Event Based Approaches for Solving Multi-mode Resource Constraints Project Scheduling Problem

Ripon Kumar Chakrabortty*, Ruhul Amin Sarker, and Daryl L. Essam

School of Engineering and Information Technology
University of New South Wales, Canberra, Australia
ripon.chakrabortty@student.adfa.edu.au,
ripon_ipebuet@yahoo.com

Abstract. Over the last few decades, a number of mathematical models have been introduced for solving Multi-mode Resource Constrained Project Scheduling Problems (MRCPSPs). However the computational effort required in solving those models depends on the number of variables. In this paper, we attempt to reduce the number of variables required in representing MRCPSPs by formulating two new event-based models. A comparative study was conducted by solving standard benchmark instances using a common objective function for the developed as well as the existing mathematical models. The study provided interesting insights about the problem characteristics, model sizes, solution quality, and computational effort of these approaches.

Keywords: Multi-mode Resource Constrained Project Scheduling Problem, Mixed Integer Linear Programming formulations, Event based formulations.

1 Introduction

The Resource Constrained Project Scheduling Problem (RCPSP) has been a challenging research topic over the last few decades [1]. For it, each activity in RCPSP requires some resources for its processing, such as: machines and tools, human and their skills, raw materials, natural resources (energy, water, land, etc.), information and money. A variant of RCPSP is known as the Multi-mode Resource Constrained Project Scheduling Problem (MRCPSP) and for it each activity can be performed using one out of a set of different resources (known as modes), with specific activity durations and resource requirements. MRCPSP provides the most comprehensive framework for realistically representing the flexibilities that decision makers have [2]. MRCPSPs have many applications within computer integrated manufacturing, planning in make-to-order production, as well as other production planning problems, such as flow shop and job shop scheduling problem. These applications have stimulated many mathematical formulations for the MRCPSPs. From the last decades or so, CPM and PERT were the fundamental techniques to solve project scheduling problems that minimize the project completion time, while satisfying the precedence

* Corresponding author.

K. Saeed and V. Snášel (Eds.): CISIM 2014, LNCS 8838, pp. 375–386, 2014.

relationships among the activities, with an assumption that the resources are unlimited. In reality, as the resources are limited, the formulation of MRCPSPs becomes complex and the formulation varies with the assumptions made.

To solve RCPSPs, integers linear programming (ILP) based approaches are widely used [3]. In the formulation by Pritsker et al. [4], the binary decision variable x_{it} is defined to be one if activity i finishes at time instant t, and to be 0 otherwise. This formulation requires the use of at most nT binary decision variables and $O(n^2 + mT)$ constraints, where n is the number of jobs and m is the number of arcs. Later, Klein and Scholl [5] developed two different models where binary variable x_{it} is defined to be one if activity i is in progress in period t, and to be 0 otherwise. These formulations require nT binary decision variables and $O(n^2T)$ constraints, which is more than the previous one. Therefore, real life projects with many activities are impossible to solve by using exact algorithms, as they involve a large number of variables and constraints. Again, according to Lawler [6], the MRCPSP belongs to $(prec; r_j|C_{max})$ and the solution time is a function of $O(n^2)$, which is further increased exponentially with the modes, number of variables and constraints [7]. Many researchers have tested their approaches by solving benchmark instances from PSPLIB (Project Scheduling Problem Library), with different number of activities, such as 30, 60, 90 and 120 for single mode and 10-30 activities for multi-modes. According to Horroelen and Leus [8] and Alcaraz et al. [9], many of the 60-activity and most of the 90- and 120-activity instances from PSPLIB are beyond the capability of exact methods. Koné et al. [10] has recently indicated that MILP model can deal with up to 25–35 activities within the convenient computational time of 500 seconds. For this reason, any mathematical model that can reduce the number of variables and constraints in representing MRCPSPs would be very practically useful.

Pinto and Grossmann [11] provided an example from a plastic compound plant where the activity duration varied from 0.783 to 11.25 days. The conventional discrete-time formulations are not suitable to tackle this class of problem unless the processing time is rounded or the time horizon is decomposed into smaller time intervals, which only yields approximate solutions. Although Koné et al. [10] presented two event based approaches for single mode RCPSP that can deal with non-integer duration, there is no such model available for MRCPSP. In this regard, the continuous time formulation, along with the event based formulation, can be interesting approaches to study MRCPSPs. From the literature review, it is clear that the trend in MRCPSP research and practice is to formulate the problem as a mathematical model, such as Mixed Integer Linear Programming (MILP), and then solve the model using a conventional optimization technique. However, considering the state of current computational capability, the conventional methods are incapable of solving MILP models with a large number of variables and constraints. In some models, the number of variables increases exponentially with the number of time periods, which encourages the development of alternative mathematical models which by redefining the variables, attempt to minimize the number of variables and constraints.

In this paper, we have formulated two event based models for representing MRCPSPs. For reducing the number of binary decision variables, in those models, we have considered continuous time or time-independent approaches that are known as

‘Start/End Event based formulation (SEE) and ‘On/Off Event based formulation (OOE)’. The proposed formulations are variants of that of Zapata et al. [2], but with fewer variables. Thus, instead of the $3mn^2$ binary variables for Zapata et al. [2], it took $2mn^2$ variables for SEE and mn^2 binary variables for OEE, where m represents number of modes and n represents the number of activities. It was conjectured from the earlier researchers that the solution time of integer programming can be bounded as an exponential function of the number of variables [12]. So, the main intention of this paper is to reduce the number of variables and also the solution times for solving MRCPSP. Our developed mixed integer programming models were solved by using a commercial optimization solver. The proposed model is capable of dealing with MRCPSPs of up to 30-35 activities within reasonable computing time (500 sec). To demonstrate the applicability of the developed models, a number of 10-30 activity multi-mode benchmark instances from the Project Scheduling Library (PSLIB) were solved. The solutions were analyzed for problems with different levels of complexity and are compared with the traditional discrete time approach. The computational studies were also conducted to analyze the effects of different factors that relate to performance.

The structure of the paper is as follows: in section 2, we define the basic MRCPSPs. The terminologies, the proposed event based MILP formulations are described in section 3. In section 4, solution approaches are discussed. The experimental studies, along with the computational results and analysis are provided in section 5. Finally, we provide conclusions in the last section.

2 Problem Description

MRCPSP belongs to the set of combinatorial optimization problems [10]. The MRCPSP under study in this paper is based on the following assumptions: (i) the activities composing a project have certain and known durations that pertain to certain modes; all predecessors must be finished before an activity can start; (iii) resources can be either renewable or both renewable and non-renewable. Renewable resources are available in limited amounts, whereas non-renewable resources are fixed for the whole project; (iv) activities are non-preemptive (i.e., cannot be interrupted when in progress); (v) the main objective is to minimize the project completion time. Let I be the number of activities to be scheduled, R be the number of available renewable resources to be allocated and M_i be the number of modes available for activity i. The activities constituting the project are represented by a set $\{0,..., I+1\}$, where 0 and $I+1$ are the dummy nodes representing the start and end respectively. Accordingly, the set of renewable resources are defined by $\{0,1,...,r\}$, whereas the non-renewable resources are defined by $\{0,1,...,w\}$. The other important notations are depicted in the next section as Nomenclature. The traditional formulations for scheduling are discrete in manner, i.e., it depends on discretization of the time horizon, T, which is represented as $\{0, ..., t\}$. Each resource type has a certain capacity limit which cannot be exceeded throughout the project life.

3 The Mathematical Models

In this section, firstly, we have presented a discrete time approach and then two new formulations are developed.

3.1 Nomenclature

V_w	*Capacity of non-renewable resource w*
U_{irm}	*Resource usage of activity i for resource r at mode m*
L_{iwm}	*Non-renewable resource 'w' usage for activity i on mode m*
d_{im}	*Duration of activity i at mode m*
ES_i, LS_i	*Earliest & latest starting time for activity i*
EF_i , LF_i	*Earliest & Latest Finish time for activity i*
TF_i	*Total Float time of activity i.*
T	*Total planning horizon/upper bound of the project duration.*
P_{prec}	*Represents the precedence set*
Y_k	*Capacity of resource K*

3.2 Discrete Time Approach (DT)

The standard MRCPSP requires sequencing the project activities, so that the precedence constraints are met, the execution mode for each activity is determined, the resource constraints are met and the project duration is minimized. In spite of having several mathematical models for MRCPSP [2] and [13], the model from Talbot [14] is still being used as a basis for discrete time approach and hence is employed throughout this paper as the discrete time (DT) approach for comparing with our proposed models. In the MRCPSP model, a binary decision variable x_{imt} is defined to be 1 if activity i starts at mode *m* at time instant t, and 0 otherwise. The model can be presented as follows:

$$\text{Minimize } Z = \sum_{m\in M_i} \sum_{t\in T} t x_{Nmt}$$

Constraints:

$$\sum_{m}^{M_i} \sum_{t=0}^{T} x_{imt} = 1 \qquad \forall i \in I \tag{1}$$

$$\sum_{m}^{M_i} \sum_{t=0}^{T} t x_{jmt} \geq \sum_{m}^{M_i} \sum_{t=0}^{T} (t + d_{im}) x_{imt} \qquad (i, j) \in P_{prec} \tag{2}$$

$$\sum_{i=1}^{I} \sum_{m\in M_i} \sum_{q=t-d_{im}+1}^{T} U_{irm} x_{imq} \leq K_{rt} \qquad \forall r \in R \ \ and \ \forall t \in T \tag{3}$$

$$\sum_{i=1}^{I} \sum_{m\in M_i} \sum_{t=0}^{T} L_{iwm} x_{imt} \leq V_w \qquad \forall w \in W \ \ and \ \forall t \in T \tag{4}$$

$$f_i = \sum_{m\in M_i} \sum_{t\in T} (t + d_{im})\, x_{imt} \quad \forall i \in I \tag{5}$$

$$x_{imt} \in \{0,1\} \qquad \forall m \in M,\ \forall i \in I,\ t = 1, \dots, T \tag{6}$$

Here, constraint equation (1) represents that every job or activity must be handled exactly one time for all modes. Constraint set (2) ensures that precedence relationships are maintained. Meanwhile constraint sets (3) and (4) respectively represent that the capacity of the renewable and non renewable resources are satisfied. Finally, constraint set (5) defines the finish time of activity i.

3.3 Start/End Event Based Continuous Time Formulations (SEE)

In discrete time formulations, the variables are indexed by time. In contrast to that formulation, in this section, a new event based formulation is proposed for MRCPSP. It is based on the concept of the single mode RCPSP model proposed by Koné et al. [10]. For classical MRCPSP, the numbers of activities are often considered as n+2 (where n is the activity number with 0 & n+1 being the dummy source and sink nodes). On the contrary, this event based formulation considers only (n+1) activities as events. Event-based formulations also have the advantage that they can deal with non-integer activity processing times. More importantly, for instances with long scheduling horizons, event-based models involve fewer variables in comparison to the models indexed by time. Here, we used only two types of binary variables one for start event (x_{ime}) at mode m and the other one is for the end event (y_{ime}) at mode m. A continuous variable t_e represents the date of event e and a continuous ble, r_{emk}, is used for the quantity of resource k that is required immediately after event e at mode m. That is why this proposed approach is termed as a start/end event based formulation (SEE).

$$\text{Min } t_n$$

$$t_0 = 0 \tag{7}$$

$$t_f \geq t_e + p_{im}x_{ime} - p_{im}\left(1 - y_{imf}\right) \quad \forall (e,f) \in \varepsilon^2, f > e, \forall i \in A, \forall m \in M_i \tag{8}$$

$$t_{e+1} \geq t_e \quad \forall e \in \varepsilon, e < n \tag{9}$$

$$\sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{e\epsilon\varepsilon} x_{ime} = 1 \qquad \forall i \in A \tag{10}$$

$$\sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{e\epsilon\varepsilon} y_{ime} = 1 \qquad \forall i \in A \tag{11}$$

$$\sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{\tau=e}^{n} y_{im\tau} + \sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{\tau=0}^{e-1} x_{jm\tau} \leq 1 \qquad \forall (i,j) \in E, \forall e \in \varepsilon \tag{12}$$

$$r_{omk} = \sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{i \in A} b_{imk} x_{im0} \qquad \forall k \in R \tag{13}$$

$$r_{emk} = r_{e-1,\mathrm{mk}} + \sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{i \in A} b_{imk} x_{ime} - \sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{i \in A} b_{imk} y_{ime} \quad \forall e \in \varepsilon, e \geq 1, k \in R \tag{14}$$

$$r_{emk} \leq B_k \quad \forall e \in \varepsilon,\ k \in R,\ \forall m \in M_i \tag{15}$$

$$ES_i x_{ime} \leq t_e \leq \mathrm{LS_i x_{ime}} + \mathrm{LS_{n+1}}(1 - \mathrm{x_{ime}}) \quad \forall \mathrm{i} \in \mathrm{A},\ \forall \mathrm{e} \in \varepsilon, \forall \mathrm{m} \in \mathrm{M_i} \tag{16}$$

$$ES_{n+1} \leq t_n \leq LS_{n+1} \tag{17}$$

$$(ES_i + p_{im})y_{ime} \le t_e \le (LS_i + p_{im})y_{ime} + LS_{n+1}(1 - y_{ime}) \;\; \forall i \in A, \forall e \in \varepsilon, \; \forall m \in M_i \quad (18)$$

$$t_e \ge 0 \;\; \forall e \in \varepsilon \quad (19)$$

$$\sum_{i=1}^{I} \sum_{m \in M_i} \sum_{t_e} L_{iwm} x_{imt} \le V_w \qquad \forall w \in W \;\; and \; \forall t \in T \quad (20)$$

$$r_{emk} \ge 0 \;\; \forall e \in \varepsilon, \; k \in R, \; \forall m \in M_i \quad (21)$$

$$x_{ime} \in \{0,1\}, \; y_{ime} \in \{0,1\} \;\; \forall i \in A \cup \{0, n+1\}, \; \forall e \in \varepsilon, \; \forall m \in M_i \quad (22)$$

Constraint (7) stipulates that event 0 starts at time 0. Inequalities (8) ensure that if activity i starts at event e and ends at event f, then $t_f \ge t_e + p_{im}$. Any other combination of values for x_{ime} and y_{ime} yield either $t_f \ge t_e$ or $t_f \ge t_e - p_{im}$, which are redundant with constraint (9). Constraints (10) and (11) ensure that the start and end events must have a single occurrence for all of its available modes respectively. Constraint (12) describes the precedence relationship between activities. Constraint (13) represents that the total resource demands for each resource starts at event 0. Constraint (14) defines that for each resource *r*, its demand immediately after event *e* is equal to its demand immediately after the previous event *e*-1, plus the demand required by the activities (on any mode *m*) that start at event *e*, minus the demand required by the activities that end at event *e*. Constraint (15) limits the demanded resources at each event to the availability of resources. Constraints (16)-(18) are valid inequalities based on activity time windows. Inequality (20) is for non-renewable resource *w*, while the usages of that resource throughout the whole event date t_e must to be within its maximum available limit.

3.4 On/Off Event Based Continuous Time Formulation (OOE)

In this section, we propose another model that uses only one type of binary variable per event. Similar to the last section, this is also an extension of an earlier model proposed by Koné et al. [10]. The SEE model needs more variables for representing events than doe the OEE model [3]. For it, the number of events is exactly equal to the number of activities. The representation of the resource constraints is also simpler and easier. Again, rather than using tree decision variables for start/end formulation, here in this model only two decision variables are used. A decision variable x_{ime} is set to be 1 if activity i starts at event *e* at mode *m* or if it is still being processed immediately after event *e*. Here, in the same way as in the previous model, a continuous decision variable (t_e) represents the date of event *e*. The OOE formulation is presented below.

$$\text{Min } C_{max}$$

$$\sum_{m}^{M_i} \sum_{e \in \varepsilon} x_{ime} \ge 1 \qquad \forall i \in A \quad (23)$$

$$C_{max} \ge t_e + (x_{ime} - x_{im,e-1})p_{im} \qquad \forall e \in \varepsilon, \forall i \in A, \; \forall m \in M_i \quad (24)$$

$$t_0 = 0 \quad (25)$$

$$t_{e+1} \geq t_e \quad \forall e \neq n-1 \in \varepsilon \tag{26}$$

$$t_f \geq t_e + \left((x_{ime} - x_{im,e-1}) - (x_{imf} - x_{im,f-1}) - 1 \right) p_{im} \quad \forall (e,f,i) \in \varepsilon^2 \times A, f > e,\ \forall m \in M_i \tag{27}$$

$$\sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{e'=0}^{e-1} x_{ime'} \leq e \left(1 - (x_{ime} - x_{im,e-1}) \right) \quad \forall e \in \varepsilon \backslash \{0\} \tag{28}$$

$$\sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{e'=e}^{n-1} x_{ime'} \leq (n-e) \left(1 + (x_{ime} - x_{im,e-1}) \right) \quad \forall e \in \varepsilon \backslash \{0\} \tag{29}$$

$$x_{ime} + \sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{e'=0}^{e} x_{jme'} \leq 1 + (1 - x_{ime}) e \quad \forall e \in \varepsilon, \forall (i,j) \in E \tag{30}$$

$$\sum_{\mathrm{m}}^{\mathrm{M_i}} \sum_{i=0}^{n-1} b_{imk} x_{ime} \leq B_k \quad \forall e \in \varepsilon,\ \forall k \in R \tag{31}$$

$$ES_i x_{ime} \leq t_e \leq LS_i (x_{ime} - x_{im,e-1}) + LS_n \left(1 - (x_{ime} - x_{im,e-1}) \right) \ \forall e \in \epsilon, \forall i \in A, \forall m \in M_i \tag{32}$$

$$ES_{n+1} \leq C_{max} \leq LS_{n+1} \quad \forall e \in \varepsilon, \forall i \in A \tag{33}$$

$$t_e \geq 0 \quad \forall e \in \varepsilon \tag{34}$$

$$x_{ime} \in \{0,1\} \ \forall i \in A, \forall e \in \varepsilon,\ \forall m \in M_i \tag{35}$$

The objective is once again to minimize the makespan. Constraints (23) ensure that each activity is processed at least once during the project. Constraints (24) build up the relationship between the makespan and the event dates: $C_{max} \geq t_e + p_i$ if i is in process at event e but not at event e-1, i.e., if i starts at event e. Constraints (25) and (26) ensure the basic definition of event sequencing. Constraints (27) link the binary optimization variable x_{ie} to the continuous variable t_e, and ensure that if activity i starts immediately after event e and ends at event f, then the date of event f is at least equal to the date of event e plus the processing time of activity i. Constraints (28) and (29) are called contiguity constraints, and they ensure non pre-emption (i.e. that the events after which a given activity is being processed are adjacent). Constraints (30) maintain the precedence relationship. Constraints (31) are the resource constraints limiting the total demand of activities in process at each event. Constraints (32) and (33) ensure that the start time of any activity is between its earliest and latest start time.

4 Experimental Study

The mathematical models developed in the previous section for MRCPSPs are integer programming models which can be solved using standard optimization algorithms. The developed models were coded with the LINGO optimization software, and were executed on an Intel core i7 processor with 16.00 GB RAM and a 3.40 GHz CPU. For solving the models, the Branch and Bound algorithm (B&B) algorithm within LINGO was applied. To compare the models, we have selected a number of MRCPSP benchmark instances from the popular test library known as PSPLIB-Project Scheduling Problem Library [15]. The characteristics of the MRCPSP instances vary a lot in terms of the average number of resources required (resource factor, RF), the resource feature incorporates time parameters (resource strength, RS) and finally the integration of resource and precedence features. For justifying the proposed formulations, we considered the RS and RF for both renewable and non renewable resource to be 0.5, which means that if the project has 4 renewable or non renewable resources then for any particular time each activity can demand two resources.

As suggested by the earlier models, the conventional DT approach needs mnT binary variables, where m is the number of modes, n is the number of activities including the dummy ones and T is the total project completion time. But for the SEE and OOE approaches, they need only $2mn^2$ and mn^2 binary variables to represent any MRCPSP. This indicates the binary variables required for SEE are twice that of OOE. To compare the above mentioned three models, we have considered a number of 10-activity test instances, with low activity duration time, that have an average makespan of 18. As shown in Table 1, the conventional DT approach needs on average 648 binary variables, while OOE and SEE require 432 and 864 respectively. After multiplying each activity duration time by 10 (now considered as a higher activity duration), the number of binary variables for DT becomes 6156 and the makespan increases from 18 to 81. However, the number of variables for SEE and OOE remain the same. As the number of variables is the dominating factor for the computational time required in solving a model, the time recorded in Table 1 consistently shows that the models with higher numbers of variables takes longer computational time than the same ones with lower numbers of variables.

Table 1. Binary variable and constraint based comparison among all mentioned approaches

10-activity Instances	Parameters	DT	SEE	OOE
With low activity-duration time	No. of Variables	648	864	432
	Solution time (sec)	2.0	3.0	1.5
With high activity-duration time	No. of Variables	6156	864	432
	Solution time (sec)	5.0	3.0	1.5

We have further analyzed and compared the above mentioned three models using a number of test instances from PSLIB with 10, 12, 14, 16, 18, 20 and 30 activities for both high and low duration times. Here, high duration means five times longer than low duration. For the low activity duration, the number of variables and the computational time required, with respect to the number of activities, are presented in Figures 1 and 2 respectively. For the high activity duration, these plots are shown in Figures 3 and 4. In these figures, for either low or high activity duration of all three models, the number of variables increases with the number of activities and the computational time consistently increases with the increase of the number of variables. There is no impact on the number of variables in SEE and OOE when the activity duration is changed (say changing from low to high). However, as can be seen in Table 1 and

Figures 1 and 3, the number of variables significantly increases in DT as activity duration increases. From these four figures, it is clear that OOE is the best model irrespective of the length of activity duration. However, SEE is better than DT only for high activity duration.

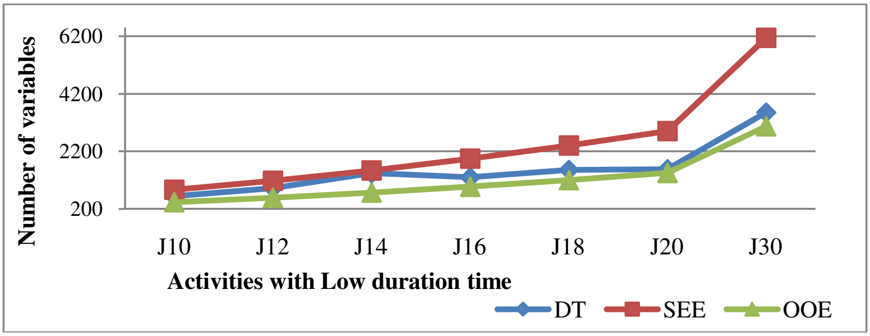

Fig. 1. Relationship between number of variables and activities with low duration time

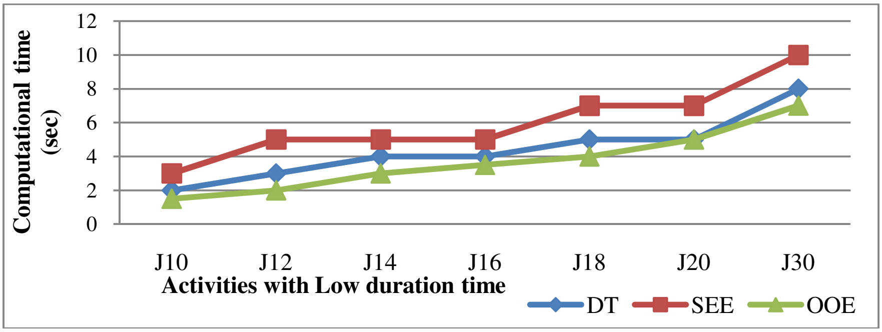

Fig. 2. Relationship between computing times and activities with low duration time

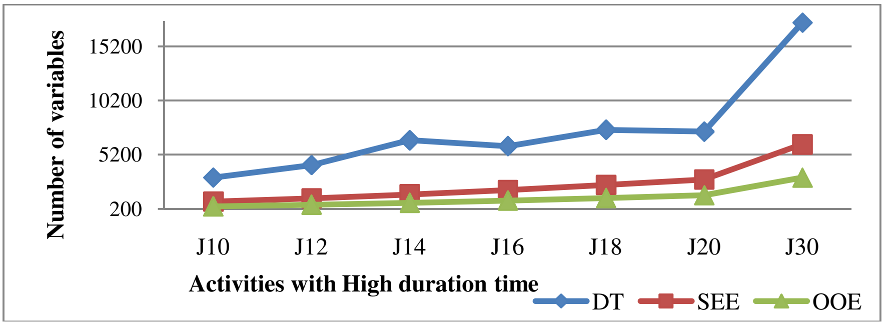

Fig. 3. Relationship between number of variables and activities with high duration time

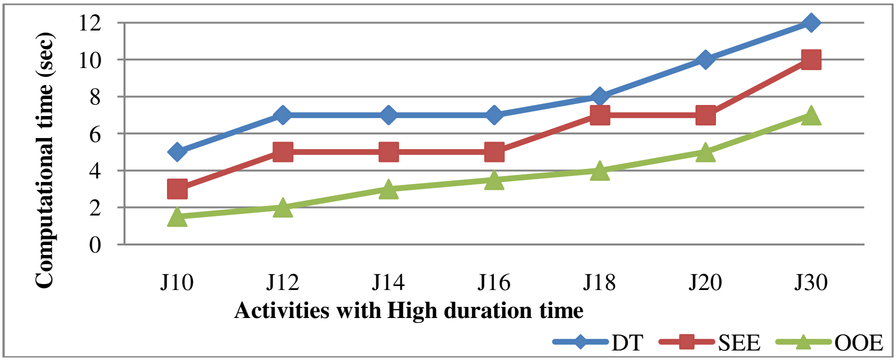

Fig. 4. Relationship between computing times and activities with high duration time

4.1 Effects of New Activity Insertion

In real life projects, some activities may be added or removed during project execution. In this section, we will analyze the impact of new activity insertion. To do this, we assumed that a set of new activities A_N with precedence relations P_N need to be added to the project network. Here we have considered three cases for a 10-activity instance (J1010_4) with randomly generated duration and resource usage requirements on each mode. In figure 5, Ins-2 represents the insertion of two new activities which replaced 2 old activities. From the figure, it is quite clear that the impact of new activity insertion is insignificant for event based approaches. However it has a clear impact on DT model. Note that, for all approaches, the number of constraints increases significantly with the increase of new activities.

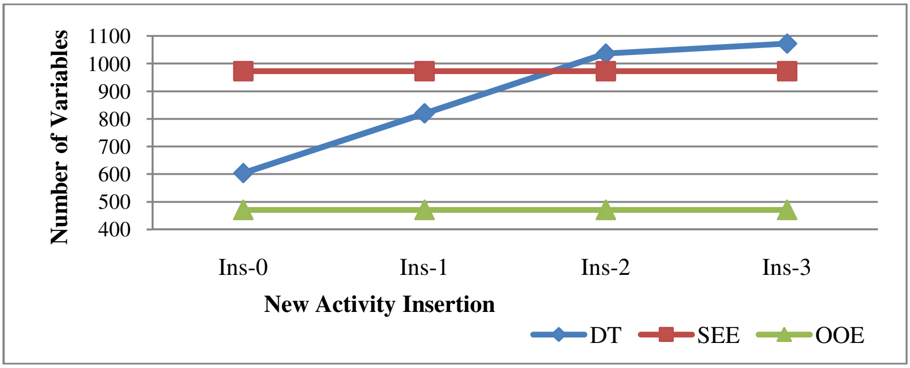

Fig. 5. Distribution of number of variables with new activity insertion

4.2 Effects of Precedence Modifications

In a project, the activity priority may be revised and this may change the precedence relationship. In this section, we have also analyzed the impacts of any particular precedence modifications. An example of precedence modification is when the project network for A needs to satisfy the additional precedence relationship P_A, and that it also no longer needs to satisfy the relations in P_R. We have considered four randomly generated precedence modifications for analyzing the impact on the three models. As shown in figure 6, similar conclusion can be drawn for precedence modifications as was earlier found for new activity insertions. From the figure, it can be observed that with increasing numbers of precedence modifications (PM), the number of variables remain same for both event based approaches, while for the DT approach, the number of variables is increased. So, it can be claimed, that the proposed event based approaches are independent of precedence modifications. But for all approaches, the number of constraints is slightly increased with the increase of precedence modifications.

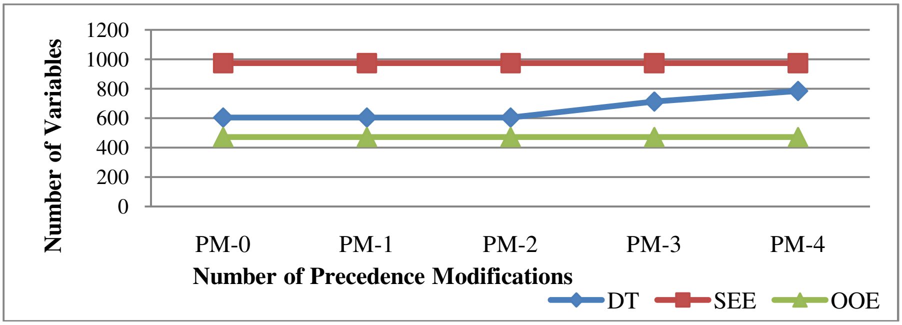

Fig. 6. Distribution of number of variables with precedence modifications

5 Conclusions

In this paper, we have developed two new event based mathematical models (mixed integer linear program) for MRCPSP that reduced the number of variables and hence the required solution times. For solving the models, the branch and bound techniques were implemented within LINGO. A number of benchmark problems with different numbers of activities and resources were solved and compared. From the comparisons, it can be concluded that (i) there is no impact on the number of variables in SEE and OOE with activity duration, (ii) the number of variables significantly increases in DT with increases of activity duration, (iii) OOE is the best model irrespective of the length of activity duration, (iv) SEE is better than DT only for high activity durations, (v) the effect of new activity insertion or precedence modification on the number of variables are insignificant for SEE and OOE, and (vi) there is a clear negative effect for new activity insertion or precedence modification on the DT model. So practitioners will benefit by choosing the proposed event based model (OOE) for MRCPSP, as it requires the lower number of variables and lower computational time in comparison to other existing models. The future research should be on proposing some real time event based RCPSP approaches for handling uncertainties or research disruptions. Implementing this model for more complex problems with large number of activities could be another important extension of these proposed models.

References

1. Hartmann, S., Briskorn, D.: A survey of variants and extensions of the resource-constrained project scheduling problem. European Journal of Operational Research 207(1), 1–14 (2010)
2. Zapata, J.C., Hodge, B.M., Reklaitis, G.V.: The multimode resource constrained multiproject scheduling problem: Alternative formulations. AIChE Journal 54(8), 2101–2119 (2008)
3. Koné, O., Artigues, C., Lopez, P., Mongeau, M.: Comparison of mixed integer linear programming models for the resource-constrained project scheduling problem with consumption and production of resources. Flexible Services and Manufacturing Journal 25(1-2), 25–47 (2013)
4. Pritsker, A.A.B., Waiters, L.J., Wolfe, P.M.: Multiproject scheduling with limited resources: A zero-one programming approach. Management Science 16(1), 93–108 (1969)
5. Klein, R., Scholl, A.: Computing lower bounds by destructive improvement: An application to resource-constrained project scheduling. European Journal of Operational Research 112(2), 322–346 (1999)
6. Lawler, E.L.: Combinatorial optimization: networks and matroids. Courier Dover Publications (1976)
7. Sabzehparvar, M., Seyed-Hosseini, S.M.: A mathematical model for the multi-mode resource-constrained project scheduling problem with mode dependent time lags. The Journal of Supercomputing 44(3), 257–273 (2007)
8. Herroelen, W., Leus, R.: Project scheduling under uncertainty: Survey and research potentials. European Journal of Operational Research 165(2), 289–306 (2005)
9. Alcaraz, J., Maroto, C., Ruiz, R.: Solving the Multi-Mode Resource-Constrained Project Scheduling Problem with genetic algorithms. Journal of the Operational Research Society 54(6), 614–626 (2003)
10. Koné, O., Artigues, C., Lopez, P., Mongeau, M.: Event-based MILP models for resource-constrained project scheduling problems. Computers & Operations Research 38(1), 3–13 (2011)
11. Pinto, J., Grossmann, I.: A continuous time MILP model for short term scheduling of batch plants with pre-ordering constraints. Computers & Chemical Engineering, 20, S1197–S1202 (1996)
12. Lenstra Jr, H.W.: Integer programming with a fixed number of variables. Mathematics of Operations Research 8(4), 538–548 (1983)
13. Beşikçi, U., Bilge, Ü., Ulusoy, G.: Different resource management policies in multi-mode resource constrained multi-project scheduling. In: 2011 World Congress on Engineering and Technology, Shanghai, China (2011)
14. Talbot, F.B.: Resource-constrained project scheduling with time-resource tradeoffs: The nonpreemptive case. Management Science 28(10), 1197–1210 (1982)
15. Kolisch, R., Sprecher, A., Drexl, A.: Characterization and generation of a general class of resource-constrained project scheduling problems. Management Science 41(10), 1693–1703 (1995)