A Solution of Human Resource Allocation Problem in a Case of Hotel Management

Kayoko Murakami
School of Arts and Sciences
Shibaura Institute of Technology
Saitama, Japan
kayoko@shibaura-it.ac.jp

Seren Ozmehmet Tasan
Department of Industrial Engineering
Dokuz Eylul University
Izmir, Turkey
seren.ozmehmet@deu.edu.tr

Mitsuo Gen
Fuzzy Logic Systems Institute
Fukuoka, Japan
mitsuogen@gmail.com

Takashi Oyabu
Graduate School of Strategic Management
Kanazawa Seiryo University
Kanazawa, Japan
oyabu@seiryo-u.ac.jp

Abstract— **The purpose of this study is to optimally allocate the human resources to tasks while minimizing the total daily human resource costs and smoothing the human resource usage. This human resource allocation problem (hRAP) has two kinds of special constraints, i.e. operational precedence and skill constraints in addition to the ordinary constraints. To deal with the multiple objectives and the special constraints, first we designed this hRAP as a network problem and then proposed a Pareto multistage decision-based genetic algorithm (P-mdGA) to solve it. During the evolutionary process of P-mdGA, a Pareto evaluation procedure called generalized Pareto-based scale-independent fitness function approach was used to evaluate the solutions. Additionally, in order to improve the performance of P-mdGA, we used fuzzy logic controller for fine-tuning genetic parameters. Finally, in order to demonstrate the applicability and to evaluate the performance of the proposed approach, P-mdGA was applied to solve a case study in a hotel, where the managers usually need helpful automatic support for effectively allocating hotel staff to hotel tasks.**

Keywords- resource allocation; human resource; operational precedence constraint; skill, genetic algorithm; Pareto evaluation; smoothing resource usage

I. INTRODUCTION

Simply, a human resource allocation problem (hRAP) can be defined as a problem of allocating the human resources to tasks. Ever since, the importance human resource allocation has been recognized in many business fields, a variety of application areas of this problem [1] such as transportation systems (airlines, railways, and buses), health care systems (emergency room and hospitals), protection and emergency services (police and fire and security services), call centers, civic service and utilities, venue management (ground operations at an airport, cargo terminals, casinos, and sporting venues), financial services, hospitality and tourism industries (hotels, tourist resorts, and restaurants), retails, and manufacturing industry can be regularly seen. However, the requirements for human resource allocation vary according to the application area. The success of the system is usually attributed to how to deal with human resource management. It is a key issue to increase the satisfaction and the profit.

Especially in the hospitality and tourism industries, when hotels use their human resources, they need to consider appropriate and effective allocation of the resources to tasks because the hotel's success or failure depends on its human resource management. Therefore, effective human resource allocation is essential for the success of the hotel. Besides the simple allocation constraints, the real-world hRAP usually contains special constraints such as precedence and skill constraints. For precedence constraints, the operational precedence relations among tasks represent the physical relations between tasks, such as breakfast can be served after cooking/preparing the foods. For skill constraints, the skill requirement of a task represents some special skill required to perform that task such as accounting skills are needed to perform the accounting task. When a task requires some skills, a human resource who deals with the task should possess the skills [1]. If a task is appropriately allocated to a human resource, the human resource's performance level of the task will increase.

In order to solve hRAP, many approximation methods such as simulated annealing (SA) [2, 3], tabu search (TA) [4, 5], and genetic algorithms (GA) [6-10] have been introduced. Especially GA were proved to be good methods to solve hRAP, however, some important issues still remain unsolved in hRAP because only a few researchers have dealt with the realistic skill requirement constraints. Moreover, none of them considered the operational precedence constraints, which exist in real-world problems. Our previous study with single objective of minimizing total human resource costs for the hRAP in a hotel, considered operational precedence between subtasks and skill requirement of human resources [11]. The difficulties to seek an optimum human resource allocation come from these constraints.

In this study, we focus our attention on designing an effective Pareto multistage decision-based genetic algorithm (P-mdGA) approach solving bi-objective hRAP in a hotel while considering operational precedence and skill requirement constraints. The objectives considered in this study are the minimization of the total daily human resource costs and the smoothing of the human resource usage. Additionally, the generalized Pareto-based scale-independent fitness function (gpsiff) approach is used to evaluate the solutions and Fuzzy Logic Controller (FLC) is used for fine-tuning of genetic parameters.

II. Bi-objective Human Resource Allocation Problem with Precedence and Skill Constraints

The hRAP considered in this study can be stated as follows: A system consists of a set of subtasks $U = \{1, 2, \ldots, n_i\}$ categorized into tasks $V = \{1,2, \ldots, m\}$ where each subtask has to be processed. The dummy activities s and t represent the start and the end of the system. The processing time of subtask j in task i is denoted by p_{ij} where processing time of dummy subtasks are zero.

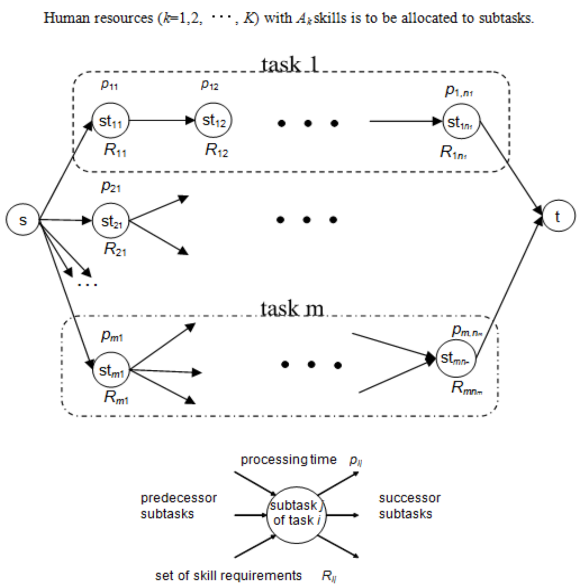

Figure 1. Conceptual hRAP model.

Additionally, the subtasks are interrelated through operational precedence constraint and skill constraint. The operational precedence constraints ensure that the subtask j is not started before all its predecessors have been finished. The skill constraints ensure that there is a possible set of skills R_{ij} which is required to perform subtask j in task i. There are K human resources in the system with a skill set of A_k to be allocated to subtasks.

In Fig. 1, the conceptual model of hRAP is illustrated using the notations and indices.

The minimization of the total resource costs traditionally the main objective. However, smoothing of resources is also an important objective at the same time. Since a good hRAP solution depends on the how much the human resources costs and the balance of human resource usage, in this study, we considered these two objectives together in the context of a bi-objective hRAP. For the resource smoothing, we try to minimize the variance of full-time resources' usage for all activities.

For hRAP considered in this study, the following assumptions are made. Task consists of a set of subtasks, which cannot be interrupted. Subtask processing times are deterministic and do not differ among resources. A subtask should be completed within the required time. Certain subtasks require special skills. An operational precedence constraint exists between subtasks and it is constant. Each subtask must perform at least by predetermined number of resources. Resources are available in limited quantities and reusable type like staff. A resource can be allocated to more than one subtask as long as the subtasks do not overlap in time. Resources have different employment status such as full-time and part-time. Resources can only be used for a limited amount of time and cannot be used overtime. Resources have different skills. Costs of resources are known. Costs for part-time resources are associated with hourly wages. For full-time resources, costs are associated with daily wages.

The following indices, parameters, and decision parameters are used to formulate the mathematical model:

Indices

i: index of task, i=1,2,…, m

j: index of subtask in task i, j=1,2,…l, …, n_i ($l \prec j$)

k: index of staff, k=1,2,…, K

q: index of skill, q=1, 2,…, Q

Parameters

m: total number of tasks

n_i: total number of subtasks in task i

K: total number of staff

Q: total number of skills

Parameters for Subtask:

t_{ij}^S: available starting time of subtask j in task i

t_{ij}^T: available termination time of subtask j in task i

p_{ij}: processing time of subtask j in task i

r_{ijq}: skill q required by subtask j in task i

R_{ij}: the possible set of skills required by subtask j in task i; $r_{ijq} \in R_{ij}$

Parameters for Staff:

w_{ij}: number of staff required for subtask j of task i

c_k: cost of staff k per hour

H_k: maximum hours that staff k can work

$$b_k^1 = \begin{cases} 1, & \text{if staff } k \text{ is part-time (type 1)} \\ 0, & \text{otherwise} \end{cases}$$

$$b_k^2 = \begin{cases} 1, & \text{if staff } k \text{ is full-time (type 2)} \\ 0, & \text{otherwise} \end{cases}$$

$$a_{kq} = \begin{cases} 1, & \text{if staff } k \text{ has skill of } q \\ 0, & \text{otherwise} \end{cases}$$

A_k : skill set of staff k; $a_{kq} \in A_k$

F : set of full-time staff

w_k : total working hours for full-time staff k

w : average total working hours for all full-time staff

Decision Variables

$$x_{ijkq} = \begin{cases} 1, & \text{if staff } k \text{ with skill } q \text{ is assigned to subtask } j \text{ of task } i \\ 0, & \text{otherwise} \end{cases}$$

t_{ij}: actual starting time of subtask j in task i

The mathematical model for the bi-objective mdGA can be stated as follows:

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n_i}\sum_{k=1}^{K}\sum_{q=1}^{Q} b_k^1 p_{ij} c_k x_{ijkq} + \sum_{k=1}^{K} c_k \min\left\{ \sum_{i=1}^{m}\sum_{j=1}^{n_i}\sum_{q=1}^{Q} b_k^2 x_{ijkq},\ 1 \right\} \tag{1}$$

$$\min \ V = \frac{1}{\sum_{k=1}^{K} b_k^2} \sum_{k\in F} \left(w_k - w\right)^2 \tag{2}$$

$$w_k = \sum_{i=1}^{m}\sum_{j=1}^{n_i}\sum_{q=1}^{Q} b_k^2 p_{ij} x_{ijkq},\ k \in F$$

$$w = \frac{1}{\sum_{k=1}^{K} b_k^2} \sum_{k\in F} w_k$$

s. t.

$$t_{il} + p_{il} \le t_{ij}^S,\ \forall i, j, l \tag{3}$$

$$\sum_{j=1}^{n_i}\sum_{k=1}^{K}\sum_{q=1}^{Q} x_{ijkq} \le n_i,\ \forall i \tag{4}$$

$$t_{ij} \ge t_{ij}^S,\ \forall i, j \tag{5}$$

$$t_{ij} + p_{ij} \le t_{ij}^T,\ \forall i, j \tag{6}$$

$$\sum_{l\in L}\sum_{j=1}^{n_i}\sum_{q=1}^{Q} x_{ljkq} \le 1,\ \forall k,\ L = \left\{ l \mid t_{il}^S \le t_{il} + \alpha \le t_{il}^T \right\} \tag{7}$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n_i}\sum_{q=1}^{Q} p_{ij} x_{ijkq} \le H_k,\ \forall k \tag{8}$$

$$\sum_{k=1}^{K}\sum_{r_{ijq}\in R_{ij}} a_{kr_{ijq}} x_{ijkr_{ijq}} = w_{ij},\quad \forall i, j \tag{9}$$

$$x_{ijkq} \in \{0, 1\},\quad \forall i, j, k, q \tag{10}$$

$$t_{ij} \ge 0,\quad \forall i, j \tag{11}$$

In this mathematical formulation, the objective function (1) represents the minimization of total staff costs in a daily schedule. The objective function (2) represents the smoothing of full-time resource usage, where the minimizing the variance of full-time resource usage is considered. Constraint (3) represents the precedence constraints between each subtask. Constraint (4) ensures that all the subtasks in tasks must be done by one staff. Constraint (5) represents the constraint for the start time for each subtask. Constraint (6) ensures that each subtask is completed within the required time. Constraint (7) shows that each staff member deals with only one subtask at a time. Constraint (8) represents resource working-hour limitation. Constraint (9) ensures that each subtask is done by one staff member with required skills.

III. PARETO MULTISTAGE-BASED GENETIC ALGORITHM

In this study, to solve the bi-objective hRAP with special constraints, a Pareto multistage decision-based genetic algorithm approach is proposed. In the P-mdGA, first the hRAP model is constructed and divided into several stages. For hRAP, the subtasks are considered as the stages. The order of the stages represents the sequence of subtasks. Later, possible states of each stage are determined. For hRAP, the states are represented by human resources. To deal with the multiobjective concept, generalized Pareto-based scale-independent fitness function approach is adopted and to adjust the rate of crossover and mutation, fuzzy logic controller is adopted. Let P (t) be parents and C (t) be offspring in current generation t. The overall procedure of P-mdGA can be stated as follows:

```
procedure: P- mdGA for bi-objective hRAP
input: hRAP data, GA parameters
output: Pareto optimal solutions E
begin
  t ←0;
  initialize P(t) by priority-based and resource permutation encoding
        routines;
  calculate objectives f1(P(t)) and f2(P(t)) by priority-based and resource
        permutation decoding routines;
  create Pareto E(P(t));
  evaluate eval(P(t)) by gpsiff routine;
  while (not terminating condition) do
    create C(t) from P(t) by position-based crossover routine;
    create C(t) from P(t) by swap mutation routine;
    calculate objectives f1(C(t)) and f2(C(t)) by priority-based and resource
          permutation decoding routines;
    update Pareto E(P(t),C(t));
    evaluate eval(P(t),C(t)) by gpsiff routine;
    if t > u then
      fine-tuning pM and pC by FLC;
    select P(t+1) from P(t) and C(t) by modified elitist selection routine;
    t ←t+1;
  end
  output Pareto optimal solutions E;
end
```

A. *Genetic Representation*

In order to apply P-mdGA to hRAP effectively, converting the problem model into a chromosome representation is the primary concern [12]. The P-mdGA consists of three phases: creating a feasible subtask sequence, assigning subtasks to resources, and designing a schedule. For hRAP, the individual is composed of two chromosomes. Fig. 2 represents a feasible solution (individual) showing both subtask sequence and resource assignments.

Phase 1: Creating a feasible subtask sequence

step 1.1: Generate a random priority to each subtask in a task by using an encoding procedure : In this step, the priority-based encoding method [13], which is an indirect representation scheme, is used. In this method, the position of a gene represents a subtask node and the value of the gene represents the priority of the subtask node for constructing a schedule among candidates. In Fig. 2, an example output of priority-based chromosome is shown as chromosome v_1. As an initialization, this encoding method is used. After initialization, genetic operators, which are crossover and mutation, are applied in order to make solution candidates. These genetic operators will be explained in detail in the following section.

step 1.2: Decode a feasible subtask sequence T_S that satisfies the operational precedence constraint : After generating priority-based chromosomes in step 1.1, the priorities of each subtask are used to create a feasible subtask sequence that satisfies the precedence constraint in the model. In Fig. 2, a feasible subtask sequence is shown.

Phase 2: Assigning subtasks to resources

step 2.1: Assign each subtask to resources by using resource permutation coding procedure : In this phase, the assignments of subtask to resources are formed using the permutation coding procedure while satisfying the resource skill requirement for each subtask. In Fig. 2, the resource assignments are shown as chromosome v_2.

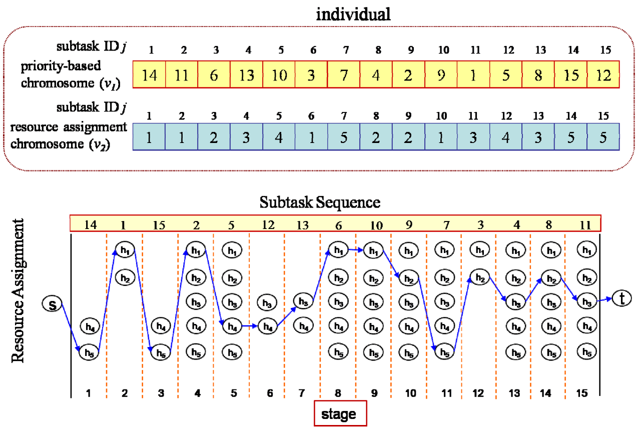

Figure 2. Genetic representation of an individual in P-mdGA.

step 2.2: Obtain a feasible assignment according to the subtask sequence found in step 1.2 and the resource assignment found in step 2.1 : Using the task sequence and the resource permutation encoding, a feasible solution is obtained using the resource permutation decoding procedure. Fig. 2 illustrates the feasible subtask sequence and its resource assignment. After this, next generation will be produced by using the selection operator, which is discussed in the following section.

Phase 3: Designing a schedule

step 3.1: Create a schedule S using the resource assignment to each subtask found in step 2.2 : According to the feasible resource assignment, the schedule using the starting and the termination times of subtask can be constructed as follows:

S = {(st_{11}, h_1: 4:00-6:00), (st_{12}, h_1: 7:00-8:00), (st_{13}, h_2: 15:00-18:00), (st_{14}, h_3: 18:00-19:00), (st_{21}, h_4: 6:00-8:00), (st_{22}, h_1: 8:00-10:00), (st_{23}, h_5: 15:00-17:00), (st_{24}, h_2: 18:00-20:00), (st_{31}, h_2: 10:00-12:00), (st_{32}, h_1: 10:00-12:00), (st_{33}, h_3: 21:00-22:00), (st_{41}, h_4: 8:00-12:00), (st_{42}, h_3: 12:00-16:00), (st_{51}, h_5: 7:00-10:00), (st_{52}, h_5: 12:00-15:00)}

step 3.2: Draw two Gantt charts for this schedule from two points of view: staff schedule and subtask schedule : Final step is to effectively visualize the schedule using two Gantt charts showing the schedule from the human resource and subtask point of view separately.

B. Genetic Operators

In the proposed P-mdGA, the position-based crossover, swap mutation, and modified elitist selection are used as genetic operators for priority-based chromosome v_1.

The position-based crossover is used because the characteristics of gene order in parent chromosomes can be kept while doing this crossover [13]. On the other hand, this crossover does not keep the much of the characteristics of parents like one-cut crossover. During the position-based crossover, some genes are taken from one parent at random, and they fill the same positions in offspring. Then, the vacuum positions in offspring are filled with genes from the other parent by a left-to-right scan.

The swap mutation operator makes the chromosomes a big change in terms of gene position by swapping only two genes [13]. Since the position-based crossover does not change the characteristics of gene position much, this swap mutation is used. In this mutation, two positions are selected at random and their contents are swapped in order to produce random changes in various chromosomes spontaneously.

The modified elitist selection is used [13]. This selection gives GA better search ability in the solution areas without being stuck at local optima. Usually, the elitist selection makes the search area get smaller, which means less variety, by choosing the elitist chromosome. However, this modified version chooses the good one among the best. While preserving the best chromosome in the next generation, this selection gives more diversification to the solution area. Also, this selection overcomes the stochastic errors of sampling. 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 from the current population is added to the new population.

C. Pareto Evaluation with gpsiff Approach

In our bi-objective hRAP model, we have used the generalized Pareto-based scale-independent fitness function approach for the evaluation of the Pareto solutions during P-mdGA. The gpsiff approach makes the use of Pareto dominance relationship to evaluate individuals using a single measure of performance [14]. It uses a pure Pareto-ranking fitness assignment strategy, which differs from the traditional Pareto-ranking methods. Let the fitness value of an individual x be a tournament-like score obtained from all participant individuals by the following Eq. (12).

$$eval(x_i) = p(x_i) - q(x_i) + c,\ i = 1,2,\ldots,popSize \tag{12}$$

where $$p(x_i) = \left\{ \left| x_j \right| \underset{x_i \neq x_j}{f_k(x_j) \geq f_k(x_i)},\ \forall\, k, j,\ (k = 1,2,\ldots,q,\ j = 1,2,\ldots,popSize) \right\}$$

$$q(x_i) = \left\{ \left| x_j \right| \underset{x_i \neq x_j}{f_k(x_i) \geq f_k(x_j)},\ \forall\, k, j,\ (k = 1,2,\ldots,q,\ j = 1,2,\ldots,popSize) \right\}$$

where p is the number of individuals which can be dominated by the individual x, and q is the number of individuals which can dominate the individual x in the objective space. Generally, a constant c can be optionally added in the fitness function to make fitness values positive. In this study, c is the number of all participant individuals.

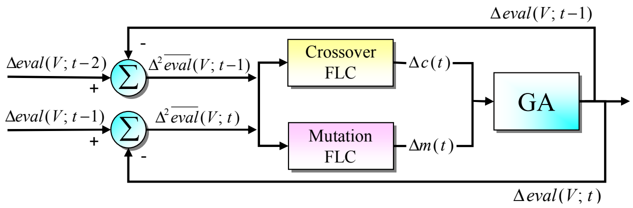

Figure 3. Structure of fuzzy logic controller.

D. *Genetic Parameter Tuning with FLC*

For the regulation of GA parameters, fuzzy logic controller which improves the exploration performance of GA by tuning parameters automatically for each generation and assigning better conditions for exploring optimal solution, has been proved to be very useful. In this research, for the fine-tuning of genetic parameters, Wang *et al.*'s FLC concept [15] is used. This concept consists of two FLCs, *i.e.*, crossover FLC and mutation FLC, which are implemented independently. Using these two FLCs, we adaptively adjust the crossover probability and mutation probability during the optimization process as shown in Fig. 3.

IV. CASE STUDY IN JAPANESE HOTEL SECTOR

In order to show the applicability of the proposed approach, we solved a case study with the data modified from real data collected from several Japanese-style hotels which are particularly called as ryokan. In a ryokan, guests can experience the elements of Japanese culture and customs. Since almost all ryokans provide breakfast/dinner, and also feature common or private hot-spring bathing areas in addition to bathrooms, services and their associated tasks are quite different from normal hotel systems. Additionally, staff can do all the services if he/she have the required service skill. Usually, the full-time staff member working for the ryokan lives within the hotel premises so that the working hours can be more flexible than staff members working for the ordinary hotel systems.

TABLE I. THE LIST OF SKILLS IN THE CASE STUDY

Skill *q*	Skills
1	chef with more than 10 years experience
2	chef with less than 10 years experience
3	foreign language
4	accountant with more than 10 years experience
5	accountant with less than 10 years experience

In this case study, 5 staff members are to be allocated to 15 subtasks. Each subtask requires skills which are defined in Table 1. The information related with subtasks is shown in Table 2. The set of required skills to perform each subtask is shown in the last column of this table. For example in order to perform subtask st_{13}, which requires chef skills, the required skill is skill 1 or skill 2.

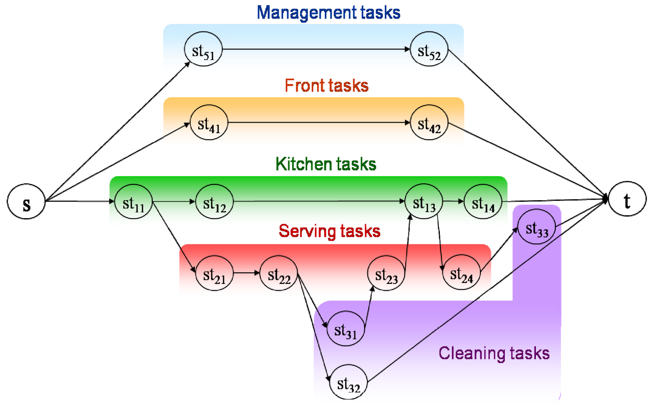

Figure 4. Operational precedence relations in the case study.

The operational precedence relations are shown in Fig. 4. There are 5 staff members. Each staff has their employment status, which are full-time or part-time. They have their own skills. Their wages are different from each other.

In proposed P-mdGA approach without FLC, these values of p_M and p_C are kept constant as the generation is increased. On the other hand, in P-mdGA with FLC, the values of p_M and p_C are adaptively regulated by FLC. Additionally, we compared the performance of the proposed P-mdGA using gpsiff and P-mdGA using adaptive weight approach (awa) [12]. To compare the results for Pareto evaluations, we have used 3 performance measures, *i.e.*, number of obtained solutions $|S_j|$ that counts the number of obtained solution set, ratio of nondominated solutions $R_{NDS}(S_j)$ that counts the number of solutions which are member of the reference solution set, and average distance $D1_R(S_j)$ that calculates the closeness of a obtained solution set from the reference solution set. To calculate the average distance, we define S_j as a solution set (j=1, 2) for each algorithm, *i.e.*, P-mdGA with gpsiff and awa.

Based on the three performance measure found for gpsiff and awa, we can state that the performance of P-mdGA can be improved by using gpsiff. For this bi-objective hRAP, there is a set of solutions that cannot simply be compared with each other. However, in real-world, the human resource managers are asked to select one of the solutions as a best compromised solution. This means that the best compromised solution is dependent on the subjective preference of the manager. For this case by employing P-mdGA with gpsiff, we used the factor weight method, which uses factor weights for cost and smoothing objectives, respectively as the preferences of manager. For this best compromised solution, the total staff costs found to be 243,600 yen.

Based on the computational experiment, the following statements can be made about the proposed P-mdGA approach; P-mdGA has proven to be an efficient solution algorithm for solving hRAP with operational precedence and skill constraints. Additionally, gpsiff can improve the performance of the proposed P-mdGA approach.

TABLE II. THE LIST OF SUBTASKS IN THE CASE STUDY

Task category i	Subtask st_{ij}	Subtask Index	Processing time (h) p_{ij}	Earliest starting time t_{ij}^S	Latest finishing time t_{ij}^F	Successor subtask	Possible set of skills R_{ij}
1: kitchen	st_{11}: breakfast	1	2	4:00	8:00	st_{12}, st_{21}	R_{11}={1,2}
	st_{12}: dish and cleaning (b)	2	1	7:00	10:00	st_{13}	R_{12}={1,2,3,4,5}
	st_{13}: dinner	3	3	15:00	20:00	st_{14}, st_{24}	R_{13}={1,2}
	st_{14}: dish and cleaning (d)	4	1	18:00	22:00	t	R_{14}={1,2,3,4,5}
2: serving	st_{21}: breakfast serving	5	2	6:00	9:00	st_{22}	R_{21}={1,2,3,4,5}
	st_{22}: check-out	6	2	7:00	11:00	st_{31}, st_{32}	R_{22}={1,2,3,4,5}
	st_{23}: check-in	7	2	14:00	21:00	st_{13}	R_{23}={1,2,3,4,5}
	st_{24}: dinner serving	8	2	18:00	21:00	st_{33}	R_{24}={1,2,3,4,5}
3: cleaning	st_{31}: rooms and hallway	9	2	9:00	14:00	st_{23}	R_{31}={1,2,3,4,5}
	st_{32}: laundry	10	2	9:00	15:00	t	R_{32}={1,2,3,4,5}
	st_{33}: bath	11	1	21:00	0:00	t	R_{33}={1,2,3,4,5}
4: front	st_{41}: morning	12	4	8:00	13:00	st_{42}	R_{41}={3}
	st_{42}: afternoon	13	4	12:00	17:00	t	R_{42}={3}
5:manage ment	st_{51}: accountant morning	14	3	7:00	13:00	st_{12}	R_{51}={4,5}
	st_{52}: accountant afternoon	15	3	12:00	19:00	t	R_{52}={4,5}

V. CONCLUSION

In this study, we proposed an effective P-mdGA approach for solving the bi-objective hRAP with operational precedence and skill constraints. Traditionally, many researchers who dealt with this kind of problem either simplified or ignored the operational precedence, which exist between tasks, even though the issue of operational precedence naturally comes up when considering the work in a hotel. During evolutionary process of P-mdGA, gpsiff approach is used to deal with the bi-objective hRAP. The proposed P-mdGA has five main advantages. The first advantage is the appropriate usage of operational precedence relations and skill requirements in multistage concept of hRAP with two chromosomes, i.e. priority-based and resource permutation-based chromosomes. The second advantage is the usage of priority-based chromosome representation, which tries to select the activities according to their priorities and schedule them at its earliest operational precedence. The third advantage is the usage of resource permutation-based chromosome which is used to represent stages, i.e. possible staff allocation solutions particularly using skill requirements of the subtasks and staff skills together. The fourth advantage is the usage of FLC, which is used for auto-tuning of P-mdGA parameters and assigning better conditions to explore the optimal solution. Finally, the fifth advantage is the usage of gpsiff, which is used for evaluating Pareto solutions. According to the case study, it can be stated that the proposed P-mdGA with gpsiff approach has proven to be an efficient solution algorithm.

REFERENCES

[1] A.T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier, “Staff scheduling and rostering: A review of applications, methods and models,” European Journal of Operational Research, vol. 153, pp. 3-27, 2004.

[2] M.J. Brusco, L.W. Jacobs, R.J. Bongiorno, D.V. Lyons, B. Tang, “Improving Personnel Scheduling at Airline Stations,” Operations Research, vol. 43, no. 5, pp. 741-751, 1995.

[3] F.F. Easton, D.F. Rossin, “Overtime schedules for full-time service workers,” Omega, vol. 25, no. 3, pp. 285-299, 1997.

[4] R. Alvarez-Valdes, E. Crespo, J.M. Tamarit, “Labour scheduling at an airport refueling installation,” Journal of the Operational Research Society, vol. 50, no. 3, pp. 211-218, 1999.

[5] J. Gartner, N. Musliu, W. Slany, “Rota: A research project on algorithms for workforce scheduling and shift design optimization,” AI Communications, vol. 14, no. 2, pp. 83-92, 2001.

[6] K. Nammuni, , J. Levine, J. Kingston, “Skill-based resource allocation using genetic algorithms and ontologies,” in Proc. of International Workshop on Intelligent Knowledge Management Techniques, IOS Press, Amsterdam, 2002.

[7] Z.J. Lee, S.F. Su, C.Y. Lee, Y.S. Hung, “A heuristic genetic algorithm for solving resource allocation problems,” Knowledge and Information Systems, vol. 5, pp. 503-511, 2003.

[8] L. Rachmawati, D. Srinivasan, “A hybrid fuzzy evolutionary algorithm for a multi-objective resource allocation problem,” in Proc. of 5th International Conference on Hybrid Intelligent Systems, pp. 55-60, 2005.

[9] M. Yoshimura, Y, Fujimi, K. Izui, S. Nishiwaki, “Decision-making support system for human resource allocation in product development projects,” International Journal of Production Research, vol. 44, no. 5, pp. 831-848, 2006.

[10] C.M. Lin, M. Gen, “Multi-criteria human resource allocation for solving multistage combinatorial optimization problems using multiobjective hybrid genetic algorithm,” Expert Systems with Applications, vol. 34, pp. 2480-2490, 2008.

[11] K. Hirano, S. O. Tasan, M. Gen, T. Oyabu, “Skill-based resource allocation problem by multistage decision-based genetic algorithm,” in Proc. of Asia Conference on Intelligent Manufacturing & Logistics Systems, pp. 463-471, 2008.

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

[13] M. Gen, R. Cheng, Genetic Algorithm and Engineering Optimization, New York, US: John Wiley and Sons, 2000.

[14] S.-Y. Ho, L.-S. Shu, J.-H. Chen, “Intelligent evolutionary algorithms for large parameter optimization problems,” IEEE Trans. on Evolutionary Computation, vol. 8, no. 6, pp. 522–541, 2004.

[15] P.Y. Wang, G.S. Wang, Z.G. Hu, “Speeding up the search process of genetic algorithm by fuzzy logic,” European Congress on Intelligent Techniques & Soft Computing, pp.665-671, 1997.