Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management
Kuala Lumpur, Malaysia, March 8-10, 2016

Simulation of Extended Resource Constrained Scheduling Problem with a Soft Computed Priority Heuristic

Taiba Zahid, Michael Völker, Thorsten Schmidt
Department of Mechanical Engineering
Technical University of Dresden
Germany

***Abstract*—In the recent decade, numerous researchers have proposed extended version of scheduling problem named as multi-mode resource constrained scheduling problem (MMRCSP), claiming it to be a more practical way of capturing production characteristics. However, this flexibility comes with increased complexity. The present work proposes a framework for such multi-mode scheduling problems in order to provide solution via simulation of various heuristics and execution strategies. It allows user to select preferred parameters for transparent scheduling as compared to the commercial scheduling softwares which behave like a black box. As this extended constrained scheduling problem involves decision of activity allocation time and choice of performing method (mode option), the proposed framework provides heuristics for both decisions. The aim of the newly proposed heuristic for listing activity priorities is to incorporate the information of position, slack and size of an activity according to the project network characteristics. Additionally, a non-greedy heuristic has been adopted for selection of a performing method described as mode option. The developed framework seeks to aid decision manager for multiple trade-offs among cost, time and resource utilization in case of an in-exact optimal; a situation widely encountered in the field of operations management.**

***Keywords*—Heuristics; Scheduling; Multi-modes; Simulation**

I. INTRODUCTION

Owing to the multiple skills acquired by resources, researchers have focused on the scheduling problems where activities require specific skill or set of skills to be performed whereas time required for completion of the activity is dependent on the type and/or amount of resource being used. For example, consider a jobshop environment where an activity requires drilling. The production floor has two machines which can perform drilling while one being smaller takes a longer time than the other machine. In another case, consider a number of ordered products which need to be delivered. In this case, the duration to complete this task would depend on the number of workers devoted to this particular task.

In essence, activities can be performed in multiple ways which have been termed often as mode options in literature [1]. These numerous possibilities can be modelled variably, basically depending upon the type of industry. Some researchers cope with this problem by considering it a multi-skill problem where each resource has multiple skills with varying skill levels. For example, a resource A can be used as an expert electrician or a lousy driller while for another resource B, these set of skill levels could be vice versa. Hence, the time required for performing a task in such type of production environment would depend only on the choice of resource and not on the quantity. In contrast, in case of a delivery company, the time required to transport a specific order varies significantly with the amount of dedicated resources. This extended version of modelling aids decision managers to manipulate their limited resources and to maximize resource utilization, additionally, proving to be a more practical way of modelling production environment. On the other hand, it also increases the complexity of the problem by increasing the number of decision variables. This class of problem is declared as NP-hard problem [2] that led researchers to explore efficient inexact methods for providing solutions.

Resource constrained scheduling problem is basically defined as a task sequencing problem with the addition of resource constraints provided by limited availability of resources in any production environment. In the past couple of years, researchers came up with numerous heuristics to solve this extended resource constrained project scheduling problem (RCPSP) with multiple modes. In addition to the decision of allocation of a time slot to perform a task (as required in RCPSP), the choice of mode is essential as well in this extended version of MMRCPSP. The state-of-the-art literature in this area mainly addresses the problem by splitting it into two sub-levels and locating the optimal decision for each stage sequentially. The present study utilizes an interesting approach of proposing decision making for activity time slot and mode options, coupled with each other. Additionally, comparison of well-known priority heuristics in this extended version of constrained scheduling needs attention.

As stated earlier, the previous commercial softwares available behave like a black box where user is not able to select and analyse various parameters for heuristic scheduling. This transparency of scheduler is an essential need since it has great impact on the performance of the baseline schedule. Unfortunately, present commercial softwares are unable to provide this flexibility. The performance of these softwares decreases with increased number of activities and resource types [3] in addition to the fact that most of them do not support the multi-mode option for activities. This paper aims in providing a scheduling tool that compares multiple priority heuristics along with schedule generation scheme for the sake of finding the best match in case of multiple mode options and provides control to the user in order to improve transparency. In addition to the existing priority heuristics available for task scheduling, mode selection and shift rule have been provided in order to use the best mode option under different project instances. Moreover, an additional priority rule named as Critical Index (C.I) has been proposed for ranking activities. This rule is based on the position and slack of the activities calculated through soft computing fuzzy methodology. Simulation experiments have been performed through the tool proposed in order to evaluate the performance of the schedule under the effect of various factors. The aim of the present study is to focus on the working and parameters of proposed framework and comparison of heuristics with a novel generation scheme provided for the MMRCPSP. Details on the development and adaptation of fuzzy inference system are not discussed in this study. The next section provides a background of related work for the sake of readers. In the later sections, after a brief problem description, procedure adopted for simulation and scheduling is explained in detail followed by simulation experiments and drawn conclusions.

II. Related Work

Priority rule based heuristics for scheduling and planning have been explored along with meta-heuristics over the past couple of years for resource constrained scheduling [4]. One of the earliest researches on priority rule heuristics can be found in [5]. These heuristics are comprised of two main fragments which are ranking of tasks (priority heuristic rules) and schedule generation schemes. A priority rule is deployed in order to rank the activities and is essential for assigning limited resources. In general, priority heuristics are divided into two main characteristics of being greedy or either non-greedy. However, in state-of-the-art literature related to scheduling, researchers have widely focused on greedy heuristics [6]. These heuristic rules can be obtained through the information assembled from project network such as number of predecessors or successors, resource utilization, average duration of the task or it can be acquired via scheduling information such as total slack of the activity. Details for priority rule based heuristics can be found here [7]. Many researchers have proposed, evaluated and compared the effect of priority rules on performance measures [8-10]. In [8] simple priority rules mostly based on slack information were compared while concluding that MINSLK (minimum slack rule) works the best with proposed dataset. In another research conducted [9], priority rule based on resource utilization was deployed where at each decision level, resource availability was checked and then the activities were ranked based on their resource utilization with respect to the available resources at that point of the decision. However, this involved several calculation steps. A similar heuristic was proposed [10] where the activities were allocated resource in a stage-wise fashion. They suggested a new dataset created with the help of Volvo-IT department in Wroclaw. They concluded that resource based priority rule perform the worst and slack based measures were simple in calculations and provide better performance results. In [11], it was deduced that FIFO came out to be the best in case of achieving number of finished products while shortest processing times rule (SPT) works the best with the objective of maximum machine utilization. In a research conducted by Kühn et al. [12], sensitivity analysis was performed to measure the impact of priority rules on various factors of a single performance measure.

Some researchers have compared the standard priority rules in the light of single and multi-pass heuristics. Multi-pass heuristics is a type of sampling method in which priority rules are iteratively checked in each simulation run for the sake of exploring the best for the target value. As most of the researchers have agreed that multi-pass heuristics provide better results [13], they have argued on the matter that whether it is worth to implement it due to the massive increase in computation load [14]. Schedule generation schemes are typically of two types termed as series or activity scanning method [15] and parallel generation schemes [16]. Despite the fact that priority rule based scheduling is composed of priority rules and a generation scheme, where numerous efforts are being made towards the development of priority rules, hardly any attention has been given to the effect of generation schemes on performance measures. In a study conducted by [17] it was concluded that generally parallel algorithms provide non-delay schedules while series generation scheme provides active schedules. In essence, these efforts were found to be limited in their capacity along with the fact that they only concentrate on the resource constrained version of scheduling problem.

As discussed previously, writers have proved this constrained scheduling problem to be categorized as NP-hard and suggested the use of heuristics to find near optimal solutions. However, some decomposition techniques can be found in literature that suggested the use of exact heuristics. The earliest efforts can be found in the works of Danzig and Wolfe [17], Bender [19] and Deckro et al. [20] which are mainly based on block angular structures which models the problem in set of linear equations. In the context of exact approaches which are applied to solve NP Hard problems, implicit enumeration method are the most famous one. The solution procedure used implicit enumeration and computed all feasible schedules but it was shown that for the projects having 50 or more activities, possibility of finding a solution in reasonable computation time was low [21]. Later on, meta-heuristics were proposed in order to optimize these NP-hard problems. A comparison of the priority heuristics was made in a research conducted [22] where GA proved to provide comparable results. In another research [7], it was concluded that the drawback of using scheduling priority heuristics was their less flexibility in providing final results for multiple conflicting objectives like time and cost. Later on, several hybrid techniques were suggested in the recent decade [23-25] which made use of priority rule based heuristics for the generation of initial schedules and afterwards, evolutionary algorithms were utilized to reach a near optimal solution in short computational time. Hence deduced, that although meta-heuristics can provide flexible results, still, priority heuristics are implemented as an obligatory method for the initial feasible solutions. These state-of-the-art researches explore the performance of different priority rules for time and cost analysis of the project. The studies evaluating the performance of these priority heuristics in pair with generation schemes are almost none to be found. Moreover, these comparisons revolve around RCSPSP and the same results are assumed and employed for the MMRCPSP. The research conducted for multiple mode resource constrained problems is primarily concerned with exploring optimization algorithms. As mentioned earlier, the solutions mainly composed of splitting the problem into two sub steps for the decision making of mode selection and allocation of starting time to an activity [26-28]. Some literature was dedicated to simulation studies in this area with the realization of problem size [29-32]. Various priority rules are compared in order to evaluate the performance measures while using the activity scanning method to run the simulation program. The simulation based optimization strategies comprise of two loop strategies where scheduler and optimizer work in interaction with each other. However, almost none of the studies proposed have been performed under the category of MMRCPSP while searching for the best mode available in sequence with the activity time allocation decision.

This study is a step towards providing this analysis in addition to the comparison of well-known priority heuristics along with generation schemes, an area which needs attention. Moreover, since the past researches prove to provide good results for resource constrained problems while working with the slack based or anteriority based priority rules, this study proposes a new heuristic rule for ranking activities by combining these two factors. Researches have proposed that in order to provide reliable robust schedules, near critical path activities should be considered along with critical path activities. This statement also supports the fact that since the problem under discussion has been declared as NP-hard, soft computing methods could prove to be more instrumental in this area. Hence, this simulation based scheduler uses fuzzy logic in order to identify near critical path activities as well and then proposes a heuristics rule based on the soft computing technique of fuzzy logics, for ranking activities with this information. The next section provides in detail description of the problem area and of the proposed scheduling tool followed by simulation experiments.

III. Procedure Description

The problem under consideration can be described as group of activities that need to be performed with limited resources. Activities pre-emption is not allowed and an activity may or may not have multiple possible ways of execution known as mode options. The project network constitutes of precedence constraints among the activities which need to be followed essentially along with non-renewable resource constraints. No hard constraints on the time and budget have been applied in the present scenario and it is left up to the choice of decision manager to choose best option according to the production aims. The aim of the scheduler is not only to provide feasible schedules with best mode options, but to provide comparison between various coupled priority rules with generation schemes. Activities consume resources (cost and time) and are divided into four basic states which are *Active*, *Recess*, *In-process*, and *End state*. Activities jump from one state to another in case of an event. An event is defined in general as a decision point in a generation scheme which would be the earliest finish time from the group of activities currently in the state *In-process.* Due to the limited resources, activities need to be sorted and prioritized for the assignment of resources. Additionally, each activity may or may not be assigned a different resource or different amount of the same resource (depending upon the production model), in case the lowest priced or the shortest duration mode is unavailable (insufficient resources). The algorithm adopted from [33] is further extended in the current research. Fig. 1 below describes the functioning of the developed scheduler, described further below in detail for the readers.

A. Upload Project Network

This step involves inputting project information in excel format. The main parts of this data include activity anteriority, resource, and general information regarding resource requirements for activities.

B. Parameter Setting

This step involves multiple parameter setting options which control the next phase of simulation run. The main contents of this part are as follows.

a) Priority Rule Selection: To resolve conflicts when competing for limited resources, activities are prioritized according to these rules. Several simple rules adopted from literature [9, 10] have been implemented in the scheduler. Among various heuristics rules that have been proposed over the past couple of years, these were chosen based on state-of-the-art literature on their comparison [9]. Apart from these standard rules, another heuristic i.e. critical index (C.I) has been proposed. The details regarding this heuristic rule would be provided for the readers in the later sub-section.

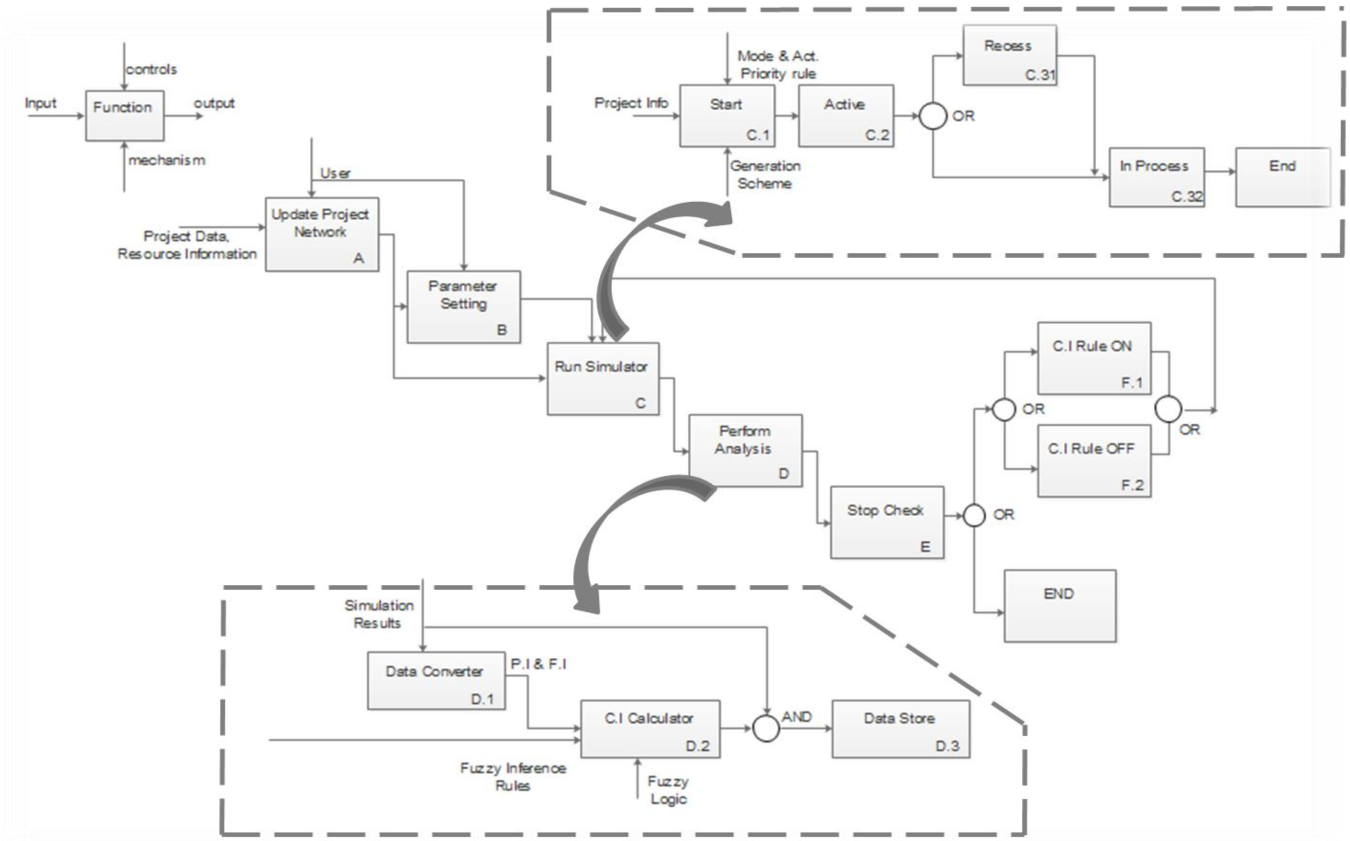


Fig. 1. Proposed Framework for Scheduler

b) Mode Selection: Modes can be ranked based on their cost and duration. At the beginning of simulation, these modes according to the selection are chosen for initial calculations followed by mode shifting rule. The mode shifting rule is based on non-greedy heuristic. This rule is named as conditional mode change rule (CMCR) [33]. According to this heuristic, an activity in remains in state Recess if the waiting time between Recess and Active state is less than the difference of time duration in case of performing the respective activity in another mode (different resource/amount of resource option).

c) Scheme Selection: User can select from two options available which are series and right shift (loosely based on parallel generation) method. Both of the schemes have been modified to incorporate mode selection and conditional shifting rule.

C. Run Simulator

This step uses parameters selected in the previous step as controlling measures to run the scheduler. Completion of the run occurs when every activity of the input project network has reached its *End* state.

D. Perform Analysis

After the scheduling run, defined data values are stored in the analyzer for further processing. In addition to the allocation of time and resources to an activity, its slack and position based information is also stored in the analyser. This information is stored for further calculating C.I of an activity. This Index may or may not be used by the user as a priority rule in the next simulation runs.

Critical Index: In state-of-the-art comparisons on priority heuristics for resource constrained scheduling, it has been proposed several times that heuristics based on float/slack properties derived from critical path without resource constraints tend to deliver good performance [8] in terms of makespan. Furthermore, some writers have advocated the use of position based heuristics such as FIFO, to provide better results [8-10]. However, it may be noted that these procedures work on exact knowledge of float or position of an activity in the project network and disregard near critical path activities. Consequently, the total float of such activities can lead to overestimation of actual free time slots available which may result in delay of the project.

This problem partially arises due to the usual human behaviour according to which work will expand and utilize all the resources (either in the form of time or cost) available. Goldratt [34] proposed the concept of critical chain management to avoid such behaviour and to consider non-critical chain activities as well but many researchers argue on this technique of being arbitrary while the technicians are hesitant to practice [35].

The proposed heuristic uses fuzzy based knowledge to consider the effect of near critical path activities. Moreover, the critical index assigned to each activity for priority heuristics is not solely based on its float but on its position as well. This is to take in account the phenomena which is usually observed and depicts that activities at the end of the project are more susceptible to be delayed, and hence, can prove to be more critical in nature for project's in time completion. Thus a position index (P.I) is assigned to each activity based on its execution time determined by the scheduler during the simulation run. Additionally, a novel float index (F.I) is implied in the present study to take account of the particular characteristics of each individual activity which seemed to be ignored in all other past researches. For instance, a float of 5 time units cannot have the same impact on two activities having duration of 10 and 100 time units. Equations (1) and (2) describe calculation of these indexes.

$$\text{Float Index} = \text{F.I} = 0.5 * \frac{Makespan}{Activity\ Duration} + 0.5 * \frac{Next\ Float}{Activity\ duration} \qquad (1)$$

$$\text{Position Index} = \text{P.I} = \frac{Act\ Position\ in\ project\ chain}{Total\ no.\ of\ activities} * 100 \qquad (2)$$

The analyser separates deterministic values of F.I and P.I among high, medium or low categories according to the predefined boundaries provided by the user. C.I is then computed afterwards from these fuzzy variables with the application of fuzzy inference rules. The purpose of using such categories is to acknowledge the fact that in practice, linguistic variable range such as high and low are more often used by managers rather than exact values, and thus to include near critical path activities. It can be seen from fig. 1 that fuzzy inference rules are provided as an input from the user. For example, if the F.I of an activity is high and P.I is low, then this activity has a low C.I. These fuzzy values of C.I are then assigned a crisp value by centre of gravity method.

The function of analyser ends with storing data values and calculation of above mentioned indexes for each activity. Fuzzy inference rules provided by the user are utilized to determine C.I for each activity. For brevity purposes, details on the fuzzy inference rules and system are not provided here. Readers can refer to [36] for details. Afterwards, these values can adopt two possible ways. Either the C.I index of the activities would be used in the next Simulation Run as priority selection rule OR it will remain the same as previous. This depends (as described in fig. 1) on the C.I rule being ON or OFF. In case, user switches it OFF, the C.I values would be stored for each activity in every run for the evaluation purposes, but would not play a part in sorting priority list for the activities. This option is controlled by user during the phase of parameter setting.

E. Stop Check

The simulation runs are stopped after maximum number of runs or when the output values become constant, i.e. whichever comes first. The later section deals with simulation experiments performed for realizing the effect of various parameters on the performance measures.

IV. EXPERIMENT RESULTS

The scheduler was developed with Python 2.7.10 and the experiments were performed on Windows 7 with an Intel (R) Core (TM) 2 Duo Processor with 3.00 GHz. J20 instances from PSPLib [37] were adopted for the initial tests. The experiments test with different parameters setting of priority rules with respect to the adopted generation scheme can be seen in fig. 2a, 2b and 2c below with time, cost and resource utilization as measures of performance respectively. The selected modes were ranked according to the increasing order of their duration and the C.I rule was tuned ON for all simulation experiments. It was realized that parallel generation scheme tends to provide better results for regular performance measures. The resource utilization (RU) seems to have direct relation with project makespan. However, the same cannot be said for the cost function. As the instance was adopted from PSPLib and thus, was assigned random costs for comparison; it is limited in its capacity for realizing the efficiency of the algorithm and comparison among various performance measures.

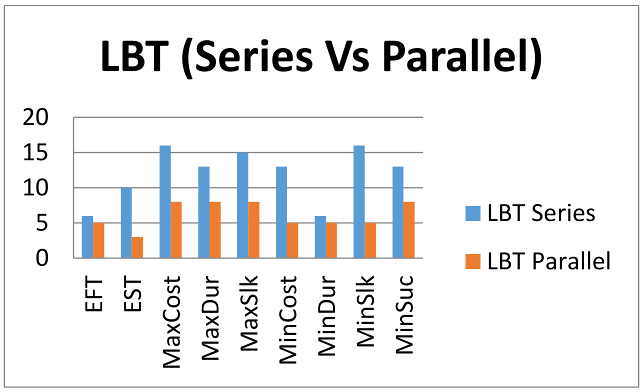


Fig. 2a. Comparison of Time as performance measure

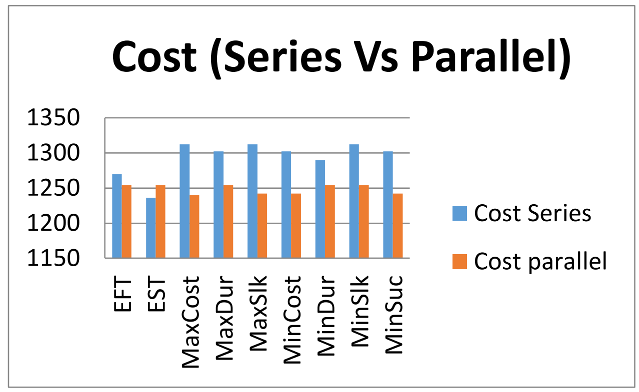
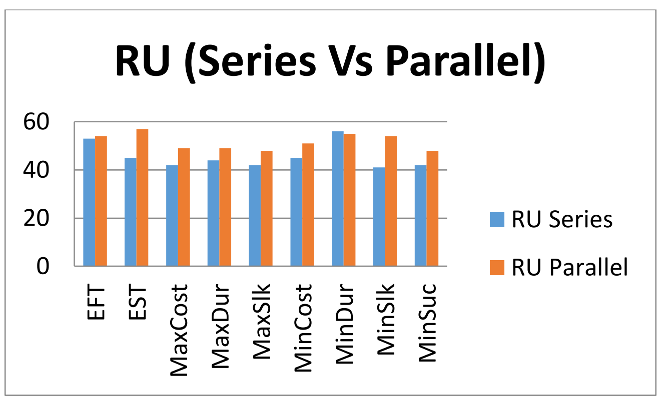


Fig. 2b. Comparison of Cost as performance measure



LBT = Time difference between heuristic result and critical path
RU = Average Resource Utilization
EF/ST = earliest finish/start time
Max/MinSlk = Max/Min slack/float
Max/MinDur = Max/Min duration
MinSuc = Min no. of successors

Fig. 2c. Comparison of Resource Utilization as performance measure

Tests were performed on a case study adopted from a real construction project [38] and produced approximately 850 feasible solutions with various combinations of priority heuristics for mode and activity selection. Figure 3 describes the pareto front solutions which were sorted from the total set of feasible solutions obtained (see Table I for details). The scheduler provided comparable results with the genetic algorithm (GA) proposed for optimization. The best makespan achieved was 190 time units at the cost of 148000 as compared to the results of proposed GA which gives completion time of 190 time units at the cost of 147700. Moreover, it was realized that the heuristics seems to provide better options in some cases for non-dominated solutions as well as providing more solution options which remained unexplored in case of GA.

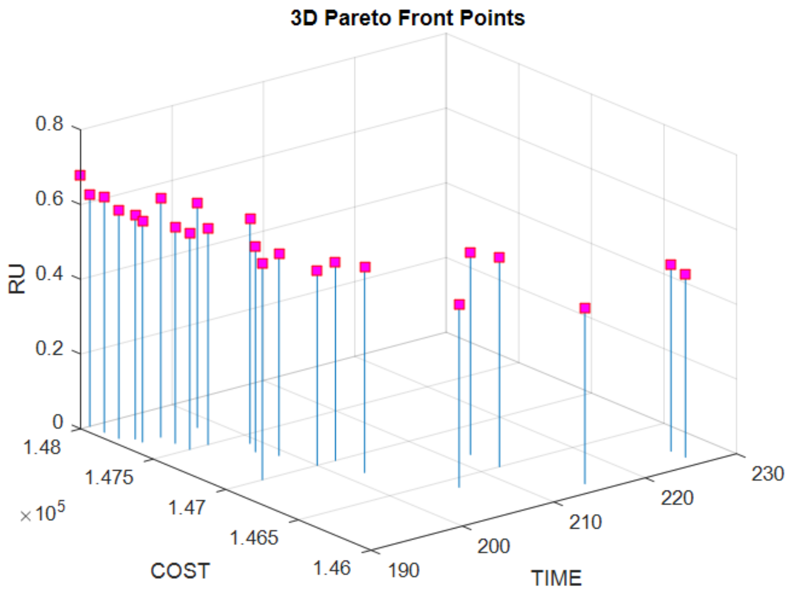


Fig. 3. Performance results for Construction Case Study

It was observed that no conclusion can be drawn for an exact optimal for all three objectives and requires a trade-off analysis. These trade-off points obtained from non-dominated solutions can be analyzed in more clear 2D pictorial views of three objective functions (time, cost and resource utilization) depicted in parts of figure 4. For instance, a decreasing trend can be analyzed between the time and cost function (see figure 4a). The resource utilization (RU) and time graph (figure 4b) depicts a decreasing trend. However, during the selection, the risk factor should be considered which increases with the increase of resource utilization. Hypothetically, 100% use of resources may seems to be tempting as to fully utilize the resource potential but it can lead to worst scenarios since no resource cushion would be present for uncertain disturbance in schedules. It is upto the decision manager to choose according to the needs of production floor among the different optimal options or to assign weightage to normalized value of each objective function as per choice. In general, the best results for minimum makespan were obtained by using minimum float as initial parameter for ranking of activities. For achieving the purpose of minimum cost, it was observed that mode selection criteria should be set for minimal cost as a control parameter.

TABLE I. Performance Measures for the Adopted Case Study [38]

Sol.	Time	Cost	RU	Sol.	Time	Cost	RU
1	190	148000	0,68	13	198	147700	0,6
2	191	147800	0,61	14	199	147200	0,54
3	191	147900	0,63	15	199	147400	0,6
4	191	148000	0,62	16	200	147000	0,52
5	192	147700	0,59	17	202	147000	0,53
6	192	147750	0,6	18	202	146800	0,55
7	194	147700	0,64	19	206	146400	0,49
8	194	147600	0,58	20	212	146700	0,54
9	194	147000	0,57	21	212	146500	0,56
10	194	147500	0,58	22	215	146100	0,47
11	196	147500	0,58	23	226	146100	0,49
12	198	147300	0,55	24	226	146200	0,5

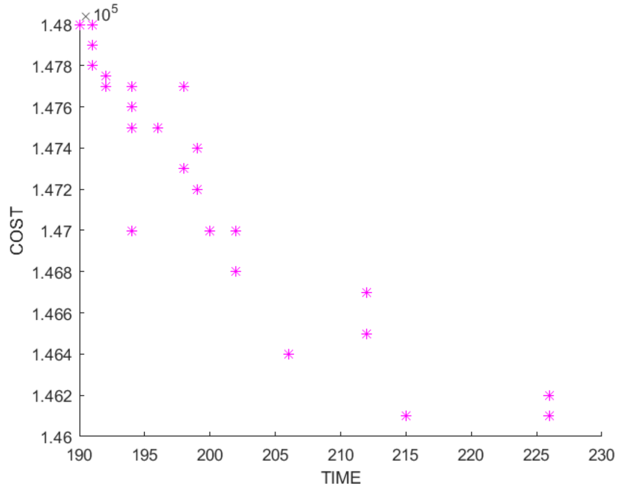


Fig. 4a. Comparison of Time & Cost Pareto Solutions

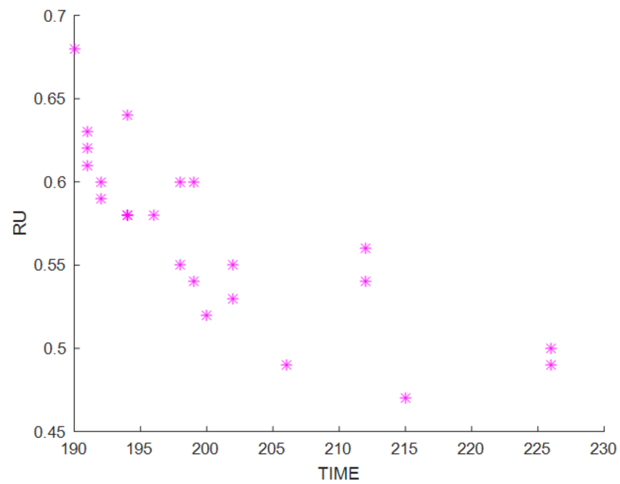


Fig. 4b. Comparison of Time & RU Pareto Solutions

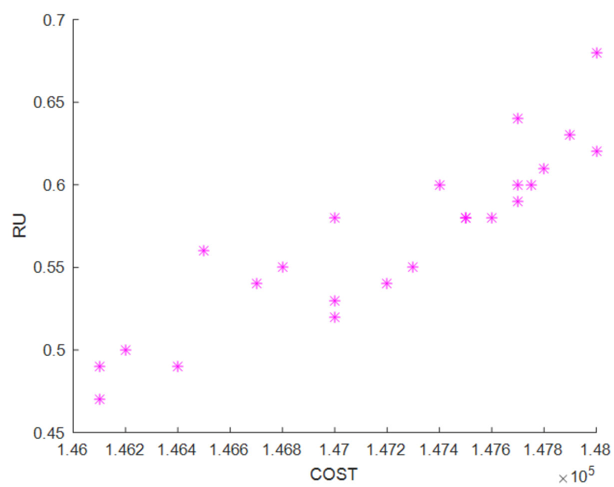


Fig. 4c. Comparison of Cost & RU Pareto Solutions

V. Conclusion & Future Works

This experiments conducted in the present study were intended to realize the compatibility of the proposed scheduler. The initial experiments provide an insight to the utilization of the scheduling heuristic. The heuristics provide near optimal and in some cases optimal results for test instances. The proposed heuristic accomplishes best decision for both the time and mode allocation for the corresponding task in one loop rather than two loop strategy proposed by meta-heuristic optimization algorithms. This characteristic has a potential to save programming effort and significant computation time while dealing with large project data. In addition, it includes near critical path activities in the decision making process. Although it is observed that C.I rule seems to provide slightly better results for regular performance measures (in some cases equal), the non-greedy heuristic of CMCR serves an important role in this extended multi-mode constrained problem in producing better results.

Although, the scheduler proves to provide good results, no conclusion should be drawn for the best heuristic rules. This is due to the fact that solution results tend to vary with network characteristics. The scheduler provides various options for the decision manager to choose the best path under given circumstances. Additionally, these solutions are applicable while considering deterministic project data. For flexible solutions applicable under uncertain disturbances, studies should be conducted in future to find the degree of impact of activities on project's performance measures. Critical index was calculated by assigning fuzzy boundaries to the position and float of each activity. The purpose of this index was not only to suggest a novel heuristic, but also to analyze the nature of criticality of an individual activity based on its position and float. This information would be further evaluated to produce a stochastical model for activity durations in order to increase robustness of the schedule.

References

[1] Kuster, J., Jannach, D. and Friedrich, G., 2007. Handling alternative activities in resource-constrained project scheduling problems. Proceedings of the International Joint Conference on Artificial Intelligence, pp. 1960-1965, Hyderabad, India.

[2] Blazewicz, J., Lenstra, J.K. and Rinnooy Kan, A.H.G., 1983. Scheduling subject to resource constraints: classification and complexity. Discrete Applied Mathematics, 5(1), pp.11-24.

[3] Herroelen, W. and Leus, R., 2001. On the merits and pitfalls of critical chain scheduling. Journal of Operations Management, 19(5), pp. 559-577.

[4] Kolisch, R., 1996. Efficient priority rules for resource-constrained project scheduling problem. Journal of Operations Management, 14(3), pp.79-192.

[5] Conway, R., 1965. Priority dispatching and work-in-process inventory in a job shop. Journal of Industrial Engineering, 16a, pp.123-130. Conway, R., 1965. Priority dispatching and job lateness in a job shop. Journal of Industrial Engineering, 16b, pp.228-237.

[6] Conway, R., Maxwell, W.L. and Miller, L.W., 2003. Theory of Scheduling, Dover Publications Inc.

[7] Myszkowski, P.B., Skowronski, M.E. and Podlodowski, L., 2013. Novel heuristics solutions for multi-skill resource constrained project scheduling problem. Proceedings of Federated Conference on Computer Science and Information System (FedCSIS), IEEE Transactions, pp. 159-166.

[8] Boctor, F.F., 1993. Heuristics for scheduling projects with resource restrictions and several resource-duration modes. International Journal of Production Research, 31(11), pp.2547-2558.

[9] Buddhakulsomsiri, J. and Kim, D.S., 2007. Priority rule-based heuristics for multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting. European Journal of Operational Research, 178(2), pp. 374-390.

[10] Myszkowski, P.B., Skowronski, M.E. and Podlodowski, L., 2013. Novel heuristics solutions for multi-skill resource constrained project scheduling problem. Proceedings of Federated Conference on Computer Science and Information System (FedCSIS), IEEE Transactions, pp. 159-166.

[11] Iringova, M., Vazan, P., Kotianova, J. and Jurovata, D., 2012. The comparison of selected priority rules in flexible manufacturing system. Proceedings of the World Congress on Engineering and Computer Science (WCECS), 24-28 October, San Francisco, USA.

[12] Kühn, M., Völker, M., Angelidis, E., Bohn, D. and Zou, Z., 2015. Sensitivity analysis of a cost objective function to derive criteria fort eh simulation based optimization of planning processes. Simulation and Production in Logistics 2015, Frauenhofer IRB Verlag, Stuttgart, Germany, pp. 89-99.

[13] Kolisch, R., 1995. Project scheduling under resource constrained- Efficient heuristics for several problem classes. Physica-Verlag Heidelberg, Springer Inc.

[14] Yoo, T., Cho, H.B. and Yücesan E., 2010. Enhancing the effectiveness of multi-pass scheduling through optimization via simulation. Artificial intelligence techniques for networked manufacturing enterprise management, chap. 12, pp. 341-365, Springer eries in Advance Manufacturing.

[15] Kelley, J.E., 1963. The critical path method: Resources, Planning and Scheduling. Industrial Scheduling.

[16] Bedworth, D.D. and Bailey, J.E., 1987. Integrated Production Control systems – Management, Analysis, Design. Wiley, New York.

[17] Kolisch, R., 1996. Series and parallel resource-constrained project scheduling problems revisited: theory and computation. European Journal of Operational Research, 90, pp. 320-333.

[18] Dantzig, G.B. and Wolfe, P., 1961. The decomposition algorithm for linear programs. Econometrica, 29(4), pp. 767-778.

[19] Benders, J.F., 1962. Partitioning procedures for solving mixed variable programming problems. Numerische Mathematica, 4, pp. 238-252.

[20] Deckro, R.F., Winkofsky, E.P., Hebert, J.E. and Gagnon, R., 1991. A decomposition approach to multi-project scheduling. European Journal of Operational Research, 51(1), pp. 110-118.

[21] Talbot, B. and Patterson, J.H., 1978. An Efficient Integer Programming Algorithm with Network Cuts for Solving Resource-Constrained Scheduling Problems, Management Science, 24(11), pp. 1163-1174.

[22] Hartmann, S., 1997. A competitive genetic algorithm for resource constrained scheduling. Journal of Naval Search Logistics, 45, pp.733-750.

[23] Valls, V., Ballestin, F. and Quintanilla, S., 2008. A hybrid genetic algorithm for resource constrained project scheduling problem. European Journal of Operational Research, 185(2), pp. 495-508.

[24] Peres, S.D., Roux, W. and Lasserre, J.B., 1998. Multi-resource shop scheduling with resource flexibility. European Journal of Operational Research, 107(2), pp. 289-305.

[25] Ashfar-Nadjafi, B., Rahimi, A. and Karimi, H., 2013. A genetic algorithm for mode identity and the resource constrained project scheduling problem. European Journal of Operational Research, 20(3), pp. 824-831.

[26] Afruzi, E. N., Najafi, A. A., and Roghanian, E., 2013, "A multi-mode resource constrained discrete time-cost tradeoff problem solving using an adjusted fuzzy dominance genetic algorithm," Scientia Iranica E., 20(3), pp. 931-944.

[27] Tseng, L.Y., and Chen, S. C., 2009, "Two-phase genetic local search method for multimode resource constrained project scheduling problem," IEEE Transactions on Evolutionary Computation., 13(4), pp. 848-857.

[28] Weng, H., and Chen, P., 2009, "A two-phase GA model for resource-constrained project scheduling," Automation in construction., 18, pp. 485-498.
[29] Anglidis, E., Bohn, D., Naumann, A., Rose, O. and Carl, S., 2013. Simulation based optimization for complex assembly lines with multi skilled resources. Proceedings of Industrial Systems Engineering Research Conference, 18-22 May, 2013, SanJuan, Puerta Rico.
[30] Zülch, G., Steininger, P., Gamber, T. and Leupold, M., 2009. Generating, benchmarking and simulating production schedules – From formalization to real problems. Proceedings of the Winter Simulation Conference, IEEE Transactions, pp. 2238-2249.
[31] Cheng. J., Fowler, J. and Kempf, K., 2012. Simulation based multi-mode resource constrained project scheduling for semiconductor equipment installation and qualification. Proceedings of the Winter Simulation Conference (WSC), 9-12 December, Berlin, Germany, IEEE Transactions.
[32] Zapata, J.C., Suresh, P. and Reklaitis, G.V., 2006. Assessment of discrete event simulation software for enterprise-wide stochastic decision problems. Manuscript, School of Chemical Engineering, West Lafayetta, USA.
[33] In press: Zahid, T., Völker, M. and Schmidt, T., 2015. A practical algorithmic approach towards multi-modal resource constrained multi-project scheduling problems. Proceedings of the International Conference of Mechanical Engineering and Exposition (IMECE), 13-19 November, 2015, ASME, Houston, Texas.
[34] Goldratt, E.M., 1997. Critical Chain. The North River Press, MA.
[35] Filiatrault, C.L.G. and Peterson, C.D., 2000. Five behaviours that can reduce schedule risk. Proceedings of the Project Management Institute Annual Seminars & Symposium, 7-16 September, 2000, Houston, Texas, USA.
[36] Khanmohammadi, S. and Jassbi, J., 2012. A Fuzzy Approach for Risk Analysis with Application in Project Management. Fuzzy Inference System - Theory and Applications, Dr. Mohammad Fazle Azeem (Ed.), ISBN: 978-953-51-0525-1
[37] Kolisch, R., and Sprecher, A., 1996, "PSPLIB—A Project Scheduling problem library," European Journal of Operational Research., 96, pp. 205-216.
[38] Ghoddousi, P., Eshtehardian, E., Jooybanpour, J. and Javanmardi, A., 2013. Multi-mode resource constrained discrete time-cost.resource optimization in project scheduling using non-dominated sorting genetic algorithm. Automation in Construction, 30, pp.216-227.

BIOGRAPHY

Taiba Zahid is working since 2013 as a research associate in the working group of facility planning at chair of logistics engineering which is a part of department of mechanical engineering of Technische Universität in Dresden. She received her Master's in Mechanical Engineering with specialization in manufacturing area early in 2013. Her master's thesis was about optimizing machine scheduling with meta-heuristic algorithms. She is recently working in a group which focuses on providing practical solutions for industries concerning production management, logistics and supply chain management. Her main research aim is to find robust schedules; insensitive to disruptions and can tolerate uncertainties by remaining close to their optimal solutions.

Prof. Dr. Michael Völker, born in 1956, studied mechanical engineering at Technische Universität Dresden. He received his doctorate in 1988. His doctoral thesis analyses the use of industrial robots for automated machine charging. In the context of various industrial projects he gained experience as a senior project manager in the planning and commissioning of more than 20 factories in different countries. In addition, he gave guest lectures at various universities. He is currently working as head of the factory planning department at Technische Universität Dresden, Faculty of Mechanical Engineering, Chair of Logistics Engineering. His expertise in teaching and research lies in particular in the planning and design of production systems and factories. Core issues are Digital Factory concepts and the organization and optimization of production processes.

Dr. Thorsten Schmidt is full professor at the Technische Universität Dresden and heads the Chair of Logistics Engineering in the Mechanical Engineering faculty since 2008. He holds a diploma degree in mechanical engineering from the Technische Universität Dortmund and a Master's degree in industrial engineering from the Georgia Institute of Technology. He received his Ph.D. from the Technische Universität Dortmund in 2001. His research areas are the design and optimization of facility logistics and production systems including a focus on the machinery and components involved. He currently works on energy efficient control strategies in material flow, fast approximation in early planning stages by means of standard design modules, on-line data analysis, formal verification of control logic, performance analysis of de-central and self-controlled systems, lightweight structures in material handling and stress analysis on wire ropes and toothed belts, respectively.