Multi-Objective Multi-Skill Discrete Time/Resource Trade-off Problem

S. H. Yakhchali[1]

Abstract—The Resource Constraint Project Scheduling Problem (RCPSP) is a critical topic in project management that focuses on finding optimal schedules under precedence and resource constraints. Over the past few decades, researchers have extensively studied RCPSP, leading to the development of various problem variants. One such variant is the Discrete Time/Resource Trade-Off Problem (DTRTP), which considers multiple execution modes for tasks, each with different resource requirements and durations.

This paper explores the classical DTRTP, which aims to minimize project makespan while adhering to resource and precedence constraints. Notable approaches for solving DTRTP include heuristic methods, genetic algorithms, and simulation techniques. Additionally, multi-skilled resource planning plays a crucial role in project scheduling, with various studies addressing the allocation of multi-skilled resources to project activities. This paper provides a comprehensive overview of the existing literature on DTRTP and multi-skilled resource scheduling, highlighting key contributions and methodologies.

***Keywords*—** Heuristic Methods, Genetic Algorithms, and Simulation Techniques.

I. Introduction

Resource constraint project scheduling problem (RCPSP), which is a problem of finding the best schedule under precedence and resource constraints, is one of the most important topics within project management. Resource constraint project schedule problems have been considered by researchers in past few decades since many organizations and companies are dealing with large and complex projects. There are exist various variants of the RCPSP in the literature (the reader may refer to the surveys of (Hartmann & Briskorn, 2010) and (Węglarz, Józefowska, Mika, & Waligóra, 2011).

(Zheng, Wang, & Zheng, 2017) solved a multi-skill RCPSP with the objective of makespan minimization using a teaching–learning-based optimization algorithm (TLBO). They presented a task-resource list-based encoding scheme by combining the resource and task list and used left shift decoding scheme in order to generate feasible schedule. (Othman, Hammadi, & Quilliot, 2015) investigated on scheduling an emergency health care service with multi-skilled medical staff members with the objective of minimizing costs and delays. Their proposed contains an assignment procedure based on fuzzy logic and an evolutionary method in order to schedule the medical staff. (Almeida, Correia, & Saldanha-da-Gama, 2016) proposed the use of parallel scheduling generation scheme to solve a multi-skill resource constrained project scheduling problem.

To the best of our knowledge, there is no literature on Multi-skilled DTRTP considering multi-level for each skill. Thus, in the presented paper a bi-objective Mixed Integer Programming for MS-DTRTP is proposed and afterwards, a solutions approach for small-size, medium-size, and large-size problems is presented.

The remainder of this paper is organized as follows. Problem description and the mathematical formulation and notations are discussed in Section 2. Section 3 describes the solution approach. Numerical results are shown in Section 4, and finally, Section 5 is dedicated to overall conclusions and future research suggestions.

II. Problem description

In this section, a bi-objective mathematical model is introduced for the multi-skilled discrete time/resource trade-off problem (MS-DTRTP). In this problem, several skills are needed for each activity to be accomplished. The following general assumptions are considered in the presented paper:

- The project is represented in an activity-on-node (AON) acyclic network N=(V,P), where V represents the set of activities and E represents the precedence relations between such activities.
- Start activity 0 and finish activity n are dummy ones with zero work content.
- All parameters considered to be deterministic.
- All the resources are renewable and multi-skilled and available from the start time of project.
- Each skill may have several levels which one of them can be used each time.
- Activity pre-emption is not allowed.
- Precedence relationship between activities are defined as finish-to-start (FS) with zero time lags.

The resources which allocated to the activity should be allocated to it until the activities finish time.

The sets and indices, parameters, and decision variables are summarized in following

[1]School of Industrial Engineering, Collage of Engineering, University of Tehran, Tehran, Iran

Sets and indices	
V	Set including all activities
P	Set including all prerequisite relations between activities
$i.j$	Index of activities $i.j \in V$
s	Index of skill types $s \in \{1.2. \dots .S\}$
t	Index of time $t \in \{1.2. \dots .T\}$
r	Index of workforce/resource $r \in \{1.2. \dots .R\}$
S_r	Set of skills of resource r
Parameters	
d_j^l	Lower bound of duration of activity j
d_j^u	Upper bound of duration of activity j
AS_{rs}	Available skill s of resource r ∈ {0,1}
RL_{jsl}	Work content coefficient of level l of skill s for activity j
CR_{rs}	Cost of skill s of resource r
RS_{js}	Required skill s for activity j
AR_r^t	Available amount of resource r in time t
M	A large number
Decision variables	
d_j^*	Duration of activity j
F_j	Finish time of activity j
x_{jrs}^t	1 if skill s of resource r allocate to activity j in time t and 0 otherwise
S_j	Start time of activity j

$$Min\ T = S_n \quad (1)$$

$$Min\ C = \sum_t \sum_j \sum_r \sum_{s \in S_r} x_{jrs}^t\ CR_{rs} \quad (2)$$

Subject to

$$d_j^l \le d_j^* \le d_j^u \qquad j=1,\dots,N \quad (3)$$

$$F_i \le S_j \qquad j=1,\dots,N \text{ and } \forall i \in P_j \quad (4)$$

$$F_j = S_j + d_j^* \qquad \forall j \quad (5)$$

$$\sum_t \sum_r x_{jrs}^t\ AS_{rs}\ RL_{jsl} \ge RS_{js} \qquad \forall j,s \quad (6)$$

$$\sum_j \sum_{s \in S_r} x_{jrs}^t \le 1 \qquad \forall t,r \quad (7)$$

$$\sum_t x_{jrs}^t = d_j^* \qquad \forall j,r,s \quad (8)$$

$$\sum_j \sum_{s \in S_r} x_{jrs}^t \le AR_r^t \qquad \forall t,r \quad (9)$$

$$x_{jrs}^{t+2} \le x_{jrs}^{t+1} + (1 - x_{jrs}^t)M \qquad \forall t = 1,\dots,T-2, \forall r,j,s \quad (10)$$

$$x_{jrs}^t \in \{0,1\},\ \ S_j, F_j, d_j^*: Integer \qquad \forall t,j,r,s \quad (11)$$

Objective functions (1) and (2) minimizes the project's makespan and its total cost respectively. Constraint (3) assures that variable d_j^* not violate its lower and upper bounds. Constraint (4) is for precedence relationships of each activity. Constraint (5) determines each activity finish time. Constraint (6) makes sure that each activity's skill requirement is satisfied. Constraint (7) assures that in each time period, each resource should assign to at most one activity. Constraint (8) determines the relationship between activity duration and its assignment to skilled resources. Constraint (9) makes sure that activity's usage of resources not exceed resource availability. Constraint (10) do not allow activity preemption. Constraint (11) shows restriction on the variables

III. Solution Approaches

DTRTP as a generalization of the parallel machine problem is strongly NP-hard (Demeulemeester, REYCK, & Herroelen, 2000). Therefore, Multi-skill DTRTP is NP-hard as it is a generalization of DTRTP. Therefore, in order to solve large-size problems, heuristic and metaheuristic approaches should be used. For small-size problems the ε-constraint method and for the large-size problems NSGA-II algorithm is used. These methods are explained in the further subsections

A. The ε-constraint method

Using so-called ε-constraint method, the presented bi-objective model in the previous section can be transferred into a single-objective formulation. The ε-constraint which is one the most popular methods for multi-objective programming methods, is first introduced by Haimes (1971).The ε-constraint method can be used when one objective has a higher importance in comparison to the other objectives. As in the presented paper, the effectiveness of patrolling system has the higher importance than the cost, this method can be used. The ε-constraint has some advantages, which the main advantages are (Mavrotas, 2009):

- For linear problems, the ε-constraint method changes the original feasible region and can produce non-extreme solutions, whereas the weighting method is applied to the original feasible region results to a corner solution. Thus, the weighting method may spend a lot of redundant runs, on the contrary, the ε-constraint exploits almost every run for producing an efficient solution.
- In multi-objective integer and mixed integer programming problems, the ε-constraint method can produce unsupported efficient solutions, while the weighting problem cannot.
- In weighting method, finding appropriate scaling factors has strong influence in obtained results, and there is no specified way to find the scaling factors, but in the ε-constraint method there is no need for scaling factors.
- Unlike the weighting method, the number of generated Pareto optimal solutions can be specified easily in the ε-constraint method by adjusting the number of grid points in each objective function ranges.

In the ε-constraint method, all objectives except for one are transformed to a set of constraints and an upper bound limitation for each is set. Therefore, in order to overcome the complexity of solving a multi-objective model, only one objective of the model is minimized or maximized at a time and the other objectives considered as inequalities constraints. Let us consider a multi-objective function as below:

$$Min_{x \in \chi} \{Z(x) = Z_1(x). Z_2(x). \dots . Z_N(x)\}$$

Where $Z(x)$ is the vector of all objective functions and x is the space of decision variables and χ is the set of feasible solutions. Using the ε-constraint method, the multi-objective problem in Eq. (34) can be transferred into a single-objective Eq. (35) and a set of constraints in Eq. (36). In the new problem, only one objective is to be minimized and the other ones considered as constraints inequalities with upper bounds.

$$Min_{x\in X}\, Z_n(x)$$
$$subject\ to$$
$$Z_i(x) \le \varepsilon_i \qquad \forall i \in \{1.2.\dots.N\} - \{n\}$$

Using the ε-constraint method for our bi-objective model, we keep the makespan objective function as the primary objective function as it has the higher importance than the cost function, and converted the cost objective function into a constraint with an upper bound which is shown as Eq. () and Eq. ()

$$Min\ T = S_n$$
$$subject\ to$$
$$\sum_t \sum_j \sum_r \sum_{s\in S_r} x_{jrs}^t\, CR_{rs} \le \varepsilon$$
$$Constraints\ (3) - (11)$$

B. NSGA II

One of the most powerful metaheuristic approaches is Genetic Algorithm (GA) which is an evolutionary algorithm, that is to generates a set of primary random solutions and improve them in each iteration. The initial population are chosen stochastically and GA operators such as crossover and mutation are used to improve the evaluated solution that is called fitness function. The new set of solution continued to be improved in the next iteration using GA operators, and furthermore, the algorithm will be continued until the expected number of iterations are achieved (M. Srinivas & Patnaik, 1994). Non-dominated Sorting Genetic Algorithm (NSGA) is an algorithm represented based on genetic algorithm to solve multi-objective problems. The algorithm includes a set of Pareto optimal solutions in order to help the decision maker to select among various options. NSGA II which is a modified version of NSGA, applies elitism process in order to opt solutions which are appropriate for the next iteration (N. Srinivas & Deb, 1994).

Possible solutions of the problem which represents as chromosomes, gather as populations and in each iteration, new offspring and mutated solution generates. Afterwards, all the solution are ranked based on their dominating rank and in order to rank non-dominated solutions crowding distance is used. In the next step, the new population of same sized with the first population is chose based on the ranking in each iteration. The pseudo-code of NSGA-II and crowding distance are shown in table and table.

Initialize population

Do non-dominated sorting with crowding distance to rank the individuals

Iter=0

While Iter<MaxIter **do**

Get offspring population form tournament selection, crossover and mutation

Merge parent population and offspring population

Do non-dominated sorting with crowding distance to rank the individuals

Select individuals by the order within the population size

Iter=Iter+1

end while

Report results in the population

The pseudo-code of crowding distance

n: size of the non-dominated solutions in S

m: numver of the objectives

$CD[i]$: the crowding distance of the i^{th} individual

$f[i,j]$: the j^{th} objective value of the i^{th} individual Initialize population

for $i = 1$ to n **do**

$CD[i] = 0$

end for

for $j = 1$ to m **do**

Sort the solution with objective j in ascending order

$CD[1] = CD[n] = inf$

for $i = 2$ to $n-1$ **do**

$$CD[i] = \frac{CD[i]+(f[i+1,j]-f[i-1,j]}{\max(f[\bullet,j])-\min(f[\bullet,j])}$$

end for

end for

IV. Conclusion

To conclude this section on the bi-objective mathematical model for the multi-skilled discrete time/resource trade-off problem (MS-DTRTP), several key assumptions have been established to address the complexity and flexibility inherent in the problem. These assumptions include representing the project as an activity-on-node (AON) acyclic network, considering deterministic parameters, using renewable and multi-skilled resources, and ensuring that activities are non-preemptive with finish-to-start precedence relationships**.**

References

[1] Hartmann, S., & Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research, 207*(1), 1-14. https://doi.org/10.1016/j.ejor.2009.11.005

[2] Othman, S. B., Hammadi, S., & Quilliot, A. (2015). Multi-objective evolutionary for multi-skill health care tasks scheduling. *IFAC-PapersOnLine, 48*(3), 704-709. https://doi.org/10.1016/j.ifacol.2015.06.165

[3] Walter, M., & Zimmermann, J. (2016). Minimizing average project team size given multi-skilled workers with heterogeneous skill levels. *Computers & Operations Research, 70*, 163-179. https://doi.org/10.1016/j.cor.2015.11.011

[4] Węglarz, J., Józefowska, J., Mika, M., & Waligóra, G. (2011). Project scheduling with finite or infinite number of activity processing modes–A survey. *European Journal of Operational Research, 208*(3), 177-205. https://doi.org/10.1016/j.ejor.2010.03.037

[5] Zheng, H.-y., Wang, L., & Zheng, X.-l. (2017). Teaching–learning-based optimization algorithm for multi-skill resource constrained project scheduling problem. *Soft Computing, 21*(6), 1537-1548 https://doi.org/10.1007/s00500-015-1866-3