Optimal constrained non-renewable resource allocation in PERT networks with discrete activity times

S.S. Hashemin\textsuperscript{a*}, S.M.T. Fatemi Ghomi\textsuperscript{b}, M. Modarres\textsuperscript{c}

\textsuperscript{a} Department of Industrial Engineering, Ardabil Branch, Islamic Azad University, Ardabil, P.O. Box 56157-31567, Iran
\textsuperscript{b} Department of Industrial Engineering, Amirkabir University of Technology, Tehran, P.O. Box 15875-4413, Iran
\textsuperscript{c} Department of Industrial Engineering, Sharif University of Technology, Tehran, P.O. Box 11155-8639, Iran

Received 26 January 2011; revised 25 January 2012; accepted 20 February 2012

Abstract In this paper, we develop an approach to optimally allocate a limited nonrenewable resource among the activities of a project, represented by a PERT-Type Network (PTN). The project needs to be completed within some specified due date. The objective is to maximize the probability of project completion on time. The duration of each activity is an arbitrary discrete random variable and also depends on the amount of consumable resource allocated to it. On the basis of the structure of networks, they are categorized as either reducible or irreducible. For each network structure, an analytical algorithm is presented. Through some examples, the algorithms are illustrated.

1. Introduction

In many real world projects, the duration of activities is stochastic. This is why these projects are formulated as a PERT-Type Network (PTN). On the other hand, completion of a project on time has a significant effect on its cost, revenue and usefulness. Therefore, the main objective of project managers is to avoid any delay. To achieve this goal, consuming extra resources can shorten the duration of each individual activity. However, due to limitations, optimal allocation of resources among activities is vital.

There are many studies in literature regarding resource allocation, in general. The estimation of completion time in PTN is closely related to the constrained resource allocation problem. This subject is surveyed in [1–7]. To see the classification of models and methods in resource-constrained project scheduling, one may refer to [8]. Furthermore, Herroelen et al. [9] presented a new classification, compatible to machine scheduling, and [10] also surveyed the recent developments in resource-constrained project scheduling. Herroelen and Leus [11] discussed the scheduling problem under uncertainty, as well as some research potentials.

Igelmund and Rademacher [12,13] and Mohring et al. [14,15] studied a stochastic, resource constrained, project scheduling problem. They assumed that the durations of project activities, equivalent to operations in job shop scheduling, were ransom. They also assumed that joint distribution of the duration of activities was known. They considered simple resource solutions, as well as schedules, created from a combination of several simple resource solutions. The authors proved various analytical properties for these classes of schedule. Golenko-Ginzburg [16] developed a two-level decision making model for controlling stochastic projects and also introduced some heuristic procedures to solve them. Martel and Ouellet [17] examined the problem of allocating a particular resource among partially interchangeable activities by formulating it as a stochastic program and then reducing it to a deterministic convex allocation problem through parametric programming. In Ref. [18], minimization is usually carried out by the steepest descent gradient search with simulation. Derivatives, with respect to parameters of an allocation, are estimated by simulating at one value of the parameters. Fernández et al. [19] considered nonanticipativity constraints, provided potentially unattainable solutions and also developed...
commercial software for stochastic project scheduling. Golenko-Ginzburg et al. [20] developed a hierarchical three-level decision making model, upper level (company level), medium level (project level) and subnetwork level. The main goal is to develop a unified three-level decision making model, and to indicate planning and control action and optimization problems for all levels. Bowers [21] showed that in a project dominated by technological dependencies rather than resource constraints, the sources of risk can be identified by examining the probabilities of each activity lying on a critical path. Similar criticality probabilities can also be derived for resource constrained stochastic networks, if the definition of the critical path is revised. Golenko-Ginzburg and Gonik [22] maximized the total contribution of accepted activities to the expected project duration by applying zero–one integer programming. The contribution of each activity is the product of the average duration of the activity and its probability of being on the critical path. Golenko-Ginzburg and Gonik [23] presented a new heuristic control algorithm for stochastic network projects, and [24] developed a lookover heuristic algorithm for resource-constrained in PTN. Each activity is of random duration, depending on the resource amounts assigned to that activity. The aim is to minimize the expected project duration. Tsai and Gemmill [25] proposed a tabu search technique to solve stochastic resource-constrained projects. Mohring and Stork [26] introduced some linear pre-selective policies by combining the benefits of pre-selective and priority policies, and derived some efficient algorithms. Gökbayrak and Cassandras [27] transformed the stochastic discrete resource allocation problem into an on-line surrogate continuous optimization problem and proceeded to solve the latter using a standard gradient-based approaches. Then, this surrogate problem methodology was generalized [28]. Golenko-Ginzburg et al. [29] developed an optimization procedure to maximize the probability confidence for project due-dates under budget constraints, or to minimize the project budget under due date chance constraints. A chance-constrained programming model was reviewed from the point of view of accuracy and validity in [30], and they obtained a lower bound for the cumulative distribution function of project completion time. Elmaghraby [31] proposed a dynamic programming approach for a problem with n jobs, processed by single and multiple processors, which has some similarities with the constrained resource allocation problem. Choi et al. [32] developed a new approach to combine heuristic solutions through dynamic programming in the state space generated by heuristics. Azaron and Memariani [33] developed a bicriteria model for the resource allocation problem in PTN in which the total direct costs of the project, as well as the project completion mean time, are the objectives to minimize. Azaron and Tavakkoli-Moghaddam [34] developed a multi-objective model for the resource allocation problem in a dynamic PERT network, where the activity durations are exponentially distributed random variables and the new projects are generated according to a Poisson process. Azaron et al. [35] developed a multi-objective model for resource allocation problems in PERT networks, with exponentially or Erlang distributed activity durations, where the mean duration of each activity is a non-increasing function and the direct cost of each activity is a non-decreasing function of the number of resources allocated to it. The objective functions are the total direct costs of the project (to be minimized), the mean of the project completion time (min), the variance of project completion time (min), and the probability that the project completion time does not exceed a certain threshold (max). The surrogate worth trade-off method is used to solve a discrete-time approximation of the original problem. The resource allocation problem, under stochastic conditions, for multimodal activity networks, was presented in [36]. This problem is solved by dynamic programming, and an approximation scheme was suggested. Elmaghraby [37] showed gross errors can be committed in cost estimates if random variables are replaced by their averages. A metaheuristic algorithm for a resource constrained project scheduling problem in PERT networks, using a hybrid scatter search approach, is developed in [38].

Some researchers considered the problem from other points of view. Haga and O’keefe [39] used a crashing strategy to minimize the value of total cost in PTN. Vanhoucke et al. [40] and Yang et al. [41] paid attention to maximizing the project net present value. Laslo [42] and Vanhoucke et al. [43] considered time-cost trade-offs, under constraints, in the project scheduling problem. Tavares [44] suggested a stochastic model for controlling project duration and expenditure, and deduced to assess the financial risk of a project. Shipley et al. [45] proposed fuzzy probability instead of Beta distribution in PTN and estimated fuzzy expected completion time. Pugh [46] compared fuzzy allocation with a random, largest queue, using a discrete event simulation model.

Elmaghraby [47] minimized the project completion time by using dynamic programming for a network with deterministic activities. To the best of our knowledge, [48] is the only paper in the literature to address the issue of optimal allocation of constrained consumable resources, among the activities of a PERT-type network (PTN), if the objective is to maximize the probability of project completion within some specified period of time (due date). Modarres et al. [48] allocated a limited resource among the activities of a PTN, where the durations of activities are a continuous random variable. Due to the complexity of computations, they applied a hybrid algorithm to obtain a Cumulative Distribution Function (CDF) of completion time, approximately. Bein et al. [49] determined the minimum number of activities to condition upon. This method is efficient when the random variables of activity times are common, and their convolutions, or a maximum of them, are known, for instance, exponential distributions, otherwise their proposed algorithm may be more complex. Tereso et al. [50] shall assume that the activity work content follows exponential distribution. Also, they optimized an economic objective in the face of uncertainty.

However, in this paper, by assuming that the activity durations are discrete random variables, and also, by applying dynamic programming, the exact maximum value of the CDF of the network completion time is obtained for different types of network. Consequently, the constrained resource can be allocated optimally among the activities.

This paper has developed an analytical method for optimal resource allocation in small projects. We believe that this is very valuable, because many researchers believe that it is impossible or very difficult to do. For example, Wan [18] says: “To minimize the expected length of a stochastic CPM-type network by allocating resources optimally is analytically insolvable and numerically impractical”. So, our proposed method may be impractical for large scale problems, but can be a suitable tool for the evaluation of heuristic methods that may be proposed in the future.

The paper has the following structure. First, the problem is described. Then, we develop an analytical approach to solve the problem in series and parallel configurations. Also, we categorize the networks as irreducible and reducible structures.
The optimal constrained resource allocation for irreducible and reducible network structures is described in two distinct sections. The last section is devoted to conclusions and recommendations for future studies.

2. Description of the problem

Consider a project formulated as a PERT-Type Network (PTN). The duration time of each activity is an arbitrary independent discrete random variable with a given probability function. However, this probability function depends on the amount of resources allocated to it. Clearly, the amount of resources allocated to each activity is limited to some specific levels. The objective is to allocate the total constrained resources among the activities, such that the probability of the project being completed before a desired due date is maximized.

We use the following notations.

\( t \): Desired network completion time (due date);
\( T \): Completion time of the network (which is clearly a random variable);
\( R_s \): Total constrained resource, which can be allocated;
\( P \): Set of paths of the network;
\( A \): Set of activities of the network where \( |A| = n \);
\( A(N) \): Set of activities of a subnetwork, say \( N \), \( (N \) can be the network itself);

Let \( P_t(s_j, t_j) \) and \( F_t(s_j, t_j) \) represent the probability and the cumulative distribution function of the duration time of activity \( j \), respectively, provided the resource allocated to this activity is \( s_j \).

The problem is to maximize \( P(T \leq t \mid RS) \), or in fact, to maximize the probability of completion within the desired time, if the total allocated resource is \( RS \).

To allocate the constrained resource among the activities optimally, we develop an analytical approach. In this approach, some algorithms are proposed, depending on the structure of the network.

3. Analytical approach

In this section, we present an analytical approach for the optimization of constrained resource allocation in a project characterized by (a) activities in series, and (b) activities in parallel. The approach in each case is illustrated by an example.

3.1. Projects with activities in series

Suppose that a project comprises \( n \) activities in series, as in Figure 1, called activity 1, 2, \ldots, \( n \).

By applying Dynamic Programming (DP), we can allocate the constrained resource among the series of activities, optimally. At each stage of DP, the resource, which is allocated to one activity, is determined. Therefore, our DP has \( n \) stages. At each stage, the state of DP represents the amount of unused resource that can be allocated to the remaining activities.

Let \( W_j(t, R') \) denote the maximal cumulative probability of realization of project completion target time \( t \) (the completion of activity \( n \)), at time \( t \), with \( R' \) units of the resource available.
The problem of maximizing $P(T \leq t \mid RS)$ can be formulated as dynamic programming. $t_j$ is the random variable of the completion time of activity $j$, and $F_j(s_j, t_j)$ is the cumulative probability function of activity $j$ when the resource allocated to it is $s_j$.

The problem may be formally stated as follows:

Maximize $W = \prod_{j=1}^{n} F_j(s_j, t)$

Subject to: $\sum_{j=1}^{n} s_j \leq RS, \quad s_j \geq 0, \forall j.$

This is the well-known knapsack model, except that its objective function is nonlinear. Its solution is achieved via DP. Let $W_j(R')$ be the maximal value of $\prod_{j=1}^{n} F_j(s_j, t)$ when the total available resource is $R'$.

$$W_j(R') = \max \prod_{j=1}^{n} F_j(s_j, t); \forall R' \leq RS.$$ Then, the recursive equations are as follows:

$$W_n(R) = \max_{m \leq R} F_n(m, t), \quad \text{(2)}$$

$$W_j(R') = \max \{ F_j(s_j, t) W_{j+1}(R' - s_j) \}, \quad \text{(3)}$$

$0 \leq R' \leq RS, \quad j = n - 1, \ldots, 1.$

The number of stages is $n$ (the number of activities) and the state of each stage is the remaining resource, which is available for the remaining activities. Finally, $W_1(RS)$, the optimal resource allocated to each activity can be determined. In other words, $\max P(T \leq t \mid RS) = W_1(RS)$.

Example 2. Consider the project shown in Figure 4.

Activities 1 and 2 are the same as in Example 1. The probability functions of Activities 3 and 4 depend on the resource allocated to them, as shown in Table 4.

The objective is to maximize $P(T \leq t \mid RS)$ where $t = 6$ and $RS = 15$. The optimal allocation for path 1–2 is obtained by applying dynamic programming, similar to Example 1, as shown in Table 5.

By applying dynamic programming in a parallel case, we can determine the optimal resource allocation, as follows:

$$(s_1^*, s_2^*, s_3^*, s_4^*) = (3, 3, 4, 5) \quad \text{and} \quad \text{Max} P(T \leq 6 \mid RS = 15) = 0.79861.$$

4. Reducible and irreducible subnetworks/networks

Consider a subnetwork (or network), say $N$, containing several paths (from the source node to the sink node.) It is called reducible if it can be divided into two (or more) subnetworks, say $N^k, k = 1, 2, \ldots, K$, such that if $a \in N^k$, then, $a \notin N^l, l \neq k$. Otherwise, it is irreducible.

Let the network of $n$ activities and $m$ paths be represented by a matrix. In that case, the element located on the $i$th row and $j$th column of the matrix is equal to 1, if the $j$th activity is on the $i$th path, and equal to 0 otherwise. Then, this network is reducible if the set of rows can be partitioned into two (or more) subsets, such that, after partitioning, all (1) elements of each column are only in one subset.

Example 3. Consider the following networks.

The network shown in Figure 5 consists of two activities in-series and has only one path, thus, it is irreducible.

a. The network shown in Figure 5 consists of two activities in-series and has only one path, thus, it is irreducible.
Figure 6: Network (b), a parallel with 3 paths.

Figure 7: Network (c), irreducible with 3 paths.

Figure 8: Network (d), reducible with 3 paths.

The path-activity matrix of this network is as below:

\[
\begin{bmatrix}
1 & 1 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1
\end{bmatrix}
\]

b. The network shown in Figure 6 contains of 3 paths that can be divided into three subnetworks: \{1, 2\}, \{3\} and \{4\}. Thus, it is reducible.

The path-activity matrix of this network is as below:

\[
\begin{bmatrix}
1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1
\end{bmatrix}
\]

c. The network of Figure 7 has 3 paths. It is irreducible.

The path-activity matrix of this network is as below:

\[
\begin{bmatrix}
1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 1
\end{bmatrix}
\]

d. The network of Figure 8 is reducible and can be partitioned into two subnetworks: \{1, 2\} and \{3, 4, 5, 6\}.

The path-activity matrix of this network is as below:

\[
\begin{bmatrix}
1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 1
\end{bmatrix}
\]

5. Resource allocation in irreducible subnetworks/networks

In projects with parallel paths, since the completion times of paths are independent random variables, we can apply dynamic programming to obtain the optimal resource allocation. However, in irreducible subnetworks (networks), some paths are cross and random variables of paths whose completion times are dependent. Consequently, dynamic programming is not an appropriate technique to apply. Here, we develop an algorithm (Algorithm 1) that solves the problem without using dynamic programming. By selection of some suitable activities and conditioning duration, and also their amount of resources, the network is considered as an independent parallel path (conditional form of project). So, first, the conditional cumulative distribution function (CDF) of the project is obtained. Then, it is transformed into an unconditional form.

5.1. Algorithm 1

Let \( M^i \) represent the set of activities located on exactly \( i \) different paths of \( N \). The set of feasible allocation of the constrained resource is denoted by

\[
Z = \{ (s_1, \ldots, s_n) \mid \sum_{j=1}^{n} s_i \leq RS \}
\]

Step 1. Set \( \bar{W} = 1 \), \( Q = \emptyset \), \( R = \emptyset \), \( i = 1 \).

Step 2. Select one activity from \( M^i \) and call it \( c \). Let \( q \) represent the set of all pathstowhich \( c \) belongs. Let \( A(q) \) represent the set of activities of all pathsthat belongsto \( q \). If \( p \) is a member of \( q \), then \( A(p) \) represents the set of activities of \( p \). Set:

\[
\begin{align*}
Q &= Q \cup \{ c \}, \\
R &= R \cup A(q), \\
\forall i \quad M^i &= M^i - A(q),
\end{align*}
\]

Step 3: \( \bar{W} \) is the conditional CDF of project completion time. So, the CDF of project completion time will be as follows:

\[
W((s_1, \ldots, s_n), t) = \sum_{s_1} \cdots \sum_{s_n} \bigg( \bar{W} \cdot F_c \left( \min_{p \in q} \left( t - \sum_{j \in p \cap A(p)} t_j \right) \right) \bigg)
\]

Example 4. Consider the network (c) in Figure 7. The probability function of activities depends on the resources allocated to them, and shown in Table 6. The objective is to maximize \( P(T \leq t \mid RS) \), where \( t = 7 \) and \( RS = 16 \).
Step 3. Select one member of $Z'$ and call it $\mathcal{S}$. (It is obvious that it is a vector.) Then, set $Z' = Z' - \{\mathcal{S}\}$. $R(\mathcal{S})$ is the sum of the components of $\mathcal{S}$.

Step 4. For each member of $Q$ denoted by $c$, $\tau$ represents $\min_{p \in Q}(t - \sum_{j \in P} t_j)$. Then, allocate the remaining resource $(RS - R(\mathcal{S}))$ to the activities of $Q$ by applying dynamic programming as follows:

\[
\begin{align*}
U^b_m(R') &= F_m^b(\tau, R') \leq RS - R(\mathcal{S}), \\
U_k(\tau) &= F_k^b(\tau, s_k)U_{k-1}(R' - s_k), \\
\hat{V}_m(R') &= \max_{b=1} B \sum_{b=1} B \hat{P}_b U_m^b(R'), \\
\hat{V}_k(R') &= \max_{b=1} B \sum_{b=1} B \hat{P}_b U_k^b(R').
\end{align*}
\]

Step 5. If $Z' = \emptyset$, then max $P(T \leq t | RS) = \max_{z \in Z'} \hat{V}_1(RS - R(\mathcal{S}))$, otherwise, go to Step 3.

\section{Example 5}

Example 4 can be solved by Algorithm 2. In Example 4, $Q = \{2, 3, 4\}$ and $R - Q = \{1, 5\}$. Resource allocation to the activities of $R - Q$ can be done in 4 different ways. Assuming a constant duration for both activities of $R - Q$, the project network is transformed into a network with 3 parallel independent paths, as shown in Figure 9.

Then, by applying dynamic programming for the 4 cases above, the optimal resource allocation is obtained, as presented in Table 7.

\section{Resource allocation in reducible networks}

To allocate the available resource among the mutually exclusive subnetworks, we apply the dynamic programming technique. Algorithm 3 is developed for this purpose.

\subsection{Algorithm 3}

Step 1: Partition the network into some mutually exclusive $K$ subnetworks of $N^k k = 1, \ldots, K$. Suppose that $R_k(N^k)$ is the minimal amount of resource required to process all activities of subnetwork $k$, and $R_k(N^k)$ is the total available resource (RS) minus the minimal amount of resource required to process all other subnetworks. In other words, $R_k(N^k)$ is the maximum amount of resource that can be allocated to this subnetwork.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
$s_1$ & $P_1(s_1, t_1)$ & $t_1$ & $s_2$ & $P_2(s_2, t_2)$ & $t_2$ \\
\hline
3 & $\frac{1}{2}$ & 1 & 3 & $\frac{1}{2}$ & 2 \\
4 & $\frac{1}{2}$ & 2 & $\frac{1}{2}$ & 3 \\
\hline
\end{tabular}
\caption{Probability function of activities in Example 4.}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
$s_3$ & $P_3(s_3, t_3)$ & $t_3$ & $s_4$ & $P_4(s_4, t_4)$ & $t_4$ \\
\hline
2 & $\frac{1}{2}$ & 2 & 4 & $\frac{1}{2}$ & 5 \\
3 & $\frac{1}{2}$ & 3 & $\frac{1}{2}$ & 6 \\
\hline
\end{tabular}
\caption{Optimal resource allocation in Example 5.}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
$s_5$ & $P_5(s_5, t_5)$ & $t_5$ \\
\hline
2 & $\frac{1}{2}$ & 2 \\
3 & $\frac{1}{2}$ & 3 \\
\hline
\end{tabular}
\caption{MaxF(16, 7) = MaxP(T ≤ 7 | RS = 16) = 0.6375}
\end{table}
It is obvious the system is infeasible if $\tilde{R}(N^k) > \tilde{R}(N^k)$. On the other hand, if $\tilde{R}(N^k)$ is greater than the maximum amount of resource needed for all activities of $N$, then this extra resource cannot be consumed.

Step 2. Suppose that $\tilde{R}(N^k)$ represents the admissible resource of subnetwork $N^k$. By applying Algorithm 2, the optimal allocation for each activity, and for every value of the admissible resource of subnetwork $N^k$, $(\tilde{R}(N^k) \leq \tilde{R}(N^k) \leq \tilde{R}(N^k))$ is determined.

Step 3. Allocate RS to the network, such that $P(T \leq t \mid RS)$ be maximized. This is performed by the dynamic programming technique, as discussed before.

Notice that in the solution of Example 2, Step 1 of Algorithm 3 results in partitioning the network into 3 subnetworks. Step 2 of Algorithm 3 has to be performed for all subnetworks. For instance, Step 2 is performed for one of the subnetworks comprising Activities 1 and 2, and the result is presented in Table 5. Then, by applying dynamic programming in Step 3, the optimal resource allocation is obtained.

**Example 6.** Consider network (d) in Figure 8. The probability function of activities depends on the resource allocated to them, which is shown in Table 8. The objective is to maximize $P(T \leq t \mid RS)$, where $t = 6$ and $RS = 20$.

Implementing Algorithm 3 results in optimal resource allocation, as Table 9.

### Table 8: Probability function of activities in Example 5.

<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$P_1(s_1, t_1)$</th>
<th>$t_1$</th>
<th>$s_2$</th>
<th>$P_2(s_2, t_2)$</th>
<th>$t_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$\frac{1}{3}$</td>
<td>3</td>
<td>$\frac{1}{2}$</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$\frac{1}{3}$</td>
<td>3</td>
<td>$\frac{1}{2}$</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$\frac{1}{3}$</td>
<td>3</td>
<td>$\frac{1}{2}$</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

### Table 9: Optimal resource allocation in Example 6.

<table>
<thead>
<tr>
<th>$s^*_1$</th>
<th>$s^*_2$</th>
<th>$s^*_3$</th>
<th>$t_1$</th>
<th>$s^*_4$</th>
<th>$s^*_5$</th>
<th>MaxP($T \leq 6 \mid RS = 20$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

7. Conclusions and recommendations for future studies

In this paper, we developed an analytical approach to allocate a limited resource to the activities of a PERT-Type Network (PTN) to maximize the probability of completion time within some desired period. This approach gains satisfactory results for moderate size networks.

For further research, the following extensions are recommended.

1. Allocation of constrained resources to the activities could be extended to continuous values.
2. Generalization of the problem for the case of more than one constrained resource.
3. A combination of extensions 1 and 2.
4. We can change the objective function to consider the cost of resource utilization, together with a reward for early completion, and a penalty for late completion, of the project.

### References


