

# A Bi-objective Pre-emption Multi-mode Resource Constrained Project Scheduling Problem with due Dates in the Activities

Zahra Sadat Hosseini<sup>a,\*</sup>, Javad Hassan Pour<sup>b</sup>, Emad Roghanian<sup>c</sup>

<sup>a</sup> MSc, Department of industrial Engineering, K.N. Toosi University of Technology, Tehran, Iran

<sup>b</sup> MSc, Department of industrial Engineering, Mazandaran University of Science and Technology, Mazandaran, Iran

<sup>c</sup> Assistant professor, Department of industrial Engineering, K.N. Toosi University of Technology, Tehran, Iran

Received 5 August, 2013; Revised 22 February, 2014; Accepted 19 April, 2014

---

## Abstract

In this paper, a novel mathematical model for a preemption multi-mode multi-objective resource-constrained project scheduling problem with distinct due dates and positive and negative cash flows is presented. Although optimization of bi-objective problems with due dates is an essential feature of real projects, little effort has been made in studying the P-MMRCPSP while due dates are included in the activities. This paper tries to bridge this gap by studying tardiness MMRCPSP, in which the objective is to minimize total weighted tardiness and to maximize the net present value (NPV). In order to solve the given problem, we introduced a Non-dominated Ranking Genetic Algorithm (NRGA) and Non-Dominated Sort Genetic Algorithm (NSGA-II). Since the effectiveness of most meta-heuristic algorithms significantly depends on choosing the proper parameters. A Taguchi experimental design method was applied to set and estimate the proper values of GAs parameters for improving their performances. To prove the efficiency of our proposed meta-heuristic algorithms, a number of test problems taken from the project scheduling problem library (PSPLIB) were solved. The computational results show that the proposed NSGA-II outperforms the NRGA.

*Keywords:* Multi-objective Project Scheduling, Resource Constraint, Preemption, Net Present Value, Meta-heuristic Algorithm.

---

## 1. Introduction

The resource-constrained project scheduling problem (RCPSP) aims at scheduling project activities in order to complete the project in the minimum possible time with respect to precedence and resource constraints. The precedence of constraints is defined in such a way that no activity can be started before finishing all of its predecessors. [Sonke Hartmann (2009)]

The RCPSP, along with some of its extensions, has been widely studied in the literature. The multi-mode RCPSP, shown by MRCPSP, is a generalized version of the RCPSP where each activity can be performed in one out of a set of modes with a specific duration and resource requirements. The objective of the MRCPSP is to find a mode and a start time for each activity such that the makespan is minimized and the schedule is feasible with respect to the precedence and the resource constraints. First method for solving the multi-mode problems was presented by Slowinski (1980)

proposing one stage and two stages linear programming. Talbot (1982) proposed an approach based on a numeric programming. Whilst Speranza and Vercellis (1993) applied a branch-and-bound algorithm to address the problem, Hartmann and Sprecher (1996) revealed that the algorithm was not able to solve the problem with one and two renewable constraints. After that, several other studies were conducted using branch-and-bound algorithm [see Sprecher et al. (1997), Hartmann and Drexel (1998), and Sprecher and Drexel (1998)]. More recently, Zhu et al. (2006) dealt with the problem by proposing a branch-and-cut algorithm. Ozdamar (1999), Brucker (1999), Jozefowska et al., Hartmann (2001), Alcaraz et al. (2003), Bouleimen and Lecocq (2003), Varma et al. (2007), and Jarboui et al. (2008) discuss a multi-mode problem without non-renewable resources. Multi-mode problems with generalized precedence constraints have been considered by Reyck and Herroelen (1999), Reyck and Herroelen (1999),

---

\* Corresponding author E-mail: zahra.hosseini46@yahoo.com

Drexl et al. (2000), Heilmann (2001-2003), Nonobe and Ibaraki (2002), Calhoun et al. (2002), and Brucker and Knust (2001). Barrios et al. (2011), Zhu et al. (2006), and Sabzehparvar and SeyedHosseini (2008) applied a multi-mode problem with generalized resource constraints.

In traditional PRCPSP and PMRCPSP activities are allowed to be pre-empted at any integer instant time and restarted later on at no additional cost. Practically, such an assumption is not often true. One of the extensions of the RCPSP and MRCPSP considers pre-emption which means the activities cannot be interrupted once they are started. Despite the widespread application of such an assumption, this problem has not received enough attention by researchers.

Kaplan (1988, 1991) was the pioneer scholar who studied the pre-emptive resource-constrained project scheduling problem (PRCPSP). Demeulemeester and Herroelen (1996), Nudtasomboon and Randhawa (1997), Bianco et al. (1999), Brucker and Knust (2001), and Debels and Vanhoucke (2007) allowed activity pre-emption at discrete points of time. That is an activity can be interrupted after each integer unit of its processing time.

For multi-mode cases, Buddhakulsomsiri and Kim (2006) considered the MRCPSP where all resources were renewable and the activities may be pre-empted due to scarcity of resources during the project. They have proved that pre-emption is very effective to improve the optimality of project makespan. Although some studies concerning pre-emption have been conducted on the pre-emptive resource-constrained project scheduling problem (P-RCPSP) and pre-emptive multi-mode resource-constrained project scheduling problem (P-MRCPSP), none of them could be used to solve large scale problems and provide the optimal solution within a reasonable period of time. Therefore, the heuristic and meta-heuristic algorithms have been introduced to solve such problems.

In order to address the MRCPSP by heuristic and meta-heuristic algorithms, Debels et al. (2006) utilized a genetic algorithm. Lova et al. (2006) presented a multi-stage priority-based heuristic algorithm. Zhang et al. (2006) and Jarboui et al. (2008) proposed a methodology based on particle swarm optimization algorithm. Ranjbar et al. (2008) used a hybrid scatter search algorithm using the path relinking methodology as the solution combination method. Coelho and Vanhoucke (2011) used a new meta-heuristic algorithm called a satisfiability (SAT) algorithm for solving multi-mode project scheduling with constraint on renewable and non-renewable resources problems. A recent study by Ballestin and Blanco (2011) showed that few research efforts have been concentrated on the multi-objective resource-constrained project scheduling problem. The authors tackled the shortcoming by applying Non-dominated Storing Genetic Algorithm (NSGAI), Strength Pareto Evolutionary Algorithm (SPEAI), and Pareto

Simulated Annealing (PSA). Deblaere et al. (2010) presented a multi-mode resource-constrained project scheduling problem (MRCPSP) with the objective of minimizing project makespan. Voss and Witt (2007) formulated the MRCPSP with an objective function that contained makespan, weighted tardiness, and setup costs. Goto et al. (2001) proposed a meta-heuristic algorithm containing a two-step tabu search to maximize the NPV. Mika et al. (2005) have deployed simulated annealing and tabu search algorithms in activity-on-node (AON) networks with the objective of maximizing the net present value.

In this paper, we address a multi-objective pre-emption multi-mode resource-constrained project scheduling problem with discounted cash flow (MMPRCPSPDCF). For the cash flow, both positive and negative values are considered. With respect to activities' constraints, we have assumed both renewable and non-renewable resources. The objective function involves NPV maximization and total weighted tardiness minimization. Given that the problem is NP-hard [Ozdamr 1998], a novel meta-heuristic algorithm called the multi-objective imperialist competitive algorithm (MOICA) is applied.

This paper contributes to the previous literature in three ways. First, since considering preemption in multi-objective multi-mode project scheduling problems brings a high level of complexity, it has hardly been assumed in modeling previous studies, while we have fulfilled this task. Second, in our model, the activities are scheduled in such a way that they are completed within their earliest and latest finish time while preemption has been taken into account. This is the first paper of its kind that models the project scheduling problem with respect to preemption. Third, this study considers due dates for each activity and the objective function is NPV maximization and tardiness minimization, which is quite novel in multi-objective project scheduling problems. Our survey of literature shows that no other study, to date, has addressed the problem with the scope this paper.

The rest of this paper is as follows. First, we describe the problem and its mathematical model in Section 2. Section 3 proposes the NSGA-II algorithm to solve some definite test problems. In Section 4, the experimental results obtained from the proposed NSGA-II are presented and compared with a multi-objective evolutionary algorithm, called Non-dominated Ranking Genetic Algorithm (NRGA). Finally, Section 5 concludes the paper and provides recommendations for future research.

## 2. Problem Description

The model is defined as the AON network  $G = (V, E)$ , where the nodes represent the activities and the arcs denote

the precedence relationship. Arc (i, j) indicates that activity j can be started if and only if activity i has been completed. The set of activities are defined by  $V = \{1, \dots, N\}$  in which activities 1 and N are dummy. Dummy activity 0 represents the first activity and dummy activity n+1 represents the last activity of the project. Each activity j is performed in a mode shown by  $m_j$ . Each activity-mode combination has a fixed duration ( $d_{jm}$ ) and requires a constant amount of one or more of K types of renewable resource ( $r_{jkm}$ ) and L types of non-renewable resources ( $r_{jlm}$ ) for the execution of the activity. The objectives of the PMMRCPS are maximizing project NPV and minimizing the total weighted tardiness.

The PMMRCPS considered adheres to the following assumptions:

- Kind of precedence relationships are finish-start with a minimal time lag of 0 (i.e. an activity can be started only if all of its predecessors completed and it does not need setup time.)
- Activities are allowed to be preempted at any time.
- Activities are performed between their earliest start time and latest finish time.
- Resources used in projects are limited and comprised of both renewable and non-renewable resources.
- Each activity has a pre-specified duration.
- All costs of an activity are paid prior to its initiation. The preemptions do not affect the costs.
- The resulting revenue from accomplishing all the activities is achieved at the end of the project.
- The budget is not limited.
- The project has a distinct due date.
- If an activity exceeds its pre-specified time, it is supposed to be tardy.

### 2.1. Mathematical formulation

#### Index

N: Number of activities ( $i, j = 1, 2, \dots, N$ )

T: Number of periods ( $t = 1, 2, \dots, T$ )

K: Renewable resources ( $k = 1, 2, \dots, K$ )

L: Non-renewable non-resources ( $k = 1, 2, \dots, L$ )

#### Parameters

$M_j$ : Modes of activity j

$d_{jm}$ : Duration of activity j with respect to mode m

$R_k$ : Total available units of renewable resources k

$R_l$ : Total available units of non-renewable resources l

$r_{kmj}$ : Number of renewable resources k used by activity j in mode m

$r_{lmj}$ : Number of non-renewable resources l used by activity j in mode m

M: A big positive number

$w_j$ : Weight of activity j representing the importance of activity j

$[r_j, d_j]$ : Time window of activity j where  $r_j$  and  $d_j$  are respectively the release time and due date of activity j.

$C_j$ : Completion time of activity j

T: Maximum project horizon time

$\alpha_i$ : Decreasing rate per unit of time

$CF_{jm}^-$ : Negative cash flow assigned to mode m of activity j

$CF_j^+$ : Positive cash flow assigned to activity j

$a_{ji} = \begin{cases} 1 & \text{If activity j is the predecessor of activity i} \\ 0 & \text{Otherwise} \end{cases}$

$b_{ij} = \begin{cases} 1 & \text{If activity i is the successor of activity j} \\ 0 & \text{Otherwise} \end{cases}$

#### Variables

$ES_j$ : The earliest start time of activity j

$EF_j$ : The earliest finish time of activity j

$LF_j$ : The latest finish time of activity j

$LS_j$ : The latest start time of activity j

$X_{jmt} = \begin{cases} 1 & \text{If activity j is executed in mode m is performed at time t} \\ 0 & \text{Otherwise} \end{cases}$

$y_{jm} = \begin{cases} 1 & \text{If activity j is executed in mode m} \\ 0 & \text{Otherwise} \end{cases}$

The PMMRCPS can be represented as follows:

$$\max Z_1 \tag{1}$$

$$= \sum_{j=1}^{n+1} \frac{CF_j^+}{(1 + \alpha_i)^{EF_j}} - \sum_{j=1}^{n+1} \sum_{m=1}^{M_j} \sum_{t=ES_j}^{LF_j} \frac{CF_{jm}^-}{(1 + \alpha_i)^t} X_{jmt} \tag{2}$$

$$\min Z = \sum_{j=1}^n w_j \cdot \max(0, C_j - d_j) \tag{3}$$

$$St: \tag{4}$$

$$ES_1 = 0 \tag{5}$$

$$ES_j \geq r_j, \quad \forall j \tag{6}$$

$$EF_j = ES_j + \sum_{m=1}^{M_j} d_{jm} y_{jm}, \quad \forall j \tag{7}$$

$$ES_j = \max\{EF_i a_{ji}\}, \quad \forall j, i \neq j \tag{8}$$

$$LS_j = LF_j - \sum_{m=1}^{M_j} d_{jm} y_{jm}, \quad \forall j \tag{9}$$

$$LF_j = \min\{LS_j b_{ij}\}, \quad \forall j, i \neq j, j \neq 1, n \tag{10}$$

$$B_{ij} = \begin{cases} 1 & \text{if } b_{ij} = 1 \\ M & \text{otherwise} \end{cases} \tag{11}$$

$$LS_n = EF_n \tag{12}$$

$$ES_j y_{jm} < t \cdot X_{jmt} + (1 - X_{jmt})M, \quad \forall j, j \neq 1, n, \forall m, t \tag{13}$$

$$t \cdot X_{jmt} \leq LF_j \tag{14}$$

$$\sum_{t=ES_j}^{LF_j} X_{jmt} = d_{jm} y_{jm}, \quad \forall j, m \tag{15}$$

$$\sum_{m=1}^{M_j} y_{jm} = 1, \quad \forall j \quad (14)$$

$$\sum_{m=1}^{M_j} r_{kmj} X_{jmt} \leq R_k \quad \forall k, t, j \quad (15)$$

$$\sum_{j=1}^n \sum_{t=ES_j}^{LF_j} \sum_{m=1}^{M_j} r_{lmj} X_{jmt} \leq R_l \quad \forall l \quad (16)$$

$$C_j \geq t \cdot \sum_{m=1}^M X_{jmt} \quad \forall j, t \quad (17)$$

$$FF_j = \max_t \left\{ t \cdot \sum_{m=1}^{M_j} X_{jmt} \right\}, \quad \forall j \quad (18)$$

$$SS_j = \min_t \{ t \cdot X_{jmt} + (1 - X_{jmt})M \} \quad \forall j, m \quad (19)$$

$$SS_j > FF_i \cdot a_{ji}, \quad \forall j \text{ \& } i \neq j \quad (20)$$

The first objective function (1) is to maximize NPV of the project and the second objective function (2) is to minimize the total weighted tardiness of the project. Constraint (3) forces the earliest start time of project to be equal to 0. Constraint (4) ensures that the earliest start time for activity  $j$  cannot be smaller than the pre-specified start time allowed to activity  $j$ . Constraint (5) represents the earliest finish time of activity with respect to preemption and precedence relations. Constraint (6) represents the earliest start time of activity with respect to preemption and precedence relations. Constraint sets (7), (8), and (9) represent the latest start and finish time of activity according to the preemption and precedence relations. Constraint (10) guarantees that the latest start time of muddy activity  $N$  is equal to the earliest finish time of muddy activity  $N$ . Constraint (11) and (12) ensure that each activity is executed within its earliest start time and latest finish time, respectively. Constraint (13) is summation of periods at which activity  $j$  is executed in any

of its  $m$  modes. Constraint (14) imposes that each activity is performed only in one mode. Constraint (15) and (16) consider the renewable and non-renewable resources limitations, respectively. Constraint (17) states that finish time of each activity cannot be smaller than the summation of periods at which mode  $m$  of activity  $j$  is executed. Constraint (18), (19) and (20) represent the start and finish time of each activity with respect to preemption.

### 3. The Proposed Algorithms

#### 3.1. Non-dominated sorting genetic algorithm II (NSGA-II)

The use of evolutionary algorithms for multi-objective optimization has significantly grown in the last years. NSGA-II is one of the most well-known and efficient algorithms of this kind introduced by Deb et al. (2000, 2002). This algorithm is a revised version of non-dominated sorting genetic algorithm (NSGA) proposed by Srinivas and Deb (1995). The general procedure of the algorithm is as follows. Firstly, a random parent solution is created. Then, the algorithm uses crossover operator and mutation operator to generate new solutions as generated offspring of size  $N$ . After that, the current population and generated offspring are combined together with size of  $2N$ . Finally, the best solutions in terms of non-dominance and crowding distance are selected from combined population as the new population. Similarly, ranking and selecting the population fronts are performed by non-dominance technique and a crowding distance. The non-dominated technique, the calculation of crowding distance, and crowding selection operator are explained below. The procedure of NSGAI algorithm is illustrated in the Figure 1.

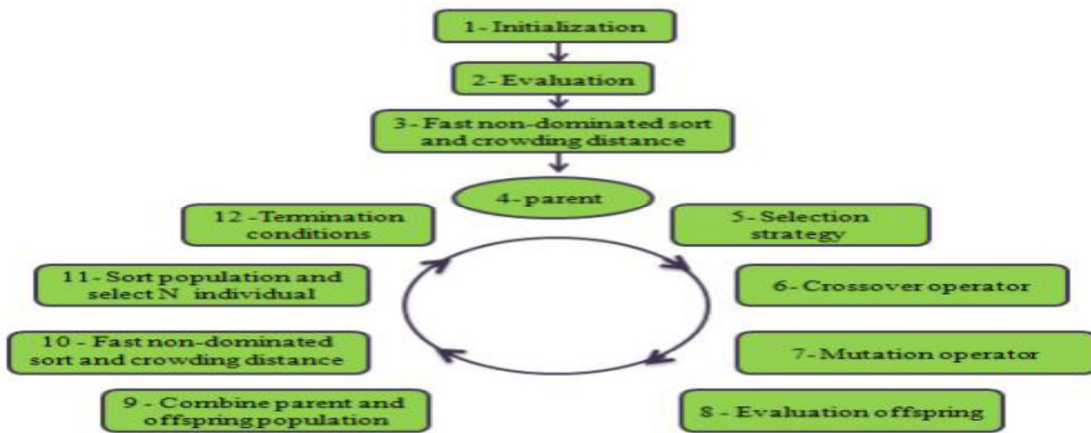


Fig. 1. Flowchart of the NSGAI algorithm

### 3.1.1. Non-dominance technique

Suppose that there are  $r$  objective functions. When the following conditions are satisfied, the solution  $x_1$  dominates another solution, let us say  $x_2$ . If  $x_1$  and  $x_2$  do not dominate each other, they are placed in the same front.

(1) For all the objective functions, solution  $x_1$  is not worse than solution  $x_2$ .

(2) For at least one of the  $r$  objective functions  $x_1$  is exactly better than  $x_2$ .

Front number 1 is made by all the solutions that are not dominated by any other solutions while front number 2 is built only by the solutions that are dominated by solutions in front number 1.

### 3.1.2. Crowding distance

Crowding distance is a measure of how close an individual is to its neighbors. Larger value for crowding distance leads to better diversity in the surrounding of a particular solution. The crowding distance that is used in the proposed NSGA-II is shown in equation (21). The solutions with lower values of crowding distance are preferred to solutions with higher values of crowding distance.

Where  $r$  is the number of objective functions,  $f_{k,i+1}^p$  is the  $k$ -th objective function of the  $(i+1)$ -th solution and  $f_{k,i-1}^p$  is the  $k$ -th objective function of the  $(i-1)$ -th solution after sorting the population according to crowding distance of the  $k$ -th objective function. Also  $f_{k,total}^{p,max}$  and  $f_{k,total}^{p,min}$  are the maximum and minimum values of objective function  $k$ , respectively.

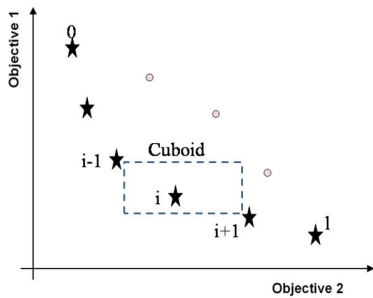


Fig. 2. Calculate the crowding distance

### 3.1.3. Tournament selection operator

A binary tournament selection procedure has been applied for selecting solutions for both of the crossover and mutation operators. This procedure works as follows. First, two solutions of the population size are selected. Then, the lowest front number is selected if two populations are from different fronts. In case the populations are from the same front, the solution with the highest crowding distance is

selected. After describing our proposed method, we will explain the en-coding procedure of each algorithm.

### 3.2. Non-dominated Ranking Genetic Algorithm (NRGA)

NRGA is structurally similar to controlled NSGA-II. They only differ in strategy selection, population ranking, and the choice of the next generation. In NRGA, the roulette wheel operator based ranking (RRWS) is used instead of crowded tournament operator. This operator is designed in a way that members have higher quality by increasing the probability of being selected for the next generation and reproduction.

Each number of the population has two attributions located on non-dominated boundary and its rank based on the distance swarm. Therefore, in order to choose an answer it must first select a non-dominated boundary and then pick an answer within its boundary. For each  $i$ , equation 22 calculates the probability of selection as a non-dominated boundary.

Where  $rank_i$  is the rank for each  $i$ ,  $N_f$  is the number of the specified boundaries in the ranking of non-dominated boundaries at each stage. Obviously, it is more likely for an answer to be chosen if the answer is situated on better boundaries. The probability of answer selection for each  $j$  in the non-dominated boundary for each  $i$  is calculated as follows.

Where  $N_j$  represents the number of boundary solutions for each  $i$  and  $rank_{ji}$  denotes the rank of answer for each  $j$  on the boundary with respect to  $i$  based on the distance swarm. According to equation 23, the greater the distance swarm, the more probability exists to select the corresponding answer.

## 4. Comparison Metrics

In this section, we express quantitative and qualitative comparisons metrics often used for comparing meta-heuristics algorithms.

### 4.1. Mean Ideal Distance Metric (MID)

By this comparison metric, the closest distance between Pareto solutions to the ideal point  $(f_1^{best}, f_2^{best})$  is calculated according to equation 24.

Where  $f_1^{best}$  and  $f_2^{best}$  are the ideal points of objective functions and  $n$  is the number of Pareto solution and  $f_{1i}$  and  $f_{2i}$  are the first and second objective function values for each  $i$ , respectively. Generally, an algorithm with a lower value of MID has a better performance.

4.2. The Rate of Achievement to Simultaneous Objectives (RAS)

At the beginning, the ideal point of the objective function is obtained. Then RAS is calculated according to equation 25. Where  $f_{i,total}^{max}$  and  $f_{i,total}^{min}$  are the maximum and minimum values of each fitness functions among all of non-dominated solutions obtained by the algorithms, respectively. The algorithm with a lower value of RAS has a better performance.

4.3. Spacing Metric (SM)

Through this measure, uniform distribution of non-dominated solutions is obtained according to the equation 26. Where  $d_i$  is the Euclidean distance between consecutive solutions in the obtained non-dominated set of solutions, while  $\bar{d}$  is the average of these distances. The algorithm with a lower value of SM has a better performance.

5. Experimental Results

Since the proposed mathematical model is quite novel, no suitable problems were found in the existing literature for testing the performance of algorithms. In this study, totally 10 scheduling programs has been selected including five small size and five large size programs from Kiel University Library website [www.om-db.wi.tum.de/psplib/datamm.html]. These programs provide the specifications of activities and their prerequisite relationships. In addition, some other data were necessary to be inserted into the model according to its requirements, which are described as below.

- The numbers of modes in each operating activity (in both of cases, small and medium size problems) is equal to three.
- Types of resources needed in small and big size problems are equal to two.
- Normal time for each activity in any mode follows the uniform distribution of U(1, 10).
- Positive and negative cash flows of each activity follow a random uniform distribution.
- Weighted tardiness of each activity follows the uniform distribution of U(0, 1).
- The due date and release time of each activity follow from a uniform distribution of U (0, 20).

5.1. Taguchi parameter design

An appropriate design of input parameters has a significant impact on the efficiency and effectiveness of the algorithm. In this section, we study the behavior of different

parameters of the proposed NREGA and NSGA-II. In order to calibrate the algorithms, many techniques have been proposed to statistically design an experimental investigation. Amongst such techniques, Taguchi's parameter design is well known for its robust optimization. It has been applied widely in the field of engineering to select proper parameter levels for engineering design or manufacturing processes. The factors in Taguchi are divided into two main categories: controllable and noise factors. The controllable factors are the factors that can be monitored and are expected to select their proper levels so as to achieve higher performance. The noise factors are uncertain factors that cannot be well controlled. The Taguchi method seeks to minimize the effects of noise factors and to determine the optimal level of important controllable factors based on the concept of robustness. In addition to determining the optimal levels, Taguchi establishes the relative significance of individual factors in terms of their main effects on the objective function.

Taguchi created a transformation of the repetition data to another value which is an indicator of variation. The transformation is the signal-to-noise (S/N) ratio, which explains why this type of parameter design is called a robust design. The aim of Taguchi design is to maximize the signal-to-noise ratio.

Taguchi classifies objective functions into three distinct categories: the smaller-the-better type, the larger-the-better type, and the nominal-is-best type. Most of the objective functions in scheduling problems fall within the smaller-the-better type, while in this paper all objective functions in resource constrained project scheduling are classified in the nominal-is-best type. The corresponding S/N ratio is calculated according to equation 27.

In the Taguchi method, orthogonal arrays are used to study a large number of decision variables with a small number of experiments. To select the appropriate orthogonal array, ascertaining the degree of freedom is necessary. The associated degree of freedom for these four factors is equal to:  $1 + (2 \times 4) = 9$ . From the standard table of orthogonal arrays, L9 is the appropriate array and satisfies these conditions. Therefore, the appropriate array should have nine rows. The orthogonal array L9 is presented in Table 1.

Table 1  
The orthogonal array L9

Experiment number	A	B	C	D
1	1	1	1	1
2	1	2	2	2
3	1	3	3	3
4	2	1	2	3
5	2	2	3	1
6	2	3	1	2
7	3	1	3	2
8	3	2	1	3
9	3	3	2	1

In the NSGA-II and NPGA algorithm  $n_{pop}$ ,  $p_c$ ,  $p_m$ ,  $Max\ gen$  are the four control factors for our proposed algorithms. For each control factor, three levels are considered. Table 2 shows the controllable factors and their respective levels for each problem. Moreover, each factor is displayed by one symbol as shown in Table 2 and 3.

Table 2  
Parameters and their levels for NSGA-II Problems

Factor	Symbol	Level 1	Level 2	Level 3
$n_{pop}$	A	100	150	200
$Max\ gen$	B	100	75	50
$p_c$	C	0.95	0.85	0.8
$p_m$	D	0.05	0.1	0.2

Table 3  
Parameters and their levels for NPGA Problems

Factor	Symbol	Level 1	Level 2	Level 3
$n_{pop}$	A	100	150	200
$Max\ gen$	B	100	75	50
$p_c$	C	0.95	0.85	0.8
$p_m$	D	0.05	0.1	0.2

In this subsection, the performance of NSGAII and NPGA algorithm are compared to solve the model. These algorithms have been programmed with *MATLAB 2011b* and run on a personal computer with a 2.53 GHz CPU and 6 GB main memory.

As already mentioned, the test problems were comprised of small and large size problems as in Table 3. In this paper, the Taguchi method is applied to both scales for parameter tuning. Parameter tuning by the Taguchi method is explained in detail by representing the step by step results for large size problems, while the obtained results from small size problems are also reported. In each scale, five test problems are chosen randomly. To yield more reliable results, each problem is solved five times. Hence, there are 25 results for each trial. The best result among the five runs of each problem is shown as the result of that problem in Table 4.

Table 4  
Description of the test problem

Problem number	Project file name at (PSPLIB)	Number of activities	Size of problem	Number of resources	Number of modes
1	j1227-8.mm	∩∩	Small	3	2
2	j1227-9.mm	∩∩	Small	3	2
3	j1227-10.mm	∩∩	Small	3	2
4	j1228-1.mm	∩∩	Small	3	2
5	j1228-2.mm	∩∩	Small	3	2
6	j189-1.mm	∩∩	Large	3	2
7	j189_2.mm	∩∩	Large	3	2

8	j189_3.mm	∩∩	Large	3	2
9	j189_4.mm	∩∩	Large	3	2
10	j189_5.mm	∩∩	Large	3	2

The optimal level of each factor in NSGA-II is illustrated in figure 3. Clearly, the most suitable levels of the aforementioned factors are A(2), B(2), C(2) and D(3) respectively.

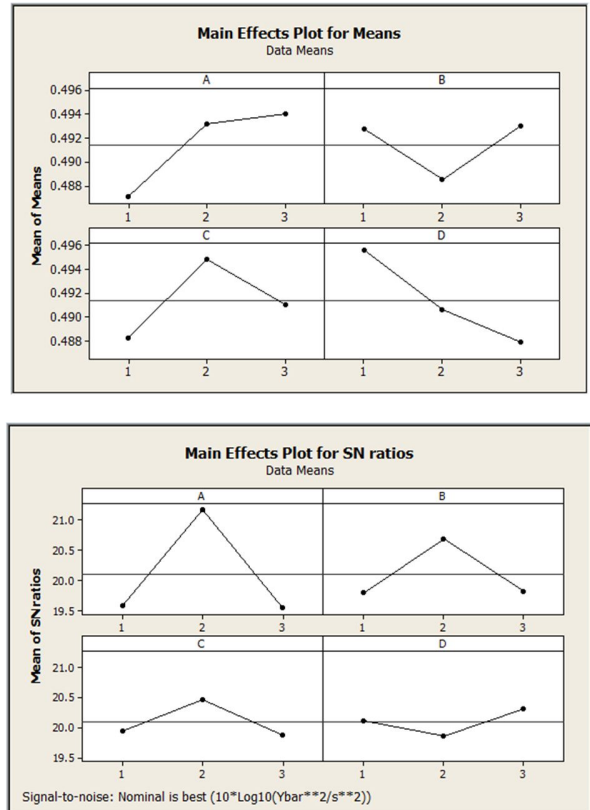


Fig. 3. Mean S/N ratio for level of each factor NSGA-II

Table 5  
S/N NSGAII

Response table for signal to noise ratios (nominal is better)				
level	A	B	C	D
1	19.58	19.80	19.94	20.12
2	21.17	20.68	20.46	19.87
3	19.54	19.81	19.89	20.31
Delta	1.63	0.88	0.57	0.45
Rank	1	2	3	4

From figure 3 and the value of data in Table 5, we can conclude that  $n_{pop}$  has the greatest impact on NSGA-II followed by  $Max\ gen$ ,  $p_c$ , and  $p_m$ , respectively. To tune the parameters of NPGA the Taguchi results are also converted into S/N ratio. Figure 4 shows the optimal level of each factor for this algorithm. From these graphs, it



can be concluded that the most suitable level of this factor are A(1),B(3),C(2), D(1), respectively.

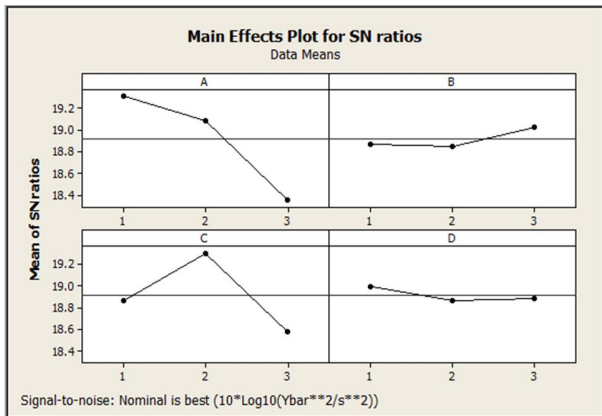
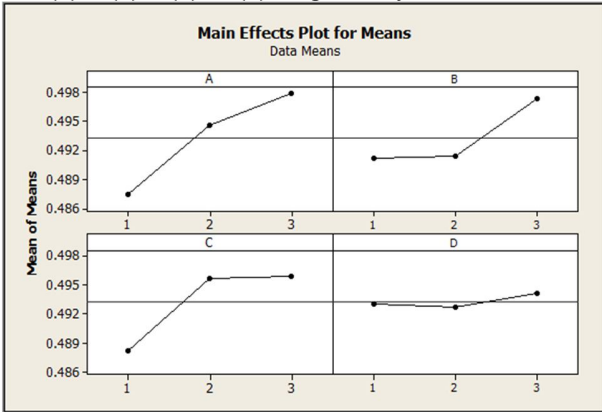


Fig. 4. Mean S/N ratio for level of each factor NRGGA

Table 6  
S/N NRGGA

Response table for signal to noise ratios (nominal is better)				
level	A	B	C	D
1	19.31	18.87	18.86	18.99
2	19.08	18.85	19.30	18.87
3	18.35	19.02	18.57	18.88
Delta	0.96	0.18	0.73	0.13
Rank	1	3	2	4

From figure 4 and the value of delta in Table 6, we can conclude that *npop* has the greatest influence on NRGGA followed by *p<sub>c</sub>*, *Max gen*, and *p<sub>m</sub>*, respectively.

After obtaining the results of the Taguchi experiment for all the trials, the optimal levels of the factors A, B, C, and D are determined and shown in Table 7.

Table 7  
Parameter setting values

Factor	Symbol	Optimal level for each problem	Factor	Symbol	Optimal level for each problem
NSGA-II			MOICA		
<i>npop</i>	A	150	<i>npop</i>	A	100
<i>Max g</i>	B	75	<i>Max gen</i>	B	50
<i>p<sub>c</sub></i>	C	0.85	<i>p<sub>c</sub></i>	C	0.85
<i>p<sub>m</sub></i>	D	0.2	<i>p<sub>m</sub></i>	D	0.05

### 5.2. The computational results

In this section, the performance of the proposed algorithm is evaluated. Table 6 shows the obtained values for the metrics of 10 test problems.

As shown in Table 8, the results of two algorithms are very close. However, the NRGGA optimal solutions are better in terms of three metrics in comparison with the other algorithm. For example, the obtained result for the 10<sup>th</sup> problem shows that the NSGGA outperforms NRGGA according to mean ideal distance (MID) metric with a value of 0.49914, while MID is equal to 0.52294 for the NRGGA (the lower the MID the better). NSGGA is also excels based on the rate of achievement to simultaneous objectives (RAS) metric with a corresponding value of 0.49922 (the lower is better). Finally, it is a superior, according to the spacing metric, SM=0.41704 (the lower, the better). Similarly, the outcomes of NSGGA in the other test problems show that the metrics for the algorithm are more favorable compared to the NRGGA.

Figure 5 illustrates the Pareto solutions obtained from the algorithms for six randomly selected problems out of 10 test problems. As shown in Figure 5, the Pareto solutions obtained by NSGGA and NRGGA are very close, while the results of NSGGA are slightly better in most cases.



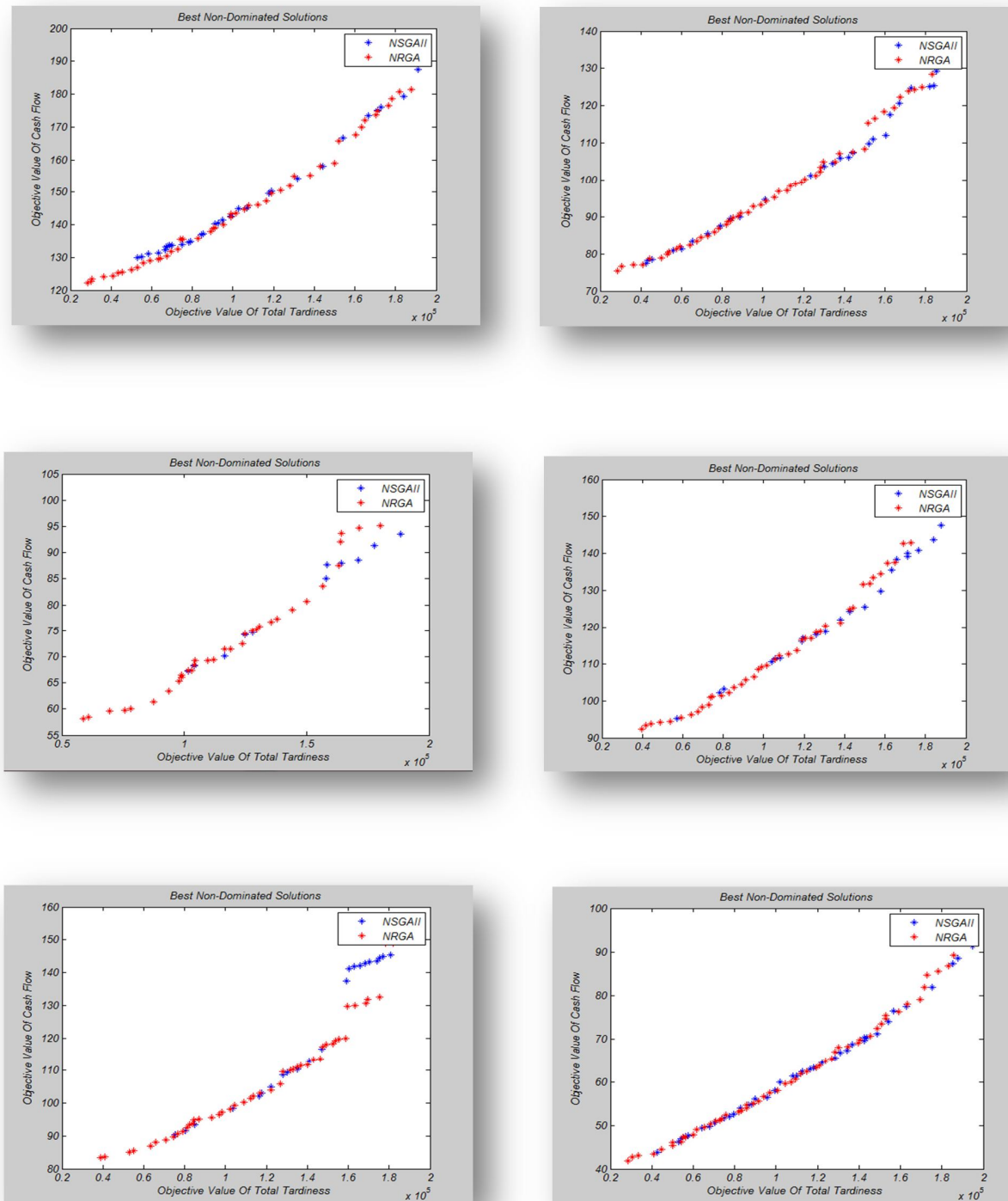


Fig. 5. Pareto solutions obtained from algorithms

Table 8  
Computational results of metrics for the algorithms

Problem/metric	MID metric		RAS metric		Spacing metric	
	NSGA-II	NRGA	NSGA-II	NRGA	NSGA-II	NRGA
<b>Problem 1</b>	0.45122	0.44558	0.45132	0.44568	0.5878	0.7191
<b>Problem 2</b>	0.521	0.505	0.52106	0.50506	0.66756	0.60496
<b>Problem 3</b>	0.38996	0.47008	0.39002	0.47014	0.71982	0.71966
<b>Problem 4</b>	0.5196	0.55194	0.51962	0.55202	0.63806	0.62026
<b>Problem 5</b>	0.46672	0.47068	0.46682	0.4708	0.6451	0.65378
<b>Problem 6</b>	0.51228	0.50812	0.5124	0.50822	0.47418	0.46604
<b>Problem 7</b>	0.47362	0.51244	0.47374	0.5125	0.46456	0.45016
<b>Problem 8</b>	0.5147	0.51978	0.51484	0.51992	0.54892	0.52098
<b>Problem 9</b>	0.51888	0.52686	0.51906	0.52702	0.49028	0.43398
<b>Problem 10</b>	0.49914	0.52294	0.49922	0.52304	0.41704	0.50796

## 6. Conclusion and Future Research

This paper presents a new bi-objective preemption multi-mode resource constrained project scheduling problem based on minimizing weighted tardiness and maximizing net present value. The study is quite novel for bringing in simultaneously the factors of preemption, completing the activity within their earliest and latest finish period and before a stated due date, NPV maximization and tardiness minimization into the multi-objective multi-mode project scheduling problem.

In order to solve this model, two multi-objective meta-heuristic algorithms, NSGAII and NRGA, were proposed. 10 test problems including large and small size problems were solved by applying the proposed algorithms. The obtained results showed the effectiveness of NSGAII and NRGA algorithms for all the problems regardless of their size. In order to evaluate the performance of the algorithms, their results were compared. Whilst the Pareto solutions obtained by NSGAII and NRGA are very close, the results of NSGAII algorithm were slightly better in terms of the assessment metrics.

As a research limitation, this study does not consider budget restriction. Future studies can model the stated problem while budget limitation is also taken into account. Moreover, several other opportunities are open for future research. First, the model would be more practical if the number of preemptions is restricted and each activity could not be stopped more than a specified number of times. Second, the problem can be modeled with respect to a constraint that conveys a minimum duration between preemptions in order to impede frequent and close interruptions within the processing period of an activity. Third, if uncertainty and vagueness appears when determining the activity durations, fuzzy logic can be used to tackle the problem. Finally, it is recommended that the problem is solved using other heuristic and meta-heuristic algorithms. The results can be compared with the ones of this study.

## 7. References

- [1] Alcaraz, J., Maroto, C., Ruiz, R., (2003). Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms. *Journal of the Operational Research Society* 54, 614–626.
- [2] Ballestin, F., Blanco, B., (2011) Theoretical and practical fundamentals for multi-objective optimization in resource-constrained project scheduling problems. *European Journal of Operational Research* 38, 51-62.
- [3] Barrios, A., Ballestin, F., and Valls, V. (2011). A double genetic algorithm for the MRCPSp/max. *Computers & Operations Research*, Vol. 38, No. 1, pp. 33-43.
- [4] Bianco, L., Caramia, M., and Dell'Olmo, P. (1999). Solving coloring techniques. In Weglarz (1999), a preemptive project scheduling problem with pages 135–146.
- [5] Bouleimen, K., Lecocq, H., (2003). A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode versions. *European Journal of Operational Research* 149, 268–281.
- [6] Brucker, P., and Knust, S. (2001). Resource-constrained project scheduling and timetabling. *Lecture Notes in Computer Science*, Vol. 2079, pp. 277-293.
- [7] Brucker, P., Drexel, A., Mähring, R., Neuman, K., Pesch, E., (1999). Resource-constrained project scheduling: Notation, classification, models, and methods. *European Journal of Operational Research* 112, 3–41.
- [8] Buddhakulsomsiri, J., Kim D., (2006). Properties of multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting. *European Journal of Operational Research* 175, 279-295.
- [9] Calhoun, K. M., Deckro, R. F., Moore, J. T., Chrissis, J. W., and Hove, J. C. V. (2002). Planning and replanning in project and production scheduling. *Omega – The international Journal of Management Science*, Vol. 30, No. 3, pp. 155-170.
- [10] Coelho, J., Vanhoucke, M., (2011). Multi-mode resource-constrained project scheduling using RCPSP and SAT solvers. *European Journal of Operational Research* 213, 73-82, 2011.
- [11] Deb, K., Agrawal, Pratap, A., Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In M.S. et al. (Ed.), *Parallel*

- Problem solving from Nature- PPSN VI, Berlin, 849-858. Springer.
- [12] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. *IEEE Transaction on Evolutionary Computation*, 6(2), 182-197.
- [13] Debels, D., Vanhoucke, M., (2007). A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem. *Operations Research* 55 (3), 457-469.
- [14] Debels, D., De Reyck, B., Leus, R., Vanhoucke, M., (2006). A hybrid scatter-search/ electromagnetism meta-heuristic for the resource-constrained project scheduling problem. *European Journal of Operational Research* 169 (3), 638-653.
- [15] Deblaere, F., Demeulemeester, E., Herroelen, W., (2010). Reactive scheduling in the multi-mode RCPSP, *computers & Operation Research* 38, 63-74.
- [16] Demeulemeester, E., Herroelen, W., (1996). An efficient optimal procedure for the preemptive resource-constrained project scheduling problem. *European Journal of Operational Research* 90, 334-348.
- [17] Drexel, A., Nissen, R., Patterson, J. H., and Salewski, F. (2000). Progen/px - an instance generator for resource-constrained project scheduling problems with partially renewable resources and further extensions. *European Journal of Operational Research*, Vol. 125, No. 1, pp. 59-72.
- [18] Goto, E., Joko, T., Fujisawa, K., Katoh, N. and Furusaka, S., (2001). Maximizing Net Present value for Generalized Resource Constrained Project Scheduling Problem, Technical Paper, Nomura Research institute, Department of Architecture and Architectural System, Kyoto, Japan.
- [19] Hartmann, S., Drexel, A., (1998). Project scheduling with multiple modes: A comparison of exact algorithms. *Networks*, 32, 283-297.
- [20] Hartmann, S., Sprecher, A., (1996). A note on hierarchical models for multi-project planning and scheduling. *European Journal of Operational Research* 94, 377- 383.
- [21] Heilmann, R., (2003). A branch-and-bound procedure for the multi-mode resource-constrained project scheduling problem with minimum and maximum time lags. *European Journal of Operational Research* 144, 348-365.
- [22] Herroelen, W., De Reyck, B., Demeulemeester, E., (1999). A classification scheme for project scheduling. In: *Handbook of Recent Advances in Project Scheduling*. Kluwer Academic Publishers, pp. 1-26.
- [23] Jarboui, B., Damak, N., Siarry, P., Rebai, A., (2008). A combinatorial particle swarm optimization for solving multi-mode resource-constrained project scheduling problems. *Applied Mathematics and Computation* 195, 299-308.
- [24] Jozefowska, J., Mika, M., Rozycki, R., Waligora, G., Weglarz, J., (2001). Simulated annealing for multi-mode resource-constrained project scheduling. *Annals of Operations Research* 102, 137-155.
- [25] Kaplan, L., (1988). Resource-Constrained Project Scheduling with Preemption of Jobs. Ph.D. Thesis, University of Michigan.
- [26] Lova, A., Tormos, P., Barber, F., (2006). Multi-mode resource constrained project scheduling: Scheduling schemes, priority rules and mode selection rules. *Inteligencia Artificial* 30, 69-86.
- [27] Mika M., Waligora G., Weglarz, J., (2005). Simulated annealing and tabu search for multi-mode resource constrained project scheduling problem with positive discounted cash flows and different payment models, *European journal of Operational Research*, 164 , 639-667.
- [28] Nonobe, K., Ibaraki, T., (2001). Formulation and Tabu Search Algorithm for the Resource Constrained Project Scheduling Problem (RCPSP). Technical Report, Kyoto University.
- [29] Nudtasomboon, N., Randhawa S., (1997). Resource-constrained project scheduling with renewable and non-renewable resources and time-resource tradeoffs. *European Journal of Operational Research* 32, 227-242.
- [30] Ozdamar, L., (1999). A genetic algorithm approach to a general category project scheduling problem. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 29, 44-59.
- [31] Ranjbar, M., De Reyck, B., Kianfar, F., (2008). A hybrid scatter-search for the discrete time/resource trade-off problem in project scheduling. *European Journal of Operational Research* 193 (1), 35-48.
- [32] Sabzehparvar, M., Seyed Hosseini, M., (2008). A mathematical model for the multi-mode resource-constrained project scheduling problem with mode dependent time lags. 257-273. June.
- [33] Slowinski, R., (1980). Two approaches to problems of resource allocation among project activities - A comparative study. *Journal of Operational Research Society* 8, 711-723.
- [34] Speranza, M., Vercellis, C., (1993). Hierarchical models for multi-project planning and scheduling. *European Journal of Operational Research* 64, 312-325.
- [35] Sprecher, A., Hartmann, S., Drexel, A., (1997). An exact algorithm for the project scheduling with multiple modes. *OR Spektrum* 19, 195-203.
- [36] Sprecher, A., Drexel, A., (1998). Solving multi-mode resource-constrained project scheduling problems by a simple, general and powerful sequencing algorithm. *European Journal of Operational Research*, 107, 431-450.
- [37] Srinivas, N. and Deb, K. (1995) Multi-Objective function optimization using non-dominated sorting genetic algorithms, *Evolutionary Computation*, 2(3):221-248.
- [38] Talbot, F., (1982). Resource-constrained project scheduling with Timeresource Tradeoffs: The nonpreemptive case. *Management Science* 28, 1197-1210.
- [39] Varma, V. A., Uzsoy, R., Pekny, J., and Blau, G. (2007). Lagrangian heuristics for scheduling new product development projects in the pharmaceutical industry. *Journal of Heuristics*, Vol. 13, No. 5, pp. 403-433.
- [40] Voß, S., Witt, A., (2007). Hybrid flow shop scheduling as a multi-mode multi-project scheduling problem with batching requirements: A real-world application. *International journal of production economics*. 105, 445-458.
- [41] Zhang, H., Tam, C., Li, H., (2006). Multimode project scheduling based on particle swarm optimization. *Computer-Aided Civil and Infrastructure Engineering*, 21, 93-103.
- [42] Zhu, G., Bard, J., Tu, G., (2006). A branch-and-cut procedure for the multimode resource-constrained project-scheduling problem. *Journal on Computing* 18 (3), 377-390.

