A Random Key-based Genetic Algorithm Approach for Resource-constrained Project Scheduling Problem with Multiple Modes

I. Okada, X. F. Zhang, H. Y. Yang and S. Fujimura

Abstract—**In the practice of scheduling of construction projects, there is a great variety of methods and procedures that need to be selected at each construction process during project. Accordingly, it is important to consider the different modes that may be selected for an activity in the scheduling of construction projects. In this study, first, we mathematically formulate the resource-constrained project scheduling problem with multiple modes while minimizing the total project time as the objective function. Following, we propose a new random key-based genetic algorithm approach which includes the mode reduction procedures to solve this NP-hard optimization problem. Finally, in order to evaluate the performance of our method, we are scheduled in the close future to implement the proposed approach on some standard project instances as the computational experiment and analyze these experimental results comparing with the bi-population-based genetic algorithm by Peteghem and Vanhoucke [1].**

Index Terms—**Bi-population-based genetic algorithm, Makespan, Random key-based genetic algorithm, Resource-constrained project scheduling problem.**

I. Introduction

The resource-constrained project scheduling problem (rc-PSP) is the optimization problem of which objective is subject to precedence relation between the activities and the limited renewable resource availabilities. In rc-PSP, one of the important objective for project managers is to minimize the total project time, which is referred as makespan. Especially in the case of construction project, the desire of the future owner and the architect, the construction technology used by builders and the building environments affect the procedures and/or methods used during the construction. Therefore, a lot of methods and/or procedures exist in each construction process. In order to deal with this issue, it is extremely essential that the methods and/or procedures applied for an activity can be treated as different several modes that can be selected in the scheduling of construction projects. In this context, rc-PSP problem is extended to a more realistic model, the resource-constrained project scheduling problem with multiple modes (rc-PSP/mM).

In rc-PSP/mM, each activity can be performed in one of several modes. Each mode of an activity represents an alternative way of combining different levels of resource requirements with a related duration. In such problems, according to Hartmann [2], the resources are categorized as renewable, non-renewable and doubly constrained resources. While renewable resources have limited per-period availability such as manpower and machines, non-renewable resources have an availability limited for the entire project, such as a budget for the project. Doubly constrained resources are limited both for each period and for the whole project. However, since doubly constrained resources can simply be considered as a combination of renewable and non-renewable resources, we do not consider them explicitly.

The rc-PSP is an NP-hard combinatorial optimization problem [3]. Since the pioneering work of Kelley [4], the resource-constrained project scheduling problem has been addressed by a great number of researchers. The meta-heuristic solution techniques have been gaining the great attention of researchers [5]. In recent years, among meta-heuristic solution techniques, the systematical report in application of genetic algorithms (GA) to solve the NP-hard combinatorial optimization problems, especially designing problem in engineering, are continuously increasing since they have been proven to be efficient [6]-[7]. Among GAs, random key-based GA (rkGA) is proven to be easily implemented with powerful search capability.

In this study, as a representation of the real-world project problems, we consider the rc-PSP/mM with minimization of the makespan as objective, and to solve this rc-PSP/mM as NP-hard optimization problem, we propose a new random key-based GA approach (rkGA) and the reduction procedure, which is very useful for reducing the large-sized problems into a more acceptable size.

The rest of this paper is organized as follows. In section II, the rc-PSP/mM is defined and the mathematical model of rc-PSP/mM is constructed. In Section III, proposed rkGA for solving the rc-PSP/mM is introduced. In Section IV, concluding remarks and future research directions are given as conclusion.

Manuscript received January 9, 2010.

Ikutaro Okada is with Department of Management and Business, Kinki University, Iizuka, Fukuoka 820-8555 Japan (corresponding author to provide phone: +81-948-22-5655(Ext.465); fax: +81-948-23-0536; e-mail: iokada@ fuk.kindai.ac.jp).

Xiao Fu Zhang, is with Graduate School of Information, Production & Systems, Waseda University, Kitakyushu, Fukuoka 808-0135 Japan (e-mail: zhang-xf@fuji.waseda.jp).

Hong Yu Yang is with Graduate School of Information, Production & Systems, Waseda University, Kitakyushu, Fukuoka 808-0135 Japan (e-mail: young@ruri.waseda.jp).

Shigeru Fujimura is with with Graduate School of Information, Production & Systems, Waseda University, Kitakyushu, Fukuoka 808-0135 Japan (e-mail: fujimura@waseda.jp).

II. Resource-Constrained Project Scheduling Model with Multiple Modes

A. *Background Information*

When there are several different modes that may be selected for an activity in rc-PSP, the problem is called as resource-constrained project scheduling problem with multiple modes [5]. In this problem, each mode corresponds to a different time and resource trade-off option for the activity under consideration. A feasible schedule specifies the implementation mode, as well as the start and finish times for each activity. The rc-PSP/mM can be stated as follows: A project consists of a set of activities $V = \{0, \dots, I+1\}$ where each activity has to be processed in exactly one of several modes. The dummy activities 0 and I+1 represent the beginning and the termination of the project, respectively. There are two kinds of resources, *i.e.*, renewable resources with K types and non-renewable resources with N types available in limited quantities to process activities. Each activity i consumes r^{ρ}_{ijk} units of renewable resource k during each unit period of its processing time and r^{ν}_{ijn} units of non-renewable resource n when mode j is used, where $r^{\rho}_{0jk} = r^{\rho}_{I+1jk} = 0$ $(r^{\nu}_{0jk} = r^{\nu}_{I+1jk} = 0)$. The maximum-limited availability of each renewable resource type k in each time period is a^{ρ}_{k} units, k =1, …, K. The maximum-limited total availability of each non-renewable resource type n is a^{ν}_{n} units, n =1, …, N. The processing time of activity i of selected mode j is denoted by p_{ij} where $p_{0j} = p_{i+1j} = 0$. All parameters are assumed to be non-negative integers. Each activity is performed exactly in one of its modes. Moreover, the activities are interrelated through two kinds of constraints, *i.e.*, precedence constraints and resource constraints, where the former ensures that the activity i in V is not started before all its predecessors have been finished, and the latter ensures that the activities are processed consuming required renewable and non-renewable resources within limited capacities. In Fig. 1, the conceptual model of activity in rc-PSP/mM is illustrated using the notations and indices. The objective of the rc-PSP/mM is to find the precedence and resource feasible finish times for all activities such that the makespan of the project is minimized.

B. *rc-PSP/mM Model*

For the rc-PSP/mM problems considered in this study, the following assumptions are made:

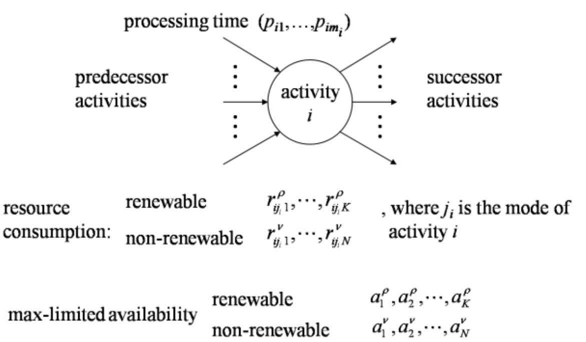

Fig. 1 Concept of activity in rc-PSP/mM model

A1. A single project consists of a number of activities with known processing time and multiple resources required to process activities.

A2. The processing times of activities are deterministic.

A3. The start time of each activity is dependent upon the completion of some other activities (precedence constraints).

A4. There are two kinds of resources, *i.e.*, renewable and non-renewable resources available in limited quantities.

A5. There is no substitution between resources.

A6. Activities can’t be interrupted.

A7. There is one or more than one execution mode for each activity. Mode represents the method and/or procedure to perform that activity.

A8. Each activity must be performed in a mode, where each activity-mode combination has a fixed duration and requires a constant amount of one or more types of renewable resources and non-renewable resources.

A9. The managerial objective is to minimize the total project time.

Table 1 presents the data set for a rc-PSP/mM model, which contains 7 activities including dummy activities (dummy beginning and dummy termination activity). That is, manpower, cost and material, and predecessors of each activity are given in Table 1. Using this data set, the precedence graph in Fig. 2 is constructed.

In order to formulate the mathematical model, the following indices, parameters and decision variables are introduced:

Table 1. Data set of the rc-PSP/mM model

Activity i	Mode j	Duration p_{ij}	Resource consumptions			Pred. of activities $Pre(i)$
			Non-renewable resource 1 r^{ν}_{ij1}	Non-renewable resource 2 r^{ν}_{ij2}	Renewable resource 1 r^{ρ}_{ij1}	
0	(Dummy activity)					
1	1	12	3	5	3	0
	2	15	4	4	2	
	3	18	3	3	4	
2	1	5	2	5	3	0
	2	11	5	2	4	
	3	13	4	3	2	
3	1	5	4	2	3	0
	2	14	5	4	2	
4	1	15	2	3	4	1, 2
	2	12	5	2	3	
	3	8	4	3	2	
5	1	13	5	3	3	1, 3
	2	12	6	4	2	
	3	15	2	3	3	
6	(Dummy activity)					1, 4, 5

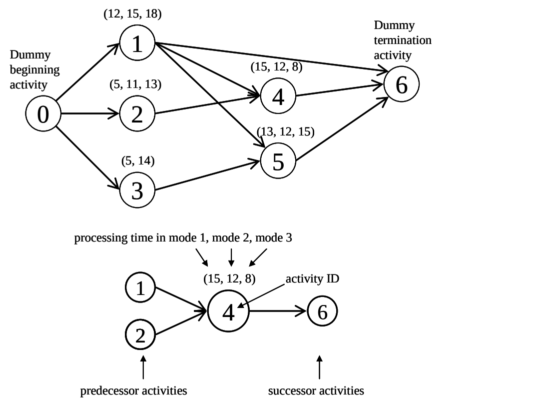

Fig. 2. Precedence graph of the rc-PSP/mM model

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Indices

i: activity index, i = 0, 1, 2, …, I, I+1.

j: mode index, j = 1, 2, …, m_i where m_i is the number of possible modes for activity i.

k: renewable resource type index, k = 1, 2, …, K.

n: non-renewable resource type index, n = 1, 2, …, N.

T: horizon, *i.e.*, upper bound on the project makespan which can be determined by the sum of the maximal activity durations.

Parameters

$Pre(i)$: set of immediate predecessors of activity i.

r_{ijk}^{ρ} : per-period amount of renewable resource k required to execute activity i when mode j is used.

r_{ijn}^{ν} : amount of non-renewable resource n required to execute activity i when mode j is used.

a_k^{ρ} : maximum-limited renewable resource k only available with the constant period availability.

a_n^{ν} : maximum-limited non-renewable resource n with the availability for the entire project .

p_{ij}: processing time of activity i of selected mode j.

Decision variables

$$x_{ijt} = \begin{cases} 1, & \text{if activity } i \text{ is executed in mode } j \\ & \text{and scheduled to be finished in time } t \\ 0, & \text{otherwise} \end{cases}$$

Mathematical model of rc-PSP/mM

The mathematical model for rc-PSP/mM can be stated as follows:

$$\min\ t_M = \sum_{j=1}^{m_{I+1}} \sum_{t=1}^{T} t x_{I+1,j,t} \tag{1}$$

$$\text{s.t.} \sum_{j=1}^{m_i} \sum_{t=1}^{T} x_{ijt} = 1,\ \forall i \tag{2}$$

$$\max_{e \in Pre(i)} \left(\sum_{j=1}^{m_e} \sum_{t=1}^{T} t x_{ejt} \right) + \sum_{j=1}^{m_i} \sum_{t=1}^{T} p_{ij} x_{ijt} \le \sum_{j=1}^{m_i} \sum_{t=1}^{T} t x_{ijt},\ \forall i \tag{3}$$

$$\sum_{i=1}^{I} \sum_{j=1}^{m_i} \sum_{s=t}^{t+p_{ij}-1} r_{ijk}^{\rho} x_{ijs} \le a_k^{\rho},\ k = 1,2,\cdots,K,\ \forall t \tag{4}$$

$$\sum_{i=1}^{I} \sum_{j=1}^{m_i} \sum_{t=1}^{T} r_{ijn}^{\nu} x_{ijt} \le a_n^{\nu},\ \ n = 1,2,\cdots,N \tag{5}$$

$$x_{ijt} = 0 \ or\ 1,\ \forall i, j, t \tag{6}$$

In this mathematical model, the objective (1) is to minimize the total project time, which corresponds to the completion time of last activity performed in this project. The constraints given in equations (2)-(6) are used to formulate the general feasibility of the problem. Constraint (2) ensures that each activity is performed exactly in one of its modes. Constraint (3) ensures that none of the precedence constraints is violated. Constraint (4) ensures that the amount of renewable resource k used by all activities does not exceed its limited quantity in any time periods. Constraint (5) limits the total resource consumption of non-renewable resource n to the available amount. Constraint (6) represents the usual integrity restriction.

III. Random Key-based Genetic Algorithm Approach

In this section, we introduce a two-phased solution approach to solve the rc-PSP/mM model corresponding large-sized real world project scheduling problem. The two phases are as follows:

- **Phase 1 - Reduction Procedure:** Preprocessing procedure for reducing search space.
- **Phase 2 – A Random key-based Genetic Algorithm**

A. Phase 1 – Reduction Procedure

In the first phase, we apply the reduction procedure of Sprecher *et al.* [8] in order to reduce search space, before the genetic algorithm is started. This reduction procedure has been introduced in order to accelerate a branch-and-bound algorithm for rc-PSP/mM and excludes those modes which are inefficient or non-executable and those resources which are redundant. They defined a mode to be inefficient if its duration is not shorter and its every resource request is not less than those of another mode of the same activity and a mode to be non-executable if its execution would violate the renewable or non-renewable resource constraints in any schedule. They also defined a non-renewable resource to be redundant if the sum of maximal requests of the activities for this resource does not exceed its availability. Clearly, inefficient and non-executable modes as well as redundant non-renewable resources may be deleted from the project data without affecting the optimal makespan. So by deleting those modes and non-renewable resources, the search space is reduced and consequently the reduction of computation time is expected.

B. Phase 2 – Random Key-based Genetic Algorithm

The second phase of the solution method includes a new rkGA. Originally, GA that was introduced by John Holland in 1975, is a stochastic search algorithm based on the mechanism of natural selection and genetics by using computer simulation. In GA, a set of initial solutions is first expressed as chromosomes, which consist of several genes. Next, the chromosomes are used simultaneously to search for an optimal solution through crossover and mutation. The chromosomes generate offspring. Then, the offspring whose fitness is high is preferentially selected. Through this process, an optimal solution will be found.

Among GAs, rkGA has been proven to be easily implemented with powerful search capability since it had been proposed by Norman and Bean for Job-shop Scheduling Problem [9]. For applications of rkGA, the reader may refer to Gen *et al.* [7], [10]. In the following subsections, overall procedure, genetic representation and initialization, and genetic operators are explained in detail.

Overall rkGA Procedure Let $P(t_{gen})$ and $C(t_{gen})$ be parents and offspring in current generation t_{gen}. The overall procedure of rkGA for solving rc-PSP/mM model is outlined as follows:

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

```
overall procedure: rkGA for rc-PSP/mM
input: reduced problem data, GA parameters
output: the best schedule
begin:
  t_gen ← 0;
  initialize P(t_gen) by random key-based and randomly mode assignment
  encoding routine;
  evaluate P(t_gen) by random key-based and randomly mode assignment
  decoding routine;
  while (not terminating condition) do
    create C(t_gen) from P(t_gen) by routine of convex hull crossover for
    activity priority with one-cut point crossover for activity mode;
    create C(t_gen) from P(t_gen) by routine of swap mutation for activity
    priority and randomly changing activity mode;
    evaluate C(t_gen) by random key-based and randomly mode
    assignment decoding routine;
    select P(t_gen +1) from P(t_gen) and C(t_gen) by elitist selection routine;
    t_gen ←t_gen +1;
  end
  output the best schedule;
end
```

The genetic procedure starts with the initialization of chromosomes as solutions by using random-key based encoding and decoding routines. Later, till the termination condition is met, convex hull crossover for activity priority with one-cut point crossover for activity mode, swap mutation for activity priority and mutation for activity mode which change the mode randomly, and elitist selection are sequentially performed.

Genetic Representation and Initialization Kim *et al.* [11]-[12] adopted priority-based encoding method for solving the single and multiple project scheduling problems. In their researches, the authors proved the effectiveness of proposed priority-based GA by numerical analysis. However, the nature of the priority-based encoding is a kind of permutation representation. Generally, traditional permutation representation will lead to illegal offspring by one-cut point crossover or other simple crossover operators, where some node's priority may be duplicated in the offspring. Some offspring may generate new chromosomes for which it is impossible to inherit the characters of parents and complex crossover process, which gives rise to the process of evolution being retarded and more computation time needed.

Later to overcome these additional shortcomings, Gen and Lin [13] proposed an extended version of priority-based encoding in real number string, *i.e.*, random key-based encoding to solve the shortest path routing problems. The effectiveness of proposed random key-based encoding has been proved in many research areas [7].

In this paper, we adopt this random key-based encoding method, and a special decoding process for solving rc-PSP/mM. In random key-based encoding method, real values are used to represent the alleles. To obtain a permutation from a chromosome, the genes are treated as random keys. Random key-based encoding is a powerful method to represent permutations, particularly, by which traditional crossover operators always produce legal offspring. Moreover, relative and absolute ordering information can be preserved after recombination.

For the representation of rc-PSP/mM models, random key-based encoding for activity sequences and randomly mode assignment for activity mode are used. The genetic representation (v, m) of an individual solution is composed of two chromosomes where the first chromosome v shows the feasible activity sequence and the second chromosome m consists of activity mode assignment. To develop this genetic representation for rc-PSP/mM model, there are two main procedures, *i.e.*, initialization and decoding procedure;

initialization: Creating an activity sequence and activity mode, *i.e.*, generating a random key priority using encoding procedure and an activity mode using randomly mode assignment procedure to each activity.

decoding procedure:
step 1: Decode a feasible activity sequence that satisfies the precedence constraints.
step 2: If the schedule found in step 1 is infeasible with respect to non-renewable resource, improve it by applying the local search procedure of Hartmann [2].
step 3: If the schedule found in step 2 is infeasible regarding renewable resource, transform it into feasible one using the revised serial method.

The detailed explanations of these procedures are as follows:
initialization: Creating an activity sequence and activity mode

Firstly, here we used the position to denote an activity ID and the random real value $v(i)$ to denote the priority associated with the activity i as shown in Fig. 3 for the example rc-PSP/mM project in Fig. 2. In the part of random key-based chromosome in Fig. 3, the value of a gene is a real number exclusively within [1, I+3). The larger the number means the higher priority. The detailed procedure of random key-based encoding is shown in Procedure 1 (see Fig. 4).

Secondly, all activities are assigned with an activity mode by randomly mode assignment procedure which assigns a mode $m(i)$ to each activity i randomly. The second part of Fig. 3 illustrates a randomly assigned mode chromosome obtained by using this encoding procedure.

activity ID i	0	1	2	3	4	5	6	
activity priority $v(i)$	2.18	6.35	7.82	3.65	4.11	1.28	5.65	Random key-based chromosome
activity mode $m(i)$	1	1	2	1	3	1	1	Randomly assigned mode chromosome

Fig. 3. An individual solution composed of random key-based and randomly assigned mode chromosomes

```
procedure 1: random key-based encoding
input: number of activities n = I + 2
output: initial chromosome v(·)
begin
    for i = 0 to n−1
        v(i) ← random[i, i+1);
    for i = 1 to ⌈n/2⌉
        repeat
            j ← random[1, n];
            l ← random[1, n];
        until j ≠ l;
        swap (v(j), v(l));
    output v;
end
```

Fig. 4. Procedure for random key-based encoding

procedure 2 : creating actitvity sequence
input : $Suc(i)$, activity priority $\nu(i)$
output : activity sequence S_A
begin :
$S_A \leftarrow \{0\}$, $\bar{S} \leftarrow \phi$;
$i \leftarrow 0$;
while $(Suc(i) \neq \phi)$ **do**
$S(i) \leftarrow \{ j \in Suc(i) | Pre(j) \subset S_A \}$
$\bar{S} \leftarrow \bar{S} \cup S(i)$;
$i^* \leftarrow \arg\max \{ \nu(l) | l \in \bar{S} \}$;
$\bar{S} \leftarrow \bar{S} \setminus \{i^*\}$;
append i^* to the end of S_A;
$i \leftarrow i^*$;
end
output activity sequence S_A;
end

Fig. 5. Procedure for creating activity sequence

decoding procedures:

Here in step 1, we identify the activity sequence S_A by the random key-based decoding process while considering activity precedence constraints. We define an activity sequence S_A containing the activities among candidates based on the chromosome v which have been already scheduled, and a decision set $\bar{S}$ containing the unscheduled activities which are eligible for scheduling. The procedure of creating activity sequence is shown in Procedure 2 (see Fig. 5).

For the illustration of this decoding procedure, the chromosome found in Fig. 3 is used. Table 2 presents the trace table for the decoding procedure. In the example, we obtained a precedence feasible sequence of S_A = {0, 2, 1, 4, 3, 5, 6}.

In step 2, if the schedule S found in step 1 is infeasible with respect to non-renewable resource, we improve it by applying the local search procedure of Hartmann [2]. For an individual (v, m) where v is a chromosome and m is a mode assignment, let $L^v_n(m)$ denote the leftover capacity of non-renewable resource $n = 1, 2, \ldots, N$, with respect to the mode assignment m, that is,

$$L_n^v(m) = a_n^v - \sum_{i=1}^{I} r_{im(i)n}^v,\ n = 1, \cdots, N. \qquad (7)$$

Then a negative leftover capacity $L^v_n(m) < 0$ implies infeasibility of mode assignment m with respect to non-renewable resource n. Let the number of non-renewable resource units that exceed the capacities be given by

$$L^v(m) = \sum_{\substack{n=1,\cdots,N \\ L_n^v(m)<0}} \left| L_n^v(m) \right| \qquad (8)$$

Then the procedure chooses an activity randomly and for that activity, a different mode is chosen. If the $L^v(m)$ remains the same or decreases, the mode for that activity is changed. This step is repeated until the mode assignment is feasible ($L^v(m)$ = 0) or until J consecutive unsuccessful trials to improve the mode assignment have been made.

Table 2. Trace table for activity sequence

i		$v(i)$	i^*	S_A
	{ }	$v(0)$=2.18	0	S_A={0}
0	{1,2,3}	$v(1)$=6.35, $v(2)$=7.82, $v(3)$=3.65	2	S_A={0, 2}
2	{1,3}	$v(1)$=6.35, $v(3)$=3.65	1	S_A={0, 2, 1}
1	{3,4}	$v(3)$=3.65, $v(4)$=4.11	4	S_A={0, 2, 1, 4}
4	{3}	$v(3)$=3.65	3	S_A={0, 2, 1, 4,3}
3	{5}	$v(5)$=1.28	5	S_A={0, 2, 1, 4, 3, 5}
5	{6}	$v(6)$=5.65	6	S_A={0, 2, 1, 4, 3, 5, 6}

In step 3, if the schedule found in step 2 is infeasible with respect to renewable resource, we transform it into renewable resource feasible one applying the revised serial method [4], [18]. Let $A(t)$ denote the set of activities in progress in period [t, t+1) and let

$$\bar{a}_k^\rho(t) = a_k^\rho - \sum_{i \in A(t)} r_{im(i)k}^\rho,\ k = 1, \cdots, K \qquad (9)$$

be the remaining capacity of renewable resource type k at time instant t. Also let F_i denote the finish time of activity i. Then a schedule is given by a vector of finish times (F_1, … , F_I). Let i_g denote the activity which is selected in iteration g of an execution of the serial schedule generation scheme (serial SGS). Then an execution of the serial SGS can be recorded by a list λ = (i_1, i_2, … , i_I) which prescribes that activity i_g has been scheduled in iteration g. Note that this list is precedence feasible in our case. Let S_g be the schedule set containing the activities which have been already scheduled at iteration g and set $\Omega_g = \{ F_i | \ i \in S_g \}$. Given a list λ, we can give the serial SGS (the revised serial method) for activity list as in procedure 3. The finish time of i is calculated by firstly determining the earliest precedence feasible finish time EF_i and then calculating the earliest (precedence- and) renewable resource-feasible finish time F_i within [EF_i, LF_i], where LF_i denotes the latest finish time as calculated by backward recursion [19] from upper bound of project's finish time T.

procedure 3 : serial schedule generation scheme for activity lists
input : precedence feasible schedules,
output : renewable resource feasible schedules
begin :
$F_0 = 0, S_g = \{0\}$;
For $g = 1$ to I **do**
calculate Ω_g, $\bar{a}_k^\rho(t)(k = 1, \cdots, K; t \in \Omega_g)$;
$i = i_g$;
$EF_i = \max_{h \in Pre(i)} \{F_h\} + p_{ij}$;
$F_i = \min \{ t \in [EF_i - p_{ij}, LF_i - p_{ij}] \cap \Omega_g |$
$r_{ijk}^\rho \leq \bar{a}_k^\rho(\tau), k = 1, \cdots, K, \tau \in [t, t + p_{ij}) \cap \Omega_g \} + p_{ij}$;
$S_{g+1} = S_g \cup \{i\}$;
end
$F_{I+1} = \max_{h \in Pre(I+1)} \{F_h\}$;
output renewable resource feasible schedule
end

Fig. 7 Serial schedule generation scheme for activity list

Genetic Operators In this subsection, the crossover, mutation and selection operators used in the rkGA approach are explained in detail.

Crossover Operator: As crossover operator, *Convex Hull Crossover* [14] for activity priority together with *one-cut point crossover* for mode are used simultaneously for two individuals. Combining two parents' chromosome $v_j(i)$ and $v_k(i)$, Convex Hull Crossover makes two new offspring chromosome $v_j'(i)$ and $v_k'(i)$;

$$v_j' = \lambda_1 v_j(i) + \lambda_2 v_k(i), \quad \forall i \qquad (10)$$

$$v_k' = \lambda_1 v_k(i) + \lambda_2 v_j(i), \quad \forall i \qquad (11)$$

$$(\lambda_1 + \lambda_2 = 1, \quad \lambda_1 > 0 \text{ and } \lambda_2 > 0) \qquad (12)$$

Mutation Operator: As a mutation operator, in this research, we have used the *swap mutation* for activity priority and changed activity mode randomly for rc-PSP/mMs. The swap mutation (SM) operator by Syswerda [20] was used here, which simply selects two positions (genes) at random and swaps them.

Evaluation Function: We denote the makespan of a schedule related to an individual (v, m) as $C_{max}(v, m)$. Makespan is an appropriate fitness value for a single mode

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

rc-PSP. However, using the makespan as fitness value for the rc-PSP/mM is inappropriate since the infeasible schedule can have the same fitness value as the feasible schedule. So, Hartmann [2] defined the fitness function for an individual as follows:

$$f_{HART} = \begin{cases} C_{\max}(\nu,m) & \text{if } L^{\nu}(m)=0, \\ T + L^{\nu}(m) & \text{otherwise,} \end{cases} \quad (13)$$

where T is the upper bound on the project's makespan which is given by the sum of the maximal durations of the activities.

However, Alcaraz *et al*. [21] have shown that two different individual with the same value of $L^{\nu}(m)$ and a different makespan can have the same fitness value f_{HART} and they have defined their fitness function as follows:

$$f_{ALC} = \begin{cases} C_{\max}(\nu,m) & \text{if } L^{\nu}(m)=0, \\ C_{\max}(\nu,m) + \max\_feas\_C_{\max} & \\ \quad -CP^{\min} + L^{\nu}(m) & \text{otherwise,} \end{cases} \quad (14)$$

where $\max\_feas\_C_{max}$ gives the maximal makespan of the feasible schedules related to individuals of the current generation and CP^{min} is the critical path using the minimal duration of each activity. We use this fitness function.

Selection Operator: As a selection operator, the elitist selection, which preserves the best chromosomes in the next generation and overcome the stochastic errors of sampling, is used. With the elitist selection, if the best individual in the current generation is not reproduced into the new generation, one individual is randomly removed from the new population and the best one is added to the population.

IV. Conclusion

In this study, we have designed a novel two-phased solution approach for rc-PSP/mM. In the first phase, we applied the reduction procedure which excludes those modes and resources which may be deleted from the project data without affecting the optimal makespan. By this procedure, the reduction of computation time is expected. In the second phase, we proposed a novel rkGA. The proposed rkGA has two main advantages. The first is the usage of random key-based encoding with randomly mode assignment which is used to represent permutations by which traditional crossover operators always produce legal offspring. The second is the usage of a serial schedule generation scheme with local search procedure which improves the mode selection with respect to non-renewable resource feasibility. For the future research directions, the computational experiments which are supposed to show the effectiveness of the proposed approach, should be shown in the close future. In those experiments, we will implement the proposed approach on some standard project instances as the computational experiments and analyze their results comparing with the bi-population-based genetic algorithm by Peteghem and Vanhoucke [1] which is considered as one of the most powerful meta-heuristics to solve rc-PSP/mM today.

References

[1] V. V. Peteghem and M. Vanhoucke, "A Genetic Algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 201, pp. 409-418, 2009.

[2] S. Hartmann, "Project scheduling with multiple modes : A genetic algorithm," *Annals of Operations Research*, Vol.102, 2001, pp.111-135.

[3] J. Blazewicz, "Complexity of computer scheduling algorithms under resource constraints," *Proc. 1st meeting AFCET-SMF on Applied Mathematics*, 1978, pp. 169-178.

[4] J. Kelley, "The critical path method: Resource planning and scheduling," in Muth and Thompson editors, *Industrial Scheduling*, Prentice Hall, 1963, pp. 347-365.

[5] P. Brucker, A. Drexl, R. Mohring, K. Neumann, and E. Pesch, "Resource-constrained Project Scheduling: Notation, Classification, Model, and Methods," *European Journal of Operational Research*, vol. 112, 1999, pp. 3-41.

[6] M. Gen and R. Cheng, *Genetic Algorithm and Engineering Optimization*, John Wiley and Sons, New York, 2000.

[7] M. Gen, R. Cheng and L. Lin, *Network Models and Optimization: Multiobjective Genetic Algorithm Approach*, Springer, 2008.

[8] A. Sprecher, S. Hartmann, and A. Drexl, "An exact algorithm for project scheduling with multiple modes," *OR Spectrum*, Vol.19, 1997, pp.195-203.

[9] B.A. Norman and J.C. Bean, "A Random keys genetic algorithm for job shop scheduling: unabridged version," *Technical report*, University of Michigan, Ann Arbor, 1995.

[10] M. Gen and L. Lin, "A new approach for shortest path routing problem by random key-based GA," *Proc. of Genetic and Evolutionary Computation Conference*, Seattle, USA, 2006, pp.1411-1412.

[11] K.W. Kim, M. Gen, and G. Yamazaki, "Hybrid genetic algorithm with fuzzy logic for resource-constrained project scheduling," *Applied Soft Computing*, Vol.2, No.3, 2003, pp.174-188.

[12] K.W. Kim, Y.S. Yun, J.M. Yoon, M. Gen, and G. Yamazaki, "Hybrid genetic algorithm with adaptive abilities for resource-constrained multiple project scheduling," *Computers in Industry* 56(2), 2005, pp.143-160.

[13] M. Gen and L. Lin, "A new approach for shortest path routing problem by random key-based GA," *Proc. of Genetic and Evolutionary Computation Conference*, Seattle, USA, 2006, pp.1411-1412.

[14] M. Gen and R. Cheng, *Genetic Algorithm and Engineering Design*, John Wiley and Sons, New York, 1997.

[15] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, Vol.7, No.2, 2003, pp.174-187.

[16] Yun and M. Gen, "Performance Analysis of Adaptive Genetic Algorithms with Fuzzy Logic and Heuristics," *Fuzzy Optimization and Decision Making* 2(2), 2003, pp.161-175.

[17] P.Y. Wang, G.S. Wang, and Z.G. Hu, "Speeding up the search process of genetic algorithm by fuzzy logic," in *Proc. of European Congress on Intelligent techniques and Soft Computing*, 1997, pp.665-671.

[18] R. Kolisch, "Serial and parallel resource-constrained project scheduling method revised: Theory and computation," *European Journal of Operational Research* vol.90, 1996, 320-333.

[19] S. Elmaghraby, *Activity networks*: *Project Planning and control by network models*, John Wiley, New York, 1977.

[20] G. Syswerda, "Scheduling Optimization using Genetic Algorithms," in L. Davis editor, *Handbook of Genetic Algorithms*, pp.332-349, 1991.

[21] J. Alcaraz, C. Maroto, and R. Ruiz, "Solving the mult-imode resource-constrained project scheduling problem with genetic algorithms," *Journal of the Operational Research Society* vol.54, 2003, pp.614-626.