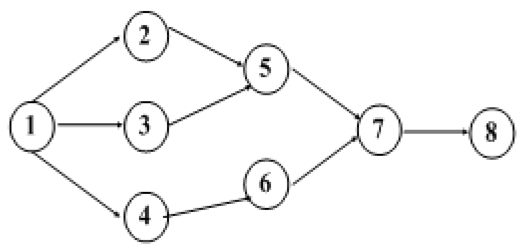

Figure 1. A simple activity-network description.

Table 1. The cost of each task.

activity (j_i)	**duration ($D(j_i)$)**	**Renewable resource ($R(j_{ik})$)**
1	0	0
2	2	3
3	3	5
4	4	4
5	2	2
6	4	2
7	3	7
8	0	0

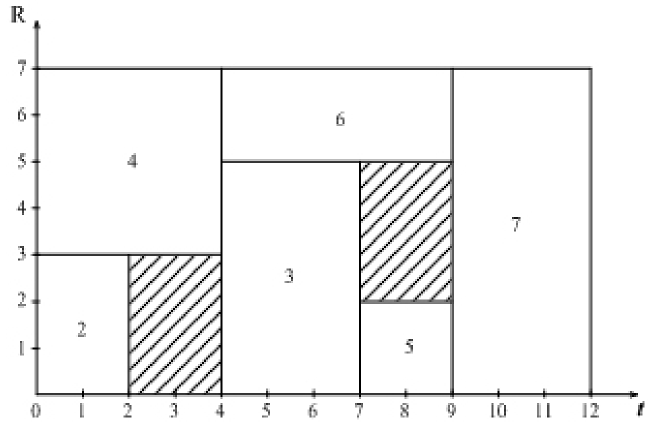

Figure 2. A feasible schedule of the Fig.1.

This section introduces the meaning of RCPSP and its mathematical model, in addition, an example is given. The rest of the paper is organized as follows: Section 2 is the example of RCPSP, and Section 3 is devoted to the description of the new hybrid genetic algorithm. Finally, the result and conclusion are included in section 4 and 5.

3. A NEW HYBRID GENETIC ALGORITHM MODIFIED

3.1 Encoding and production of the initial population

Parameters of the problem space can not be disposed directly, so they must be transformed into individuals which are composed of genes according to certain structures, that is, encoding. In a scheduling list $\lambda = (i_1, i_2, ...i_m)$, if every activity in λ is after its all pre-ordered activities, we call λ "precedence compatible list". A feasible solution must be a precedence compatible list, which is undoubted. In this paper, the "precedence compatible list" form is used to encode. For example, activity 2 is the pre-ordered activity of activity 1, and in a feasible individual L1, activity 2 must be before activity 1. In the before-mentioned example, $L_1 = (1,3,2,4,5,6,7,8)$ is a feasible schedule.

Individuals of the initial population are composed of two parts, where one is generated based on the priority rules, and the other randomly. Individuals which are based on priority rules usually has a higher fitness, while individuals generated randomly could ensure the diversity of the individuals of initial population. Therefore, the two kind parts can be combined so that diversity and high fitness can both be gained, which will accelerate the evolution process.

3.2 Selection operator

Selection operator of maximizing the fitness value is adopted, and the best individual of the previous generation will be selected to the next generation using the elite-remain strategy. Thus it can be avoid that the optimal individual being destroyed through crossover and mutation operator, which will be benefit to the spread of excellent individual.

3.3 Crossover operator

In this section, a hybrid crossover method will be produced, which is based on the threshold and the two-points-crossover method. The hybrid crossover method is superior to the two methods on capability, so in the following we will introduce it in detail, and an example will be given.

3.3.1 An crossover method based on threshold

Given a threshold μ, and consider one or several activities in a schedule list, if the utilization rate of resources is greater than or equal to the threshold, the sub-list composed of these activities can be named remaining activity sub-list. While two individuals generate the individual belongs to the next generation, the remaining activity sub-list will be remained, and the other activities should be set to the rest position one by one, where the following rules must be obeyed.

- The activity does not belong to any of remaining activity sub-lists;

- The activity has not been put into λ_s ;
- The pre-ordered activities of it have been placed in λ_s .
- Individuals λ_f and λ_m are shown in Fig.3 and Fig.4, which will be used to explain this and the following crossover operator method. In $\lambda_f(1,4,2,6,3,5,7,8)$ and $\lambda_m(1,2,3,4,5,6,7,8)$, supposing $\mu = 0.8$, the over striking parts denote in this period the utilizing of the resource exceeds the threshold μ , which are remaining activity sub-lists. We can gain the result $\lambda_s = (1,2,4,6,3,5,7,8)$ through this crossover method.

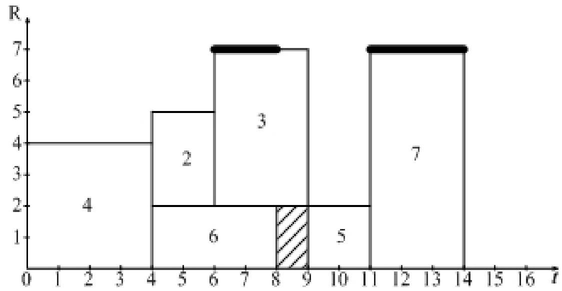

Figure 3. A feasible schedule (λ_f) of the Fig.1

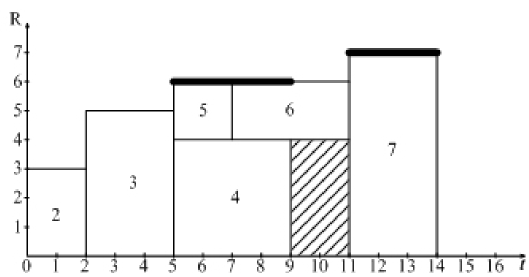

Figure 4. A feasible schedule (λ_m) of the Fig.1

3.3.2 Two-points crossover method

The two-points crossover method can be described as follows: according to crossover probability P_c , select two individuals P_1 and P_2 from the current generation randomly, and then generate two random integers t_1 and t_2 , where $1<t_1<t_2<n$ (n denotes the number of genes in an individual, which is also the number of activities in the project). Two offspring individuals C_1 and C_2 can be generated from P_1 and P_2 by the following operators through t_1 and t_2 .

- For $i = 1$ to t_1, and $i = t_2 + 1$ to n , $C_1[i] := P_1[i]$.
- For $j = 1$ to n , justify one by one, if $P_2[j] \notin C_1$, then $t_1 := t_1 + 1$ and $C_1[t_1] := P_2[j]$.
- The same as before, for $i = 1$ to t_1, and $i = t_2 + 1$ to n , $C_2[i] := P_2[i]$.
- For $j = 1$ to n , justify one by one, if $P_1[j] \notin C_2$, then $t_1 := t_1 + 1$ and $C_2[t_1] := P_1[j]$.

3.3.3 A new hybrid crossover method

This method is the combination of the two crossover method before-mentioned. Firstly, the concept of remaining activity sub-list will be given. Considering of a remaining activity sub-list, if the position of the first activity in the schedule list is i, then the position of this remaining activity sub-list can be defined as i. In the example before-mentioned, the position of remaining activity sub-lists in father individual $\lambda_f(1,4,2,6,3,5,7,8)$ and mother individual $\lambda_m(1,2,3,4,5,6,7,8)$ are different. Thus in the hybrid crossover method, the fixed phases in two-points crossover must be expanded into two remaining activity sub-lists, which can be described by four integers t_1, t_2, t_3 and t_4. Obviously, for father individual λ_f, $t_1 = 5$, $t_2 = 6$, $t_3 = 7$, $t_4 = 7$. For mother individual λ_f, $t_1 = 4$, $t_2 = 6$, $t_3 = 7$, $t_4 = 7$. The algorithm can be described as follows:

- According to the crossover method based on threshold, determine the position of remaining activity sub-lists in the two selected individual, and then mark each remaining activity sub-list using the position of first activity and last activity of it. In this process, several integers will be produced, which can be defined as $t_1, t_2 \ldots, t_m$.
- For each remaining activity sub-list in father individual, after confirming their position in the schedule list, the rest activities can be put into son individual according to the sequence in mother individual.
- For each remaining activity sub-list in mother individual, after confirming their position in the schedule list, the rest activities can be put into daughter individual according to the sequence in father individual.

Experimental results show that the before-mentioned result can be gained according to the hybrid crossover method, and the efficiency is much higher.

3.4 Fitness function

Generally, before determining fitness function of individual, the objective function value of individual is calculated, and then it is transformed into fitness function value. As the final object of RCPSP is to minimize the total duration, the initial objective function must be transformed into fitness function that is Monotone decreasing. In this way, the optimal individual will has the maximal fitness function value, which can be completed as (4):

$$f(\lambda_k) = \frac{O_{\max} - O(\lambda_k) + \gamma}{O_{\max} - O_{\min} + \gamma} \tag{4}$$

where λ_k denotes one of the chromosomes(schedule lists) of current generation, and f(λ_k) denotes the fitness function of λ_k. $O(\lambda_k)$ in (4) denotes the initial objective value of schedule list λ_k, in addition, $O_{\max}$ and $O_{\min}$ respectively represent the maximal and minimal initial objective functional value of current generation. The increment γ is a positive real, which is used to avoid denominator being 0. Furthermore, since the probability of being selected of excellent individuals is far more than poor individuals in the proportion selecting, and it can make the search extension become smaller quickly, the increment γ could make selecting of individual random completely.

4. EXPERIMENTAL RESULTS

For the example of RCPSP before-mentioned, an experiment is done using the new hybrid genetic algorithm. The relevant parameters of GA are as follows: the population size is 20, and Pc=0.5, Pm=0.01. Iterative number is 50. Results are shown in Table 2.

It can be seen from the table that on the basis of resource constraints, the optimal makespan is 12 days. As far as the equilibrium level of resource utilization is concerned, it can be easily concluded from the data obtained, whose visual representation is the same to Fig.2.

Table 2. Experiment results

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Scheduling Order	**1**	**2**	**4**	**3**	**6**	**5**	**7**	**8**
Start Time	**0**	**0**	**4**	**0**	**7**	**4**	**9**	**12**
End Time	**0**	**2**	**7**	**4**	**9**	**9**	**12**	**12**

5. CONCLUSION

In this paper, the simple genetic algorithm is modified. The method based on priority rules and random methods are both adopted to generate initial population, which can ensure not only high adaptability but also diversity of individuals; in addition, a new crossover method is proposed, which is based on two combined crossover methods-crossover based on threshold and two-point crossover. Furthermore, several examples of RCPSP have been settled using the new hybrid genetic algorithm. As a result, the new genetic algorithm is superior to simple genetic algorithm not only in evolution speed but also in the accuracy of answer.

ACKNOWLEDGMENT

This research is supported by the youth research fund of North China Electric Power University (200911024), research of genetic algorithm and matlab simulation solving RCPSP.

REFERENCES

[1] S.X. Liu, M.G. Wang and J.F. Tang, "GA for solving resource-contained project scheduling problem," Journal of system engineering. **17**, 1-7,(2002).

[2] H.L. Zhu. Project schedule management, Tsinghua University Press, Beijing(2002).

[3] X. Zeng and Z.H. Zhang, "Immune-genetic algorithm and its application to resource-constrained project scheduling problem," Journal of Guizhou University(Natural Sciences). **24**, 268-273, (2007).

[4] J. Blazewicz, J.K. Lenstra and A.H.G. Rinnooy Kan, "Scheduling subject to resource constraints: classification and complexity," Discrete Appl. Math. (Netherlands). **5**, 11-24, (1983).

[5] H. Zhang, X. Li, H. Li, et al. "Particle swarm optimization-based schemes for resource-constrained project scheduling," Autom. Constr. (Netherlands), **14**, 393-404, (2005).

[6] R. Heilmann, "A branch-and-bound procedure for the multi-mode resource-constrained project scheduling problem with minimum and maximum time lags", Eur. J. Oper. Res. (Netherlands). **144**, 348-365, (2003).