

Research Article

A Priority Rule-Based Heuristic for Resource Investment Project Scheduling Problem with Discounted Cash Flows and Tardiness Penalties

Amir Abbas Najafi¹ and Fatemeh Azimi²

¹ *Department of Industrial Engineering, Qazvin Islamic Azad University, Nokhbegan Avenue, 34185-1416 Qazvin, Iran*

² *Department of Mathematics, Qazvin Islamic Azad University, Nokhbegan Avenue, 34185-1416 Qazvin, Iran*

Correspondence should be addressed to Amir Abbas Najafi, aa_najafi@yahoo.com

Received 4 November 2008; Revised 15 April 2009; Accepted 11 May 2009

Recommended by Wei-Chiang Hong

Resource investment problem with discounted cash flows (RIPDCF) is a class of project scheduling problem. In RIPDCF, the availability levels of the resources are considered decision variables, and the goal is to find a schedule such that the net present value of the project cash flows optimizes. In this paper, we consider a new RIPDCF in which tardiness of project is permitted with defined penalty. We mathematically formulated the problem and developed a heuristic method to solve it. The results of the performance analysis of the proposed method show an effective solution approach to the problem.

Copyright © 2009 A. A. Najafi and F. Azimi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

Project scheduling is a major objective of most models, and methods propose to aid planning and management of projects. Project scheduling problems are combination of precedence constraints, resource constraints, and some side constraints in which the goal is to find a schedule such that an objective function like project duration, project total costs, or net present value (NPV) optimizes. For a comprehensive survey of project scheduling problems refer to [1–4].

Resource investment problem (RIP) is a class of project scheduling problem. RIP is known as the problem of minimizing renewable resource costs subject to a project due date. In RIP, we are concerned about completing a project consisting of a set of activities, such that a given deadline is met in time and a set of resources needed for the execution of the activities over the project is utilized. Since costs incur to provide resources, the goal is to find a schedule and resource requirement levels such that total costs of the resource utilizations minimizes.

Möhring [5] introduced the resource investment problem and proved its NP-Hardness. In addition, based on a graph algorithm, he proposed an exact solution method and solved some examples with 16 activities and 4 resources. Demeulemeester [6] developed another exact algorithm called Resource Availability Cost Problem for the RIP. His algorithm relies on the branch and bound method for single mode resource constrained project scheduling problem of Demeulemeester and Herroelen [7]. Drexl and Kimms [8] presented the lower and the upper bounds using Lagrangian relaxation and column generation techniques. Akpan [9] proposed a heuristic procedure to solve RIP. Shadrokh and Kianfar [10] developed a Genetic Algorithm for RIP. The objective is to minimize the sum of resource availability costs and tardiness penalty. As an extension of the RIP, we encounter the RIP/max problem in literature. In this problem the precedence constraints of RIP extend to generalized precedence relations where we need to observe the minimum and the maximum time lags between the activity starting and completion times. In order to solve this problem, Nübel [11] introduced a generalization of RIP/max and developed a depth-first branch-and-bound procedure to solve it.

Many of the recent researches in project scheduling focus on maximizing the NPV of the project using the sum of positive and negative discounted cash flows throughout the life cycle of the project. Russell [12] introduced the problem of maximizing NPV in project scheduling problem and proposed a successive approximation approach to solve it. Grinold [13] added a project deadline to the model, formulated the problem as a linear programming problem, and proposed a method to solve it. Doersch and Patterson [14] presented a zero-one integer-programming model for the NPV problem. Their model included a constraint on capital expenditure of the activities in the project, while the available capital increased as progress payments were made. Bey et al. [15] considered the implications of a bonus/penalty structure on optimal project schedules for the NPV problem. Russell [16] considered the resource-constrained NPV maximization problem. He introduced priority rules for selecting activities for resource assignment based upon information derived from the optimal solution to the unconstrained problem. Smith-Daniels and Smith-Daniels [17] extend the Doersch and Patterson [14] Zero-one formulation to accommodate material management costs. Icmeli and Erengus [18] introduced a branch and bound procedure to solve the resource constrained project-scheduling problem with discounted cash flows. Nadjafi and Shadrokh [19] proposed an exact branch and bound procedure for the unconstrained project scheduling problem with the objective of minimizing weighted earliness-tardiness penalty costs. For other related studies in the project NPV maximization, see [20–29].

In addition, Najafi and Niaki [30, 31] introduced a new resource investment problem in which the goal was to maximize the discounted cash flows of the project payments and called it resource investment problem with discounted cash flows (RIPDCF). The cash flows may be either the project costs or the payments made for the project during its life cycle. In this regard, they considered both the payments and the employment-releasing times of the resources. They mathematically formulated the problem and showed that it is an NP-hard problem.

In [30, 31], the model imposed deadline constraint on the completion time of the project. It means that the activities are to be scheduled such that the makespan of the project does not exceed a given deadline. In many real problems, project can be finished with delay but facing with the delay penalty. In practice, in order to make the project to be finished as soon as possible and to prevent it from being postponed beyond the deadline, setting a penalty mechanism at the deadline of the project is a usual measure taken by the clients. If the project completion is delayed beyond the deadline of the project, the total amount of the

payments will be reduced by a certain extent as the penalty. Obviously, in the problem with penalty structure there exist two trade-offs for the contractor. The first one occurs between the expense increment caused by speeding up the progress of the project, and the second one happens between saving the cost by slowing down the progress of the project and receiving the punishment because of postponing the project completion. In other word, the contractor must make trade-off between the performing modes of activities and penalty structure that depends upon the project completion time, and the client has to make decision on how to set the penalty structure effectively and economically.

In this paper we consider an RIPDCF in which tardiness is permitted with delay penalty and call it RIPDCFT. The rest of the paper is organized as follows. In Section 2, we define the problem precisely. Then in Section 3, we formulate the problem. In Section 4, we propose a heuristic solution to the problem. We report the performance of the proposed method in Section 5, and finally the conclusion comes in Section 6.

2. Problem Description

A project is given with a set of N activities indexed from 1 to N . Activities 1 and N are dummies that represent the start and completion of the project, respectively. The activities executions need K types of renewable resources. There are no resources at the initial of the project available, so it is necessary to provide the required levels of the resources at the activity execution time. In addition, the expulsion time of each resource type must be provided deterministically. Between the providing and the expulsion time of each resource type, availability level of the resource is equal to the provided level of the resource. Zero-lag finish-to-start precedent constraints are imposed on the sequencing of the activities. For each activity i , the precedent activity set is denoted as $P(i)$. A duration D_i is given where activity i is started and it runs D_i time without preemption. Activity i uses r_{ik} units per period for resource k . The resource usage over an activity is taken to be uniform. A cost of C_k is associated to use one unit of resource k per period of time. In addition to resources usage cost, each activity has some other costs such as material or overhead costs. We call these fixed costs. Fixed cost occurs over activity execution, and its amount at period t for activity i is denoted by F_{it} . Payments are received at payment points $g \in G$, where G is the set of payment points. Payment g occurs when a set of activities $PB(g)$ ends, and its amount is equal to Mg . A constant cost of CT for each unit of time delay from a due date DD is incurred. Also, α is the discount rate.

3. Problem Formulation

According to assumptions and notations introduced in Section 2, the problem can be formulated as follows:

$$\begin{aligned} \text{Max } Z = & \sum_{g=1}^G M_g e^{-\alpha T_g} - \sum_{i=1}^N \left(\sum_{t=0}^{d_i-1} F_{it} e^{-\alpha t} \right) e^{-\alpha S_i} - \sum_{k=1}^K \sum_{t=SR_k}^{FR_k-1} C_k R_k e^{-\alpha t} \\ & - (CT * \text{Max}\{0, S_N - DD\}) e^{-\alpha S_N}, \end{aligned} \quad (3.1)$$

or

$$\begin{aligned} \text{Max } Z = & \sum_{g=1}^G M_g e^{-\alpha T_g} - \sum_{i=1}^N \sum_{t=0}^{d_i-1} F_{it} e^{-\alpha(t+S_i)} - \sum_{k=1}^K C_k R_k \left(\frac{e^{-\alpha SR_k} - e^{-\alpha FR_k}}{1 - e^{-\alpha}} \right) \\ & - (CT * \text{Max}\{0, S_N - DD\}) e^{-\alpha S_N}. \end{aligned} \quad (3.2)$$

Subject to

$$S_i - S_j \geq d_j, \quad \forall j \in P(i) \quad i = 1, 2, \dots, N, \quad (3.3)$$

$$T_g \geq S_i + d_i, \quad \forall i \in PB(g), \quad g = 1, 2, \dots, G, \quad (3.4)$$

$$SR_k \leq S_i, \quad \forall i \in PR(k), \quad k = 1, 2, \dots, K, \quad (3.5)$$

$$FR_k \geq S_i + d_i, \quad \forall i \in UR(k), \quad k = 1, 2, \dots, K, \quad (3.6)$$

$$\sum_{i=1}^N \sum_{t=d_i+1}^t r_{ik} X_{it} \leq R_k, \quad k = 1, 2, \dots, K, \quad \forall t, \quad (3.7)$$

$$\sum_{t=ES_i}^{LS_i} X_{it} = 1, \quad i = 1, 2, \dots, N, \quad (3.8)$$

$$S_i = \sum_{\forall t} t X_{it}, \quad i = 1, 2, \dots, N, \quad (3.9)$$

$$X_{it} \in \{0, 1\}, \quad i = 1, 2, \dots, N, \quad \forall t, \quad (3.10)$$

$$S_i \geq 0, \quad i = 1, 2, \dots, N, \quad (3.11)$$

$$R_k \geq 0, \quad k = 1, 2, \dots, K, \quad (3.12)$$

$$SR_k, FR_k \geq 0, \quad k = 1, 2, \dots, K, \quad (3.13)$$

$$T_g \geq 0, \quad g = 1, 2, \dots, G. \quad (3.14)$$

Where, we define the decision variables as

- S_i : starting time of activity i , $i = 1, 2, \dots, N$,
- T_g : occurrence time for payment g , $g = 1, 2, \dots, G$,
- R_k : required level of resource k to be provided, $k = 1, 2, \dots, K$,
- SR_k : providing time of resource k , $k = 1, 2, \dots, K$,
- FR_k : expulsion time of resource k , $k = 1, 2, \dots, K$,
- X_{it} : a binary variable where it is one if activity i is started at period t and zero otherwise.

In addition, $PR(k)$ is an activity set that uses resource k and has no precedence, and $UR(k)$ is an activity set that uses resource k and has no successor.

The objective function (3.1) maximizes the net present value of the project. It includes positive effects of the present values of the payments, negative effects of the present values of

the fixed costs, negative effects of the present values of the costs for providing the resources and negative effects of the present value of the tardiness costs. Equation (3.3) enforces the precedent relations between activities. Equation (3.4) guarantees that payments occur when required activities have been finished. Constraints (3.5) and (3.6) correspond to the providing and the expulsion times of the resources. Equation (3.7) ensures that the provided resource units are sufficient to implement the schedule. Equation (3.8) states that every activity must be started only once. Equation (3.9) states the relationship between variables S_j and variables X_{it} . Sets of constraints (3.10), (3.11), (3.12), (3.13), and (3.14) denote the domain of the variables.

4. A Priority Rule-Based Heuristic

In this section, based on the priority rules of the RIPDCF, we propose a heuristic method to solve the problem. To do this, first we state some definitions that are required in the procedure.

Definition 4.1. Negative cash flow of an activity. It includes discounted cash flow of the resource usage cost and fixed cost at the activity starting time. It can be stated as

$$CF_i^- = -\sum_{t=0}^{d_i-1} F_{it}e^{-at} - \sum_{k=1}^K \sum_{t=0}^{d_i-1} r_{ik}C_k e^{-at} = -\sum_{t=0}^{d_i-1} F_{it}e^{-at} - \sum_{k=1}^K r_{ik}C_k \left(\frac{1 - e^{-ad_i}}{1 - e^{-a}} \right). \quad (4.1)$$

Definition 4.2. Positive cash flow of an activity. If the precedent activity set of payment occurrence contains only one activity, then we set positive cash flow of the activity to be equal to the discounted cash flow of that payment at the activity starting time. In this case, we define the positive cash flow of the activity as

$$CF_i^+ = M_g e^{-ad_i}. \quad (4.2)$$

If the precedent activity set of payment occurrence contains more than one activity, then we create a dummy activity and set positive cash flow of the dummy activity to be equal to that payment. In this case, the number of the project activities may increase to M . In the following sections we denote the number of activities by M .

Definition 4.3. Cash flow of an activity. Cash flow of an activity equals to the sum of the negative and the positive cash flows of an activity. In other words, we have

$$CF_i = CF_i^- + CF_i^+. \quad (4.3)$$

Definition 4.4. The amount of nonusage resource at a period. With (3.7) modified, the amount of nonusage resource k at a period t , W_{kt} , can be obtained by

$$\sum_{i=1}^M \sum_{l=t-d_i+1}^t r_{ik}x_{il} + W_{kt} = R_k. \quad (4.4)$$

Where,

$$W_{kt} \geq 0, \quad k = 1, 2, \dots, K, \quad \forall t. \quad (4.5)$$

Now, we simplify the problem formulation in the following form:

$$\text{Max } Z = \sum_{i=1}^M CF_i e^{-\alpha S_i} - \sum_{k=1}^K \sum_{t=SR_k}^{FR_k-1} C_k W_{kt} e^{-\alpha t} - (CT * \text{Max}\{0, S_M - DD\}) e^{-\alpha S_M}. \quad (4.6)$$

Subject to (3.3), (3.5), (3.6), (4.4), (4.5), (3.8), (3.9), (3.10), (3.11), (3.12), and (3.13).

In order to develop the solution procedure, we use the structure of the objective function given in (4.6). The double structure includes positive roles of the activities cash flows, $(\sum_{i=1}^M CF_i e^{-\alpha S_i})$, and the negative roles of the nonusage resource costs, $(\sum_{k=1}^K \sum_{t=SR_k}^{FR_k-1} C_k W_{kt} e^{-\alpha t})$ and the tardiness cost $((CT * \text{Max}\{0, S_M - DD\}) e^{-\alpha S_M})$.

Now we are ready to describe the executive steps of the proposed algorithm as follows:

Step 1. Let problem P be the RIPDCFT that we are interested to solve, and let P_{sub} be a problem obtained by removing resources of the P problem. The P_{sub} problem can be reached from the P problem by removing the resource constraints and negative roles of nonusage resource costs and the tardiness cost in the objective function. In addition, we add constraint (4.7) to the P_{sub} problem. It imposed deadline constraint on the project completion time in this step:

$$S_M \leq DD. \quad (4.7)$$

The P_{sub} problem can be described as follows:

$$\text{Max } Z_{\text{sub}} = \sum_{i=1}^M CF_i e^{-\alpha S_i}. \quad (4.8)$$

Subject to (3.3), (3.11), and (4.7).

The P_{sub} problem is a project scheduling problem with discounted cash flows and can be solved exactly [12]. Call the acquired problem as Active Problem, solve it by related methods, and obtain the optimum value of its objective function. Call the optimum solution as active scheduling. Now, enter the resources at active scheduling and determine the maximum of usage level for each type of resources. If we set the required level of each provided resource equal to the maximum of usage level of the resources, then the active scheduling is a feasible solution for the P problem and you can obtain the providing and expulsion time of each resource and obtain the discounted nonusage cost of each resource from the following equation:

$$U_k = \sum_{t=SR_k}^{FR_k-1} C_k W_{kt} e^{-\alpha t}. \quad (4.9)$$

Now, calculate the objective function value of the master problem at this solution by the following expression and call it the active objective function value:

$$Z = Z_{\text{Sub}} - \sum_{k=1}^K U_k. \quad (4.10)$$

Step 2. Eliminate constraint (4.7) from the active problem.

Step 3. Add all resources in a set, named resource candidate list.

Step 4. From the list of resource candidates, select the resource with the highest discounted cost of nonusage (U_k). If the providing level of the selected resource has not reached its lower bound, decrease its value by one unit, solve the active problem by adding the resource constraint with the acquired value, and determine the optimum value of the objective function [23]. In order to increase the value of the objective function defined for RIPDCFT problem it may be beneficial to shift certain activities. Activities shifting procedure attempts to shift the position of the activities whenever precedence and resource constraints allow, and it results in increasing the NPV of the project [31]. In the acquired solution, consider the maximum of the usage level of each resource as providing level. Then calculate the discounted nonusage cost for each type of resource by (4.9). In addition, determine the project completion time (S_M) for this solution and obtain the tardiness cost from the following equation:

$$V = (CT * \text{Max}\{0, S_M - DD\})e^{-\alpha S_M}. \quad (4.11)$$

Now, determine the objective function value of the P problem by following equation:

$$Z = Z_{\text{Sub}} - \sum_{k=1}^K U_k - V. \quad (4.12)$$

Call it the temporary objective function value. However, if the providing resource value reached its lower bound, go to step six. You can obtain the lower bound of the resource using the following expression:

$$R_k = \text{Max} \left\{ \frac{\sum_{i=1}^M (r_{ik} \times d_i)}{DD}, \text{Max}_{i=1, \dots, M} \{r_{ik}\} \right\}. \quad (4.13)$$

Step 5. If the temporary objective function value is more than the active objective function value add the selected constraint to the active problem. Then, consider the acquired problem, related scheduling, and the temporary objective function value as an active problem and go to step four. Otherwise, do not add the selected resource constraint to the active problem.

Step 6. Eliminate the selected resource from the resource candidate list and go to step seven.

Step 7. If the resource candidate list is empty, stop. The active schedule is the solution of the proposed algorithm. Otherwise, go to step four.

Table 1: Computational results.

No. of Activities	No. of Instances	A	Avg. of dev%	Avg. of LINGO CPU time (Sec.)	Avg. of the proposed method CPU time (Sec.)
10	15	10	-1.9%	388	1
20	15	7	-2.3%	1253	2
30	15	4	-2.8%	2407	4

5. Performance Evaluation

In this section, we present the performance of the proposed procedure introduced in the previous sections. For the purpose of this section we needed a set of solved problems. Since the RIPDCFT is a newly defined problem, no standard test problems could be found to examine the performance of the proposed procedure. Therefore, we are forced to use the RIPDCF test problems, suggested by Najafi and Niaki [31]. In order to generate the tardiness cost value, we randomly generate it from a uniform distribution. We consider 45 instances with 10, 20, and 30 activities with 3, 4, and 5 resources for the experiments. We coded a computer program of the procedure, and then we employed the program on the test problems. To evaluate the performance of the procedure we needed some good solutions. Since there was no other existing procedure to solve the problem, we solved the mathematical modeling of the test problems by solver software such as LINGO [32]. However due to the nature of the problem, LINGO was unable to obtain a global optimal solution for all the test problems. In these cases, we assumed that the solution obtained by LINGO was a good one to compare. We perform the experiments on a PC with 2.1 GHz processor and 2046 MB RAM, limiting the solution time to be less than or equal to 3600 CPU seconds.

Table 1 shows the computational results of the proposed method in which column A denotes the number of instances; LINGO was able to find a local optimal solution in 3600 CPU seconds.

The results of experiments showed that (a) there are many instances that the solver software is unable to solve in 3600 seconds, but there is a solution by the proposed method, (b) for problems in which LINGO was able to find a solution, there is no significant difference between the solutions obtained by LINGO and the ones obtained by the proposed method and (c) while actually there is no difference between the solutions obtained by LINGO and the proposed method, the amount of CPU time for the proposed method is much less than that of those obtained by LINGO.

6. Conclusions

In this paper, we introduced a new resource investment problem with discounted cash flows in which tardiness is permitted with penalty. We mathematically formulated the problem. In order to solve the problem we came up with a heuristic approach and through some generated test problems, we showed that it works relatively well.

The extension of this research would be to investigate an RIP/max problem in which the goal is to maximize the NPV of the project and tardiness is permitted with penalty, too. One of the other potential interests would be to develop some metaheuristics methods, such

as genetic algorithm, simulated annealing, neural networks, ant colony algorithm, and so forth, to solve the problem.

Acknowledgements

The authors thank the anonymous referees for their useful suggestions that improved the presentation of the paper.

References

- [1] O. Icmeli, S. S. Erengüç, and C. J. Zappe, "Project scheduling problems: a survey," *International Journal of Operations and Production Management*, vol. 13, no. 11, pp. 80–91, 1993.
- [2] S. E. Elmaghraby, "Activity nets: a guided tour through some recent developments," *European Journal of Operational Research*, vol. 82, no. 3, pp. 383–408, 1995.
- [3] W. Herroelen, B. De Reyck, and E. Demeulemeester, "Resource-constrained project scheduling: a survey of recent developments," *Computers & Operations Research*, vol. 25, no. 4, pp. 279–302, 1998.
- [4] E. Demeulemeester and W. Herroelen, *Project Scheduling: A Research Handbook*, Kluwer Academic Publishers, Boston, Mass, USA, 2002.
- [5] R. H. Möhring, "Minimizing costs of resource requirements in project networks subject to a fixed completion time," *Operations Research*, vol. 32, no. 1, pp. 89–120, 1984.
- [6] E. L. Demeulemeester, "Minimizing resource availability costs in time-limited project networks," *Management Science*, vol. 41, no. 10, pp. 1590–1598, 1995.
- [7] E. L. Demeulemeester and W. Herroelen, "A branch-and-bound procedure for the multiple resource-constrained project scheduling problem," *Management Science*, vol. 38, no. 12, pp. 1803–1818, 1992.
- [8] A. Drexel and A. Kimms, "Optimization guided lower and upper bounds for the resource investment problem," *Journal of the Operational Research Society*, vol. 52, no. 3, pp. 340–351, 2001.
- [9] E. O. P. Akpan, "Optimum resource determination for project scheduling," *Production Planning and Control*, vol. 8, no. 5, pp. 462–468, 1997.
- [10] S. Shadrokh and F. Kianfar, "A genetic algorithm for resource investment project scheduling problem, tardiness permitted with penalty," *European Journal of Operational Research*, vol. 181, no. 1, pp. 86–101, 2007.
- [11] H. Nübel, "The resource renting problem subject to temporal constraints," *OR Spektrum*, vol. 23, no. 3, pp. 359–381, 2001.
- [12] A. H. Russell, "Cash flows in networks," *Management Science*, vol. 16, no. 5, pp. 357–373, 1970.
- [13] R. C. Grinold, "The payment scheduling problem," *Naval Research Logistics Quarterly*, vol. 19, no. 1, pp. 123–136, 1972.
- [14] R. H. Doersch and J. H. Patterson, "Scheduling a project to maximize its present value: a zero-one programming approach," *Management Science*, vol. 23, no. 8, pp. 882–889, 1977.
- [15] R. B. Bey, R. H. Doersch, and J. H. Patterson, "The net present value criterion: its impact on project scheduling," *Project Management Quarterly*, vol. 12, pp. 35–45, 1981.
- [16] R. A. Russell, "A comparison of heuristics for scheduling projects with cash flows and resource restrictions," *Management Science*, vol. 32, no. 10, pp. 1291–1300, 1986.
- [17] D. E. Smith-Daniels and V. L. Smith-Daniels, "Maximizing the net present value of a project subject to materials and capital constraints," *Journal of Operations Management*, vol. 7, no. 1-2, pp. 33–45, 1987.
- [18] O. Icmeli and S. S. Erenguc, "A branch and bound procedure for the resource constrained project scheduling problem with discounted cash flows," *Management Science*, vol. 42, no. 10, pp. 1395–1408, 1996.
- [19] B. A. Nadjafi and S. Shadrokh, "An algorithm for the weighted earliness-tardiness unconstrained project scheduling problem," *Journal of Applied Sciences*, vol. 8, no. 9, pp. 1651–1659, 2008.
- [20] S. E. Elmaghraby and W. S. Herroelen, "The scheduling of activities to maximize the net present value of projects," *European Journal of Operational Research*, vol. 49, no. 1, pp. 35–49, 1990.
- [21] K. K. Yang, F. B. Talbot, and J. H. Patterson, "Scheduling a project to maximize its net present value: an integer programming approach," *European Journal of Operational Research*, vol. 64, no. 2, pp. 188–198, 1993.
- [22] G. Ulusoy and L. Özdamar, "A heuristic scheduling algorithm for improving the duration and net present value of a project," *International Journal of Operations and Production Management*, vol. 15, no. 1, pp. 89–98, 1995.

- [23] S. M. Baroum and J. H. Patterson, "The development of cash flow weight procedures for maximizing the net present value of a project," *Journal of Operations Management*, vol. 14, no. 3, pp. 209–227, 1996.
- [24] J. P. Pinder and A. S. Marucheck, "Using discounted cash flow heuristics to improve project net present value," *Journal of Operations Management*, vol. 14, no. 3, pp. 229–240, 1996.
- [25] N. Dayanand and R. Padman, "On modeling payments in projects," *Journal of the Operational Research Society*, vol. 48, no. 9, pp. 906–918, 1997.
- [26] C. Sepil and N. Ortac, "Performance of the heuristic procedures for constrained projects with progress payments," *Journal of the Operational Research Society*, vol. 48, no. 11, pp. 1123–1130, 1997.
- [27] B. De Reyck and W. Herroelen, "An optimal procedure for the resource-constrained project scheduling problem with discounted cash flows and generalized precedence relations," *Computers and Operations Research*, vol. 25, no. 1, pp. 1–17, 1998.
- [28] G. Ulusoy, F. Sivrikaya-Şerifoğlu, and Ş. Şahin, "Four payment models for the multi-mode resource constrained project scheduling problem with discounted cash flows," *Annals of Operations Research*, vol. 102, no. 1–4, pp. 237–261, 2001.
- [29] G. Waligóra, "Discrete-continuous project scheduling with discounted cash flows-a tabu search approach," *Computers and Operations Research*, vol. 35, no. 7, pp. 2141–2153, 2008.
- [30] A. A. Najafi and S. T. A. Niaki, "Resource investment problem with discounted cash flows," *International Journal of Engineering, Transactions B*, vol. 18, no. 1, pp. 53–64, 2005.
- [31] A. A. Najafi and S. T. A. Niaki, "A genetic algorithm for resource investment problem with discounted cash flows," *Applied Mathematics and Computation*, vol. 183, no. 2, pp. 1057–1070, 2006.
- [32] <http://www.Lindo.com>.