Minimizing the Project Cost with Generalized Precedence Relations

Zhi-Xiong Su1*, Han-Ying Wei1, Xue-Min Yu2

1 Business Administration College, Nanchang Institute of Technology, Nanchang, Jiangxi, China.
2 School of Government, Beijing Normal University, Beijing, China.

* Corresponding author. Tel.: +86-15279190886; email: suzhixiongbaner@126.com
Manuscript submitted October 8, 2015; accepted December 10, 2015.
doi: 10.17706/jsw.11.2.162-181

Abstract: Minimizing the project cost is a task of project scheduling, and usually is a starting point in the optimization about cost, for example the time-cost tradeoff is to compress the project duration from the one with minimum cost. Project cost can be minimized by letting all activities choose their minimum cost durations only when strict precedence relations exist between activities. But if generalized precedence relations (GPRs) exist between activities, letting all activities choose their minimum cost durations may not satisfy the given precedence relationships and result in a unfeasible project. In minimizing the project cost with GPRs, we transformed the mathematical programming model into two equivalent special models: a minimum cost - maximum flow model and a transportation model with balanced supply and demand. The two special models can be solved by using any current efficient algorithms.

Key words: Project scheduling, generalized precedence relations, project cost, duality.

1. Introduction

Minimizing the project cost is regarded as a main objective in production planning, and furthermore an ideal optimization process of project scheduling often starts with the minimum project cost. The most representative problem is time-cost trade-off problem [1], and a classic algorithm is to obtain the project duration at a minimum cost and shorten it with a minimum compressed cost [2]. Clearly, minimizing the project cost is directly related to the results of the optimization. If we cannot obtain the minimum total cost, the optimality and rationality of the solution will not be ensured and the minimum cost curve of the project duration will be distorted.

The difficulties of the project minimum cost problems with different precedence relations are different. For a classic one, the start-time of an activity $j$ is no earlier than the finish-time of its immediate predecessor $i$, and we can minimize the project cost by letting all activities choose their minimum cost durations. But in practice, cases of generalized precedence relations (GPRs) [3], [4] between activities are more common.

For minimizing the project cost with GPRs, if all activities are still allowed to choose their minimum cost durations, the rigid precedence relationships may be not met, and thus the project may be unfeasible. Furthermore, how to arrange an activity's duration depends on most other activities' durations and precedence relations. For example, there are three activities, $i$, $j$ and $k$, in Fig. 1, and their minimum cost durations are 10, 10 and 40, respectively. The precedence relations between them are such that: (1) the start-time of activity $i$ is no later than 2 after the start-time of activity $k$; (2) the start-time of $j$ is no later...
than 3 after the finish-time of \(i\); (3) the finish-time of \(k\) is no later than 5 after the finish-time of \(j\). If letting the three activities choose their minimum cost durations, then the finish-time of activity \(k\) will have to be later than 5 after the finish-time of activity \(j\). It offends against the precedence relations (3) and causes an unfeasible project (see Fig. 1). We can only adjust activity durations in order to satisfy these fixed relationships. In Fig. 1, we show that the project can made feasible by shortening the duration of activity \(k\), or prolonging the duration of \(i\) or \(j\). But the more difficult task is to minimize the project cost under the precondition of guaranteeing that the project will be feasible.

![Fig. 1. Times of activities \(i\), \(j\) and \(k\).](image)

Fig. 1 only shows simple GPRs, and we may minimize the project cost by directly adjusting activity durations. But when activities and precedence relationships are numerous, it is very difficult to minimize the project cost by directly adjusting durations. The main reason is that the precedence relations between activities are intricate, and satisfying some relations may result in others that cannot be met. In this paper, we studied the representation of the GPRs, viz. the activity network under GPRs, and designed an algorithm to minimize the project cost with GPRs.

The activity network under GPRs is an effective model to represent all types of precedence relationships between activities. Roy [3] and Elmaghraby [4] introduced the concept of the GPRs and Kerbosch and Schell [5] first proposed a simple and applicable model to represent them based on Roy's concept. The works of Wiest [6] and Elmaghraby and Kamburovski [7] can be regarded as a milestone in the development of the activity network under GPRs. For Wiest [6], he found abnormal critical activity under some GPRs, in which the project duration will be prolonged if the duration of the activity is shortened, and vice versa. This overturned the traditional view of critical activity and opened up a new area of activity network under GPRs investigation. For Elmaghraby and Kamburovski [7], firstly, they cleverly represented all precedence relations in a network by adding reverse arcs with negative lengths, and proposed a perfect activity network under GPRs; secondly, they further studied the anomalies under GPRs, and described them as two manifestations that, the one refers to the reduction (increase) in project completion as a consequence of prolonging (shortening) an activity, and the other one occurs when diminishing the duration of an activity results in infeasibility of the activity network; in addition, they studied the time-cost tradeoff problem with GPRs and transformed it as a special case of the uncapacitated minimum cost flow problem. On the basis of the above works, Qi and Su [8] and Su et al. [9], [10] focused on noncritical activities and time floats under GPRs, and discovered new anomalies such as the invariability and increase in time float following consumption. In these works, they have pointed out that there was a project feasibility problem, which may be caused by GPRs.

A number of scholars have researched a variety of problems with GPRs, and the most important of these, with respect to the current paper, are those dealing with project scheduling problems [11]–[17]. They mainly emphasized resource conflicts and minimized the project duration under conditions of constrained resources, rather than minimized the project cost. The time-cost tradeoff problem with GPRs is an important project scheduling problem, and Elmaghraby and Kamburovski [7] have been instrumental in presenting the problem. They transformed the problem into a minimum cost flow problem and devised a
minimum flow - maximum cutset approach. Kaveh et al. [18] provided four solution procedures for a multi-mode time-cost-quality trade-off problem with GPRs, which include the classical epsilon-constraint, the efficient epsilon-constraint method, dynamic self-adaptive multi-objective particle swarm optimization (DSAMOPSO), and the multi-start partial bound enumeration algorithm. Others have proposed that the optimal time-cost curve and the minimal schedule cost for the time-cost tradeoff problem with GPRs can be obtained by using linear/integer programming [18]–[21].

Minimizing the project cost with GPRs has close similarity to the time-cost tradeoff problem with GPRs, and the above - mentioned research could be used to solve the problem. For example, Elmaghraby and Kamburowski [7] found the minimum project duration (labeled as \( \lambda \)) and the cheapest project schedule for \( \lambda = \lambda^* \), and then increased the project duration until there was no further decrease in the project cost. Thus, the project duration was a minimum cost one. The procedure is effective in determining a project cost curve, but is long and wasteful in that it only computes the minimum project cost. Heuristic algorithms and linear/integer programming also might be used to compute the minimum project cost problem with GPRs [18]–[21]. But the former cannot guarantee an optimal solution, and the latter would be wasteful in terms of computing effort and time [2], [22]. Therefore, the current approaches for the time-cost tradeoff problem with GPRs are not normally subjected to minimizing the project cost with GPRs.

In this paper, our target is to minimize the project cost with GPRs based on the linear programming and duality theory. Elmaghraby and Kamburowski [7] created a mathematical programming model of the time-cost tradeoff problem with GPRs. By removing the constraint of project duration, we transformed the model into a minimum project cost model with GPRs. And base on duality theory, we transformed the model into two equivalent special models, which were a minimum cost - maximum flow model, and a transportation model with balanced supply and demand. Thus, we could compute optimal solutions for the two models by using current efficient algorithms, and minimize the project cost with GPRs based on the primal-dual relation.

2. The Activity Network Under GPRs

GPRs contain minimum and maximum time lags, as shown in Table 1. The activity network under GPRs [7] is a current representation of the GPRs, and its characteristics are as follows:

<table>
<thead>
<tr>
<th>Relations</th>
<th>Minimum time lags</th>
<th>Maximum time lags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Begin-to-Start (BS)</td>
<td>The start-time of activity ( i ) is no earlier than ( d ) after the begin-time of project.</td>
<td>The start-time of activity ( i ) is no later than ( d ) after the begin-time of project.</td>
</tr>
<tr>
<td>Begin-to-Finish (BF)</td>
<td>The finish-time of activity ( i ) is no earlier than ( d ) after the begin-time of project.</td>
<td>The finish-time of activity ( i ) is no later than ( d ) after the begin-time of project.</td>
</tr>
<tr>
<td>Start-to-End (SE)</td>
<td>The end-time of project is no earlier than ( d ) after the start-time of activity ( i ).</td>
<td>The end-time of project is no later than ( d ) after the start-time of activity ( i ).</td>
</tr>
<tr>
<td>Finish-to-End (FE)</td>
<td>The end-time of project is no earlier than ( d ) after the finish-time of activity ( i ).</td>
<td>The end-time of project is no later than ( d ) after the finish-time of activity ( i ).</td>
</tr>
<tr>
<td>Start-to-Start (SS)</td>
<td>The start-time of activity ( j ) is no earlier than ( d ) after the start-time of activity ( i ).</td>
<td>The start-time of activity ( j ) is no later than ( d ) after the start-time of activity ( i ).</td>
</tr>
<tr>
<td>Start-to-Finish (SF)</td>
<td>The finish-time of activity ( j ) is no earlier than ( d ) after the start-time of activity ( i ).</td>
<td>The finish-time of activity ( j ) is no later than ( d ) after the start-time of activity ( i ).</td>
</tr>
<tr>
<td>Finish-to-Start (FS)</td>
<td>The start-time of activity ( j ) is no earlier than ( d ) after the finish-time of activity ( i ).</td>
<td>The start-time of activity ( j ) is no later than ( d ) after the finish-time of activity ( i ).</td>
</tr>
<tr>
<td>Finish-to-Finish (FF)</td>
<td>The finish-time of activity ( j ) is no earlier than ( d ) after the finish-time of activity ( i ).</td>
<td>The finish-time of activity ( j ) is no later than ( d ) after the finish-time of activity ( i ).</td>
</tr>
</tbody>
</table>
1) An activity \( k \) is represented as two arcs \((2k-1, 2k)\) and \((2k, 2k+1)\) with lengths \( d_{2k-1, 2k} = x_k \) and \( d_{2k, 2k+1} = -x_k \), and \( x_k \) indicates the activity duration.

2) A minimum time lag is represented as a forward arc with length \( d = r \), a maximum time lag is represented as a reverse arc with length \( d = -r \), and \( r \) indicates the value of the time lag.

3) If there are \( n \) activities, then the beginning node of the network is \((0)\) and the end node is \((2n+1)\).

4) The duration of the activity \( k \) is \( x_k = t_{2k} - t_{2k-1} \), and \( t_i \) indicates the time of a node \( i \).

Fig. 2 shows a simple activity network under GPRs example, and the precedence relations between the three activities are listed in Table 2 (\( s_i \) and \( f_i \) respectively indicate the start and finish times of an activity \( i \)).

![Activity Network Diagram]

**Fig. 2. An activity network under GPRs.**

<table>
<thead>
<tr>
<th>Activities pair</th>
<th>Kind of relation</th>
<th>Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project and activity 1</td>
<td>BS</td>
<td>( 0 \leq s_1 )</td>
</tr>
<tr>
<td>Project and activity 2</td>
<td>BS</td>
<td>( 0 \leq s_2 )</td>
</tr>
<tr>
<td>Activities 1 and 2</td>
<td>FS</td>
<td>( f_1 + 4 \leq s_2 )</td>
</tr>
<tr>
<td>Activities 1 and 3</td>
<td>SF</td>
<td>( s_1 + 8 \leq f_2 )</td>
</tr>
<tr>
<td>Activities 2 and 3</td>
<td>SS</td>
<td>( s_1 + 1 \leq s_2 )</td>
</tr>
<tr>
<td>Activity 5 and Project</td>
<td>FE</td>
<td>( f_2 \leq f_3 + 3 )</td>
</tr>
<tr>
<td>Activity 7 and Project</td>
<td>FE</td>
<td>( f_4 + 1 \leq T )</td>
</tr>
</tbody>
</table>

### 3. The Minimum Project Cost Model with GPRs

Assume there are \( K \) activities in an activity network under GPRs \( G \) (represents a project). For an activity \( k \), its duration \( x_k \) is bound from above and below as follows: \( x_k \in [l_k, u_k] \), and its cost is a function \( g_k(x_k) \) of \( x_k \). Then the project cost is \( \sum_{i=1}^{K} g_i(x_i) \), and the objective function of the minimum project cost model is
\[
\min z = \sum_{i=1}^{K} g_i(x_i) \tag{1}
\]

Let \(\overline{A}\) indicates a set of arcs that represent the precedence relationships in the network \(G\). According to Section 2, \(t_j \geq t_i + d_{i,j}\) and \(x_i = t_{2k} - t_{2k-1}\), the minimum cost model is:

\[
\min z = \sum_{i=1}^{K} g_i(t_{2k} - t_{2k-1}) \tag{2}
\]

subject to

\[
t_j - t_i \geq d_{i,j}, \quad \forall (i,j) \in \overline{A} \tag{3}
\]

\[
l_k \leq t_{2k} - t_{2k-1} \leq u_k \tag{4}
\]

For the activity \(k\), its function cost may be arbitrary. For the sake of simplicity, we have assumed that its cost function \(g_i(x_i)\) is a piecewise-linear function having \(Q\) linear segments and \(Q+1\) breakpoints, marked as \(l_k = d_i^1 < d_i^2 < \cdots < d_i^Q = u_k\). Under normal circumstances, the cost might be lower when the duration is longer, hence the slope of the time-cost function of a segment of \(g_i(x_i)\) is negative. Generally speaking, the slopes of most segments of \(g_i(x_i)\) are negative, and it is not convenient to compute the corresponding model. In order to avoid the obstacle, we multiply all slopes of \(g_i(x_i)\) by \(-1\), that is, denote the slope (multiplied by \(-1\)) of \(g_i(x_i)\) in the interval \([d_i^q, d_i^{q+1}]\) by \(a_i^q\) for \(q = 0,1,\ldots,Q-1\), and then switch the minimize objective function \(\min z\) (Equation (2)) to a maximize objective function \(\min z'\) (Equation (5)). When \(x_i \in [d_i^q, d_i^{q+1}]\), then \(g_i(x_i)\) is decreasing or increasing linearly. A sample activity cost function having nine linear segments is shown in Fig. 3.

Since the functions \(g_i(x_i)\) are piecewise-linear, the problem is easily transformed into a linear program model as follows: introduce the variables \(y_i^q\) for \(q = 1,2,\ldots,Q\), one for each segment of \(g_i\), and replace \(x_i = t_{2k} - t_{2k-1}\) by \(\sum_{q=1}^{Q} y_i^q\). Setting the variables \(y_i^q\) is to adjust the representation of the linear programming for next conveniently creating the dual model of the primal problem. The model (Equations (2)-(4)) then becomes:
\[ \max z' = \sum_{i=1}^{K} \sum_{q=1}^{Q} y_i^q \] (5)

subject to

\[ t_j - t_i \geq d_{ij}, \quad \forall (i, j) \in \bar{A} \] (6)

\[ \sum_{q=1}^{Q} y_i^q = t_{2k} - t_{2k-1} \] (7)

\[ d_0^i \leq y_i^q \leq d_1^i, \quad k = 1, 2, \ldots, K \] (8)

\[ 0 \leq y_i^q \leq d_k^i - d_{k-1}^i, \quad k = 1, 2, \ldots, K \quad \text{and} \quad q = 2, 3, \ldots, Q \] (9)

Observe that any optimal solution of the above model satisfies the following conditions [7]:

1) If \( y_i^q \leq d_1^i \), then \( y_i^q = 0, \quad q = 2, 3, \ldots, Q \);

2) If \( y_i^q \leq d_k^i - d_{k-1}^i, \quad q \neq 1, Q \), then \( y_i^q = 0, \quad s = q + 1, \quad q + 2, \ldots, Q \);

3) If \( y_i^q > 0, \quad q > 1 \), then \( y_i^q = d_k^i \), and \( y_i^q \leq d_k^i - d_1^i, \quad s = 1, 2, \ldots, q - 1 \).

There are two kinds of variables, \( y_i^q \) and \( t_i \), in the model (Equations (5)–(9)), which might make the model difficult to solve, but both of them can represent times and durations, and the variable \( y_i^q \) can be represented by \( t_i \). Thus, we can replace \( y_i^q \) by \( t_i \) so that the model has only one kind of variable \( t_i \), and this measure will help make the primal-dual transformation (see Section 5). According to the model, and the conditions which the model satisfies, for the activity \( k \), we introduce the variables \( t_i, t_{k_1}, \ldots, t_{k_{Q_2}} \), defined by \( t_k = y_i^q + t_{k_{Q_2}} \), \( t_k = y_i^q + t_{k_{Q_2}} \) for \( q = 2, 3, \ldots, Q - 1 \), and \( t_{k_2} = y_i^q + t_{k_{Q_2}} \), which indicate the time points of \( k, k_1, \ldots, k_{Q_2} \), like 2k-1 and 2k. Then we rewrite Equations (5)–(9) as:

\[ \max z' = \sum_{i=1}^{K} \left( a_i^q \left( t_i - t_{2k-1} \right) + \sum_{q=1}^{Q_2} a_i^q \left( t_i - t_{k_{Q_2}} \right) + a_i^q \left( t_{2k} - t_{k_{Q_2}} \right) \right) \] (10)

subject to

\[ t_j - t_i \geq d_{ij}, \quad \forall (i, j) \in \bar{A} \] (11)

\[ d_k^0 \leq t_k - t_{2k-1} \leq d_k^1 \] (12)

\[ 0 \leq t_k - t_{k_{Q_2}} \leq d_k^q - d_{k-1}^q, \quad q = 2, 3, \ldots, Q - 1 \] (13)

\[ 0 \leq t_{2k} - t_{k_{Q_2}} \leq d_k^q - d_{k-1}^q \] (14)

Base on duality theory, we can transform the above model into two equivalent and special models, a minimum cost - maximum flow model, and a transportation model with balanced supply and demand.

4. Equivalent Model 1 — A minimum Cost-maximum Flow Model

Steps of transforming the minimum project cost model with GPRs into a special minimum cost - maximum flow model are as follows:

**Step 1**: Transform the network \( G \) into \( G' \).

Step 1-1: For an activity \( k \), add nodes \( \left( k_q \right) \) for \( q = 1, 2, \ldots, Q - 1 \), and transform the forward arc \((2k-1, 2k)\) into forward arcs \((2k-1, k_1), (k_1, k_2), \ldots, (k_{Q_2}, k_{Q_2})\), \((k_{Q_2}, 2k)\), and transform the reverse arc
Let each above arc \((i, j)\) above have weight \(w_{i,j}\) and \(c_{i,j}\), such that \(w_{2k-1,i} = -d_i^0\), \(w_{k,k} = w_{k,k} = \ldots = w_{k,k+2k} = 0\), \(w_{2k,k+2k} = d_k^0 - d_k^{q-1}\), \(w_{k+2k,k+2k} = d_k^{q-1} - d_k^{q-2}\), \ldots, \(w_{k,k} = d_k^2 - d_k^1\), \(w_{k,k+2k} = d_k^0\) and \(\forall c_{i,j} = +\infty\). (The Steps are shown in Fig. 4.)

\[
\min \varphi = \sum_{(i,j) \in G'} w_{i,j} f_{i,j} \tag{15}
\]

\[
\max \varphi' = \nu(f) \tag{16}
\]

subject to

\[
\sum_j f_{i,j} - \sum_j f_{j,i} = \begin{cases} \nu(f), & i = s \\ 0, & i \neq s, t \\ -\nu(f), & i = t \end{cases} \tag{17}
\]

\[
0 \leq f_{i,j} \leq c_{i,j} \tag{18}
\]

This model is a minimum cost - maximum flow model, which is a dual model of the primal model (Equations (10)–(14)). The model is a special case in that, except for arcs connecting the nodes \((s)\) or \((t)\), the capacities of the other arcs are unlimited. There are close relationships between the two models:

1) If the maximum flow of the model is \(\max \nu(f) = \sum_i c_{i,j} = \sum_i f_{i,j}\), then based on the optimal solution \(f_{i,j}^*\) and the primal-dual relationship, we can obtain optimal solutions \(\nu^*\) to the primal minimum cost.
model. The optimal duration of each activity \( k \) is \( x_k^* = t_{2k}^* - t_{2k-1}^* \), and the project duration with minimum cost is \( T^* = t_{2K+1}^* - t_0^* \).

2) If the maximum flow of the model is \( \max \nu(f) < \sum_j c_{x,j} = \sum_j c_{j,t} \), the primal minimum cost model has no optimal solution. This means that some precedence relations cannot be satisfied, and the project is unfeasible.

The proof for creating the equivalent model 1 is given in Appendix A.

5. Equivalent Model 2 — A Transportation Model with Balanced Supply and Demand

Steps of transforming the minimum cost model of the activity network under GPRs into a transportation model with balanced supply and demand are as follows:

**Step 1:** Transform the network \( G \) into \( G' \).

**Step 1-1:** It is similar to **Step 1-1** of the algorithm in Section 4.

**Step 1-2:** For an arc \( (i, j) \in A \) in the network \( G \), let it have a weight \( w_{i,j} = d_{i,j} \).

**Step 2:** Create the transportation model with balanced supply and demand of the network \( G' \).

**Step 2-1:** In the network \( G' \), for \( k = 1,2, \ldots, K \),

- for a node \( (2k-1) \), if \( a_k^i > 0 \), then the node is a demand-side with demand \( D_{2k-1} = a_k^i \); and if \( a_k^i < 0 \), then the node is a supply-side with supply \( S_{2k-1} = -a_k^i \);

- for a node \( (k) \), \( q = 1,2, \ldots, Q-1 \), if \( a_k^q - a_k^{q+1} > 0 \), then the node is a supply-side with supply \( S_{k} = a_k^q - a_k^{q+1} \); and if \( a_k^q - a_k^{q+1} < 0 \), then the node is a demand-side with demand \( D_{k} = a_k^q - a_k^{q+1} \);

- for a node \( (2r) \), if \( a_k^r > 0 \), then the node is a supply-side with supply \( S_{2r} = a_k^r \); and if \( a_k^r < 0 \), then the node is a demand-side with demand \( D_{2r} = -a_k^r \).

**Step 2-2:** Assume there are \( U \) supply-sides and \( V \) demand-sides in the network \( G' \), search the shortest path from a supply-side \( (u) \) to a demand-side \( (v) \) (marked as \( \mu_{u,v}^{\min} \)) and compute its length \( e_{u,v} \), viz

\[
e_{u,v} = \sum_{(i, j) \in \mu_{u,v}^{\min}} w_{i,j} \quad \text{(19)}
\]

**Step 2-3:** Let \( w_{i,j} \) indicate unit fare of an arc \( (i, j) \), and \( e_{p,q} \) and \( h_{p,q} \) indicate unit fare and transport volume, respectively, from a supply-side \( (u) \) to a demand-side \( (v) \). Our transportation model then is:

\[
\min \varphi = \sum_{u=1}^{U} \sum_{i=1}^{V} e_{u,v} h_{u,v} \quad \text{(20)}
\]

subject to

\[
\sum_{u=1}^{U} h_{u,v} = S_v \quad \text{(21)}
\]

\[
\sum_{i=1}^{V} h_{u,v} = D_i \quad \text{(22)}
\]

\[
h_{u,v} \geq 0 \quad \text{(23)}
\]

The model is a transportation model with balanced supply and demand. According to the transport volume \( h_{u,v} \) from a supply-side \( (u) \) to a demand-side \( (v) \), we define the transport volume \( g_{i,j} \) of an arc
In all paths with minimum fares from the supply-sides to the demand-sides, if the paths $\mu_{u_i \rightarrow v_j}$, $\mu_{u_j \rightarrow v_i}$, $\ldots$, $\mu_{u_n \rightarrow v_i}$ pass the arc $(i, j)$, the transport volume $g_{i,j}$ of the arc is the sum of the transport volumes of these paths, viz.

$$g_{i,j} = \sum_{k=1}^{n} h_{u_k \rightarrow v_k}$$  \hspace{1cm} (24)

The transportation model with balanced supply and demand is a dual one of the primal model (Equations (10)–(14)). According to the optimal solution $h^*_{u,v}$ of the transportation model, we can obtain the optimal transport volume $g^*_{i,j}$ of each arc $(i, j)$ in the network $G'$. Because $g^*_{i,j}$ is the dual variable of the primal model, we can obtain the optimal solution $t^*_i$ of the primal model based on the primal-dual relationship. The optimal duration of each activity $k$ is $x^*_k = t^*_k - t^*_{k-1}$, and the project duration with minimum cost is $T^* = t^*_{K+1} - t^*_0$.

The proof for the above dual model 1 is given in Appendix B.

![Fig. 5. Piecewise-linear duration-cost functions of the activities 1, 2 and 3.](image1)

![Fig. 6. Cost flow network $G'$.](image2)
6. Illustration

For the project in Fig. 2, assume the cost functions curves of the three activities are shown in Fig. 5. How to set the optimal duration for each activity to minimize the project cost?

We apply the two approaches to solve the problem.

Approach 1

Step 1: Transform the network in Fig. 2 into the cost flow network $G^*$. By using Step 1-1–Step 1-3 of the algorithm in Section 4, we obtain the cost flow network $G^*$ as in Fig. 6.

Step 2: Compute the minimum cost - maximum flow from the beginning node $(s)$ to the end node $(t)$ in $G^*$ (use current algorithms). The maximum flow is

$$v(f) = \sum_{(j) \in A^*} c_{i,j} = \sum_{(j) \in A^*} c_{i,j} = 6.75$$

Therefore there are feasible solutions for the primal minimum project cost problem. The optimal flows of all arcs are shown in Table 3.

Table 3. Optimal arc flows in Fig. 8

<table>
<thead>
<tr>
<th>Arc</th>
<th>Optimal flow</th>
<th>Arc</th>
<th>Optimal flow</th>
<th>Arc</th>
<th>Optimal flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0,1)</td>
<td>0</td>
<td>(2,3)</td>
<td>0.25</td>
<td>(3,5)</td>
<td>1.5</td>
</tr>
<tr>
<td>(0,5)</td>
<td>0</td>
<td>(3,2)</td>
<td>0</td>
<td>(3,3)</td>
<td>0</td>
</tr>
<tr>
<td>(1,5)</td>
<td>0</td>
<td>(2,3)</td>
<td>0.75</td>
<td>(3,5)</td>
<td>1</td>
</tr>
<tr>
<td>(1,1)</td>
<td>0</td>
<td>(2,2)</td>
<td>0.75</td>
<td>(3,3)</td>
<td>0</td>
</tr>
<tr>
<td>(1,1)</td>
<td>2</td>
<td>(2,2)</td>
<td>0</td>
<td>(3,3)</td>
<td>0.5</td>
</tr>
<tr>
<td>(1,1)</td>
<td>0</td>
<td>(2,4)</td>
<td>1</td>
<td>(3,6)</td>
<td>0</td>
</tr>
<tr>
<td>(1,1)</td>
<td>1</td>
<td>(4,2)</td>
<td>0</td>
<td>(6,3)</td>
<td>0</td>
</tr>
<tr>
<td>(1,1)</td>
<td>0.25</td>
<td>(1,4)</td>
<td>0</td>
<td>(6,2)</td>
<td>0.75</td>
</tr>
<tr>
<td>(1,1)</td>
<td>0.75</td>
<td>(4,6)</td>
<td>1.25</td>
<td>(6,7)</td>
<td>0</td>
</tr>
<tr>
<td>(1,2)</td>
<td>1.25</td>
<td>(4,7)</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(2,1)</td>
<td>0.5</td>
<td>(5,3)</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Based on the optimal flow of each arc in $G^*$ and the primal-dual relationship, we obtain the optimal time of each node in the network in Fig. 2, that $t'_0 = 0$, $t'_1 = 0$, $t'_2 = 6$, $t'_3 = 10$, $t'_4 = 16$, $t'_5 = 4$, $t'_6 = 9$, $t'_7 = 17$. The optimal durations of the activities 1, 2 and 3 are...
The minimum project cost is 3.5, and the corresponding project duration is \( T^* = t^*_i - t^*_i = 17 \).

**Approach 2**

**Step 1:** Transform the network in Fig. 2 into the network \( G' \).

By using Step 1-1 – Step 1-2 of the algorithm in Section 5, the network \( G' \) as in Fig. 7.

**Step 2:** Based on \( G' \), create the transportation model with balanced supply and demand.

By using Step 2-1 of the algorithm in Section 5, we obtain the supply-side, demand-side, supply and demand as in Table 4. Then apply Step 2-2 to set length \( w_{i,j} \) of each arc, and compute the length \( e_{u,v} \) of the shortest path connecting a supply-side \( u \) and a demand-side \( v \) as follows:

\[
\mu_{1_{1} \rightarrow 1} = (1_{1} \rightarrow 1_{1}), \quad e_{1_{1}, 1} = 2;
\]

\[
\mu_{1_{2} \rightarrow 2} = (1_{2} \rightarrow 1_{1} \rightarrow 1_{2} \rightarrow 2), \quad e_{1_{2}, 2} = 0;
\]

\[
\mu_{1_{3} \rightarrow 3} = (1_{3} \rightarrow 6), \quad e_{1_{3}, 3} = 0.
\]

Let \( e_{u,v} \) as the minimum unit fare from a supply-side \( u \) to a demand-side \( v \), as in Table 5.

### Table 4. Supply-Side, Demand-Side, Supply and Demand

<table>
<thead>
<tr>
<th>Supply-side</th>
<th>Demand-side</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>Supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>1.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.25</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.5</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

| Demand      | 2 | 0.75 | 2 | 1.5 | 0.5 |

### Table 5. The Minimum unit Fare from a Supply-Side to a Demand-side

<table>
<thead>
<tr>
<th>Supply-side</th>
<th>Demand-side</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>-4</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>0</td>
<td>-4</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>6</td>
<td>0</td>
<td>-4</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>18</td>
<td>10</td>
<td>5</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>18</td>
<td>10</td>
<td>6</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>18</td>
<td>6</td>
<td>10</td>
<td>14</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>3</td>
<td>-1</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>3</td>
<td>-1</td>
<td>3</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>3</td>
<td>-1</td>
<td>5</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**Step 3:** Compute the optimal transport volume from a supply-side to a demand-side use the (use the table- manipulation method), as in Table 6. And According to Equation (24), compute the optimal transport volume of each arc in \( G' \), as in Table 7.
Based on the optimal transport volume of each arc in \( G' \) and the primal-dual relationship, we obtain the optimal time of each node in the activity network under GPRs in Fig. 2, that \( t_0^* = 0 \), \( t_1^* = 0 \), \( t_2^* = 6 \), \( t_3^* = 10 \), \( t_4^* = 16 \), \( t_5^* = 4 \), \( t_6^* = 9 \) and \( t_7^* = 17 \). The optimal durations of the activities 1, 2 and 3 are
\[
x_1^* = t_2^* - t_1^* = 6, \quad x_2^* = t_3^* - t_2^* = 6, \quad x_3^* = t_4^* - t_3^* = 5
\]
The minimum project cost is 3.5, and the corresponding project duration is \( T^* = t_7^* - t_0^* = 17 \).

<p>| Table 6. Supply-Side, Demand-Side, Supply and Demand |</p>
<table>
<thead>
<tr>
<th>Supply-side</th>
<th>Transport volume</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>Supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>0.25</td>
<td>0.25</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0.25</td>
<td>1</td>
<td>1.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>0.25</td>
<td>0.75</td>
<td>0.5</td>
<td>1.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>0.25</td>
<td></td>
<td>0.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.25</td>
<td></td>
<td>0.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td>0.5</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>0.5</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Demand</td>
<td></td>
<td>2</td>
<td>0.75</td>
<td>2</td>
<td>1.5</td>
<td>0.5</td>
<td></td>
</tr>
</tbody>
</table>

<p>| Table 7. The Optimal Transport Volume of Each Arc in the Network ( G' ) |</p>
<table>
<thead>
<tr>
<th>Arc</th>
<th>Transport volume</th>
<th>Arc</th>
<th>Transport volume</th>
<th>Arc</th>
<th>Transport volume</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 1)</td>
<td>0</td>
<td>(2, 3)</td>
<td>0.25</td>
<td>(3, 5)</td>
<td>1.5</td>
</tr>
<tr>
<td>(0, 5)</td>
<td>0</td>
<td>(3, 2)</td>
<td>0</td>
<td>(3, 3)</td>
<td>0</td>
</tr>
<tr>
<td>(1, 5)</td>
<td>0</td>
<td>(2, 3)</td>
<td>0.75</td>
<td>(3, 3)</td>
<td>1</td>
</tr>
<tr>
<td>(1, 11)</td>
<td>0</td>
<td>(2, 2)</td>
<td>0.75</td>
<td>(3, 3)</td>
<td>0</td>
</tr>
<tr>
<td>(1, 1)</td>
<td>2</td>
<td>(2, 2)</td>
<td>1</td>
<td>(3, 3)</td>
<td>0.5</td>
</tr>
<tr>
<td>(1, 12)</td>
<td>0</td>
<td>(2, 4)</td>
<td>1</td>
<td>(3, 6)</td>
<td>0</td>
</tr>
<tr>
<td>(1, 13)</td>
<td>0.25</td>
<td>(4, 2)</td>
<td>0</td>
<td>(6, 3)</td>
<td>0</td>
</tr>
<tr>
<td>(1, 2)</td>
<td>0.75</td>
<td>(4, 6)</td>
<td>1.25</td>
<td>(6, 7)</td>
<td>0</td>
</tr>
<tr>
<td>(1, 3)</td>
<td>1.25</td>
<td>(4, 7)</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(2, 13)</td>
<td>0.5</td>
<td>(5, 3)</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7. Conclusion

For minimizing the project cost with GPRs, empirical method of letting all activities choose their minimum cost durations may cause failure to satisfy given precedence relations between activities. Current algorithms also may not perfectly apply to the problem. In order to solve this problem, we apply the mathematical programming and duality theory. We transformed the mathematical model of the problem into two equivalent and special models by using the duality theory. These were a minimum cost - maximum flow model and a transportation model with balanced supply and demand. The two models can be efficiently solved using current algorithms. Based on the optimal solutions of the two models and the primal-dual relationship, we minimized the project cost with GPRs and obtained the corresponding project duration and activity durations. Minimizing the project cost with GPRs is a main objective of project management with GPRs, and moreover it may help to tackling project scheduling with GPRs, such as the time-cost tradeoff problem with GPRs.

Appendix A. Proof for Creating the Equivalent Model 1

For the primal model (Equations (10)–(14)), we set dual variables as follows:
\[ \max z' = \sum_{k=1}^{K} d_k' (t_k - t_{2k-1}) + \sum_{q=2}^{Q_2} a_{q}^1 (t_q - t_{2q-1}) + a_{q}^2 (t_{2q-1} - t_{2q}) \]

Constraint conditions

<table>
<thead>
<tr>
<th>Condition</th>
<th>Dual Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_i - t_j \leq d_{i,j} )</td>
<td>( f_{i,j} )</td>
</tr>
<tr>
<td>( t_{2k-1} - t_k \leq -d_{k}^0 )</td>
<td>( f_{2k-1,k} )</td>
</tr>
<tr>
<td>( t_k - t_{2k-1} \leq d_{k}^1 )</td>
<td>( f_{k,2k-1} )</td>
</tr>
<tr>
<td>( t_{2k-1} - t_k \leq 0 )</td>
<td>( f_{2k-1,k} )</td>
</tr>
<tr>
<td>( t_k - t_{2k-1} \leq d_{k}^1 - d_{k}^{q-1} )</td>
<td>( f_{k,2k-1} )</td>
</tr>
<tr>
<td>( t_{2k-1} - t_k \leq 0 )</td>
<td>( f_{2k-1,k} )</td>
</tr>
<tr>
<td>( t_{2k} - t_{2k-1} \leq d_{k}^1 - d_{k}^{q-1} )</td>
<td>( f_{2k,2k} )</td>
</tr>
</tbody>
</table>

For the dual variables, \( f_{i,j} \) corresponding to an arc \((i,j) \in \overline{A}\) in the network \(G\), then \( f_{2k-1,k}, f_{k,2k-1}, f_{k,k+1}, f_{k,k+2} \) and \( f_{2k,k+1}\) should correspond to the arcs \((2k-1,k)\), \((k,2k-1)\), \((k,k+1)\), \((2k,k+2)\) and \((2k,k+1)\), \(q=2,3,\ldots,Q-1\). But the network \(G\) does not have these arcs, and the dual variables do not contain \( f_{2k-1,k} \) and \( f_{2k,k+1}\), which should correspond to the arcs \((2k-1,2k)\) and \((2k,2k-1)\). Therefore, although the network \(G\) corresponds to the primal model (Equations (10)–(14)), it does not correspond to the dual one of the model. We should transform the network \(G\) into \(G'\) by adding nodes \((k), (k), \ldots, (k)\) in \(G\) and transforming arcs \((2k-1,2k)\), \((2k,2k-1)\) into \((2k-1,k)\), \((k,k)\), \ldots, \((2k,k)\), \((2k,k)\), \((k,k)\), \((k,k)\), \((k,2k-1)\), as shown in Fig. 4.

Based on duality theory, we create the dual model of the primal model (Equations (10)–(14)) as follows:

\[ \min \varphi = \sum_{(i,j) \in \overline{A}} (-d_{i,j} f_{i,j}) + \sum_{k=1}^{K} (d_k^1 f_{2k-1,k} + d_k^2 f_{k,2k-1} + \sum_{q=2}^{Q_2} (d_q^1 - d_q^{q-1}) f_{k,k+1} + (d_q^2 - d_q^{q-1}) f_{2k,k+1}) \tag{A.1} \]

subject to

\[ \sum_j f_{i,j} - \sum_j f_{j,0} = 0 \tag{A.2} \]
\[ \sum_j f_{2k-1,j} - \sum_j f_{j,2k-1} = -a_k^0 \tag{A.3} \]
\[ \sum_j f_{i,j} - \sum_j f_{j,k} = a_k^1 - d_k^{q-1}, \quad q=1,2,\ldots,Q-1 \tag{A.4} \]
\[ \sum_j f_{k,j} - \sum_j f_{j,2k} = a_k^0 \tag{A.5} \]
\[ \sum_j f_{2k-1,j} - \sum_j f_{j,2k-1} = 0 \tag{A.6} \]
\[ f_{i,j} \geq 0 \tag{A.7} \]

We rewrite Equation (A.1) as

\[ \min \varphi = \sum_{(i,j) \in \overline{A}} w_{i,j} f_{i,j} \tag{A.8} \]

and \( w_{i,j} = -d_{i,j} \) for \( \forall (i,j) \in \overline{A}, w_{2k-1,k} = -d_k^0, w_{k,k+1} = d_k^1 - d_k^{q-1} \) for \( q=2,3,\ldots,Q-1, w_{2k,k+1} = d_k^2 - d_k^{q-1}, w_{k,2k-1} = d_k^0, \) and \( w_{i,j} = 0 \) for the other arcs \((i,j)\).

And we rewrite Equations (A.2)–(A.7) as (see the constraint (A.8) as a convention)
If defining the variable \( f \) as flow variable, we transform the network \( G' \) as follows:

1) Add a dummy beginning node \((s)\) and an end node \((t)\) to the network \( G' \).

2) For a node \((2k - 1)\), if \( \sum_{j} f_{2k-1,j} - \sum_{j} f_{2k,i} = -a_{i}^{l} > 0 \), then it means that the outflow is bigger than the inflow of the node, and we can balance the flow by adding an extra inflow, i.e. by adding an arc \((s, 2k - 1)\) with a flow \( f_{s,2k-1} = -a_{i}^{l} \). And if \( \sum_{j} f_{2k-1,j} - \sum_{j} f_{2k,i} = -a_{i}^{l} < 0 \), the inflow is bigger than the outflow, and we can balance the flow by adding an extra outflow, i.e. by adding an arc \((2k - 1, t)\) with a flow \( f_{2k-1,t} = a_{i}^{l} \).

3) Similarly, for a node \((k, q)\), \( q = 1, 2, \ldots, Q - 1 \), if \( \sum_{j} f_{k,j} - \sum_{j} f_{k,i} = -a_{q}^{l} > 0 \), then we can balance the flow of the node by adding an arc \((s, k)\) with a flow \( f_{s,k} = a_{q}^{l} - a_{q}^{i+1} \); and if \( \sum_{j} f_{k,j} - \sum_{j} f_{k,i} = -a_{q}^{l} < 0 \), then we can balance the flow of the node by adding an arc \((k, t)\) with a flow \( f_{k,t} = a_{q}^{i+1} - a_{q}^{l} \).

4) And for a node \((2k)\), if \( \sum_{j} f_{2k,j} - \sum_{j} f_{2k,i} = a_{q}^{l} > 0 \), then we can balance the flow of the node by adding an arc \((s, 2k)\) with a flow \( f_{s,2k} = a_{q}^{l} \); and if \( \sum_{j} f_{2k,j} - \sum_{j} f_{2k,i} = a_{q}^{l} < 0 \), then we can balance the flow of the node by adding an arc \((2k, t)\) with a flow \( f_{2k,t} = -a_{q}^{l} \).

The above network (marked as \( G' \)) is a standard flow network, e.g., the outflow of the beginning node \((s)\) is \( \sum_{j} f_{s,j} \), and the inflow of the end node \((t)\) is \( \sum_{j} f_{t,j} \), and the outflow and inflow of any other node is balanced. Therefore, we transform Equations (A.9) and (A.10) as followings:

\[
\sum_{j} f_{i,j} - \sum_{j} f_{j,i} = \begin{cases} 
\sum_{j} f_{j,i}, & i = s \\
0, & i \neq s, t \\
-\sum_{j} f_{j,i}, & i = t 
\end{cases} \quad (A.10)
\]

\[
f_{s,2k-1} = -a_{i}^{l}, \quad a_{i}^{l} < 0 \quad (A.11)
\]

\[
f_{2k-1,i} = a_{i}^{l}, \quad a_{i}^{l} > 0 \quad (A.12)
\]

\[
f_{s,k} = a_{q}^{l} - a_{q}^{i+1}, \quad a_{q}^{l} - a_{q}^{i+1} > 0 \quad (A.13)
\]

\[
f_{k,t} = a_{q}^{i+1} - a_{q}^{l}, \quad a_{q}^{i+1} - a_{q}^{l} < 0 \quad (A.14)
\]

\[
f_{s,2k} = a_{q}^{l}, \quad a_{q}^{l} > 0 \quad (A.15)
\]

\[
f_{2k,t} = -a_{q}^{l}, \quad a_{q}^{l} < 0 \quad (A.16)
\]

Because the new dummy arcs \((s, 2k - 1), (2k - 1, t), (s, k), (k, t), (s, 2k)\) and \((2k, t)\) do not exist in the network \( G' \), and the objective function based on \( G' \) should be the same and that in Equation (A.8), we set zero coefficients for these flows \( f_{s,2k-1} \), \( f_{2k-1,i} \), \( f_{s,k} \), \( f_{k,t} \), \( f_{s,2k} \) and \( f_{2k,t} \) that
We transform the limitation of flow into a limitation of capacity by letting the capacities of the arcs \((s, 2k - 1)\), \((2k - 1, t)\), \((s, k)\), \((k, t)\), \((s, 2k)\) and \((2k, t)\) are \(f_{s, 2k-1} = -a_i'\), \(f_{2k-1, t} = a_i'\), \(f_{s, k} = a_i^q - a_i'^q\), \(f_{k, t} = a_i^q - a_i'^q > 0\), \(f_{s, 2k} = a_i^q\), \(f_{2k, t} = -a_i^q\), \(a_i'^q < 0\) and \(f_{s, 2k} = a_i^q\) and \(f_{2k, t} = -a_i^q\), the other arcs have no limitations.

We label model 1-1 (Equations (A.10)–(A.16)) as follows:

\[
\begin{align*}
\min \varphi &= \sum_{(i,j) \in G} w_{i,j}f_{i,j} \\
\text{subject to} & \quad \left\{ \begin{array}{ll}
\sum_j f_{i,j} - \sum_j f_{j,i} &= 0 & i \neq s, t \\
\sum_j f_{i,j} &= 1 & i = s \\
f_{i,j} &= s_k^q - s_k'^q, & a_i^q - a_i'^q > 0 \\
f_{j,i} &= a_i^q - a_i'^q < 0 \\
f_{s, 2k-1} &= -a_i' \quad a_i' < 0 \\
f_{2k-1, t} &= a_i' \quad a_i' > 0 \\
f_{s, k} &= a_i^q - a_i'^q, \quad a_i^q - a_i'^q > 0 \\
f_{k, t} &= a_i^q - a_i'^q < 0 \\
f_{s, 2k} &= a_i^q \quad a_i^q > 0 \\
f_{2k, t} &= -a_i^q \quad a_i^q < 0 \\
w_{s, s_k} = w_{s, 2k} = w_{2k-1, s} = w_{t, s_k} = w_{t, 2k} = 0
\end{array} \right.
\end{align*}
\]
In a feasible solution, then the objective function of the primal model is maximum flow, the optimal solution of the model corresponds to the maximum flow of the flow network and the project is unfeasible.

According to the model 1-1, this illustrates that the optimal solution of the primal problem does not exist, because the flow limitations cannot be satisfied in all cases. But when transforming the flow limitations into capacity limitations by using the above method, the model becomes unrestricted, so we can obtain an optimal solution for the model under the conditions of (10)–(14)). If we obtain an optimal solution for the model 1-1, the condition whereby the model has a feasible solution is that the flow satisfies

\[ f_{s,t} = 0, \quad f_{i,j} = f_{i,j} - a_{i,j}, \quad f_{j,k} = a_{j,k} - a_{j,k} \]

for all arcs \((i,j,k)\), which are the non-restricted limitations to flow. This means that the model 1-2 is not exactly equivalent to the model 1-1. We need to modify the model 1-2 to make it equivalent to the model 1-1.

For the model 1-1, the condition whereby the model has a feasible solution is that the flow satisfies

\[ f_{s,t} = 0, \quad f_{i,j} = f_{i,j} - a_{i,j}, \quad f_{j,k} = a_{j,k} - a_{j,k} \]

for all arcs \((i,j,k)\), which are the non-restricted limitations to flow. This means that the model 1-2 is not exactly equivalent to the model 1-1. We need to modify the model 1-2 to make it equivalent to the model 1-1.

The model 1-2 has a feasible solution in all cases. According to the meaning of the minimum cost - maximum flow, the optimal solution of the model corresponds to the maximum flow of the flow network \(G\). In \(G\), the capacities of the arcs linked to the beginning node (s) or the end node (t) are limited, such that

\[ c_{s,t} = 0, \quad c_{i,j} = a_{i,j} - a_{i,j}, \quad c_{j,k} = a_{j,k} - a_{j,k}, \quad c_{s,t} = a_{s,t} \]

and \(c_{j,k} = -a_{j,k}\) for the other arc \((i,j)\). The parameters illustrate that for the maximum flow

\[ \max v(f) = \sum_{i,j} f_{i,j} \]

then

\[ f_{s,t} = 0, \quad f_{i,j} = f_{i,j} - a_{i,j}, \quad f_{j,k} = a_{j,k} - a_{j,k} \]

for all arcs \((i,j,k)\), which are the same as the model 1-1, and we can obtain the optimal solution of the primal problem by applying the optimal solution of the model 1-2 and the primal-dual relationship. But if

\[ \max v(f) \neq \sum_{i,j} f_{i,j} \]

the flows

\[ f_{s,t} = 0, \quad f_{i,j} = f_{i,j} - a_{i,j}, \quad f_{j,k} = a_{j,k} - a_{j,k} \]

will not be satisfied in all cases. According to the model 1-1, this illustrates that the optimal solution of the primal problem does not exist, and the project is unfeasible.

Therefore, we need judge the optimal solution of the model 1-2 to make the model perfectly equivalent to the model 1-1:

1) If the maximum flow of the model 1-2 is

\[ \max v(f) = \sum_{i,j} f_{i,j} \]

and the primal-dual relation, we can obtain optimal solutions \(\lambda' \) to the primal minimum project cost model. The optimal duration of each activity \(k\) is

\[ x_k = \tau_k - \tau_{k+1} \]

and the project duration with minimum cost is

\[ T = \tau_k - \tau_{k+1} \]
2) If the maximum flow of the model \(1-2\) is \(\max v(f) < \sum_{i,j} c_{ij} \), then the primal problem does not have an optimal solution, and the project is unfeasible in all cases.

The model \(1-2\) is a minimum cost - maximum flow model (Equations (15)–(18)), and is a dual model of the primal minimum project cost model (Equations (10)–(14)).

**Appendix B. Proof for Creating the Equivalent Model 2**

For the primal model (Equations (10)–(14)), we also consider its dual model formulations that Equations (A.8)–(A.10) and represents the dual variable as \(g_{ij}\).

If we regard \(g_{ij}\) as a transport volume from one node \(i\) to another node \(j\), viz. the transport volume of an arc \((i, j)\), then we can analyze the following cases based on the positive and negative of \(-a_{ik}^i\), \(a_{ik}^i - a_{ik}^{+1}\), and \(a_{kj}^0\).

1) For a node \((2k-1)\), if \(-a_{ik}^i > 0\), this illustrates that the transport out of the node is bigger than the transport into the node, whence we regard the extra transport volume \(-a_{ik}^i\) as a volume that is supplied from the node, and regard the node as a supply-side. And if \(-a_{ik}^i < 0\), then the transport into is bigger than the transport out of the node, and we regard the reduced transport volume \(a_{ik}^i\) as a volume that is demanded by the node \((2k-1)\), and a demand-side.

2) Similarly, for a node \((k_q)\), \(q=1,2,\ldots,Q-1\), if \(a_{ik}^i - a_{ik}^{+1} > 0\), transport out of the node is bigger than transporting in, and the node is regarded as a supply-side supplying \(a_{ik}^i - a_{ik}^{+1}\). And if \(a_{ik}^i - a_{ik}^{+1} < 0\), and transport in is bigger than transport out of the node, the node is a demand-side demanding \(a_{ik}^{+1} - a_{kj}^0\).

3) For a node \((2k)\), if \(a_{kj}^0 > 0\), this illustrates that the transport out is bigger than the transport into the node, and the node \((2k)\) is a supply-side, supplying \(a_{kj}^0\). And if \(a_{kj}^0 < 0\, and transport in is greater than transport, the node \((2k)\) is a demand-side demanding \(-a_{kj}^0\).

4) For a node with balanced transport in and out, we regard it as a transport interchange.

For \(-a_{ik}^i\, \, a_{ik}^i - a_{ik}^{+1}\) for \(q=1,2,\ldots,Q-1\) and \(a_{kj}^0\), because \(-a_{ik}^i + (a_{ik}^i - a_{ik}^0) + (a_{ik}^i - a_{ik}^0) + \ldots + (a_{ik}^{+1} - a_{kj}^0) + a_{kj}^0 = 0\), supply and demand are balanced. Therefore, Equations (A.9) and (A.10) is similar to the constraint condition of the transportation problem with balanced supply and demand. In the description of a standard transportation model, if there are \(U\) supply-sides and \(V\) demand-sides in the network \(G\), for convenience we number supply-sides \((u)\) for \(u=1,2,\ldots,U\) and demand-sides \((v)\) for \(v=1,2,\ldots,V\). Assuming \(h_{us}\) is the transport volume from a supply-side \((u)\) to a demand-side \((v)\), then we transform Equations (A.9) and (A.10) as follows,

\[
\sum_{(u)\in S} h_{us} = R_s
\]
\[
\sum_{(v)\in D} h_{us} = S_s
\]
\[
h_{us} \geq 0
\]

But this problem is differs from that of the classic transportation problem. In the classic problem, a material is directly transported from a supply- to a demand-side, a fare charged in the objective function, and an arc connects a supply-side and a demand-side. But in the network of the dual model, an arc may
connect any two of the supply-sides, demand-sides, and transport interchanges, thus, if we regard \( w_{i,j} \) as fare unit from one node \((i)\) to another \((j)\), then the objective function Equation (A.8) represents the sum of all fares from each of these combinations i.e. supply-sides to supply-sides, transport interchanges to transport interchanges etc.

In the network \( G \), there is at least one path \( \mu \) connecting any two nodes, which indicate a supply-side and a demand-side node. If the path \( \mu \) also passes other nodes, then we regard these nodes as transport interchanges between the supply- and demand-side. Therefore, we regard Equation (A.8) as the minimum fare sum from all paths connecting supply- and demand-sides. Since there may be many paths from a supply-side to a demand-side, and to realize the minimum total cost of transportation, we should choose the path with a minimum fare. Thus, we need to compute the minimum fare \( e_{u,v} \) from a supply-side \((u)\) to a demand-side \((v)\), that is

\[
e_{u,v} = \min \sum_{(i,j) \in \mu} w_{i,j}
\]

If we regard \( w_{i,j} \) as the length of an arc \((i,j)\), then the minimum fare from a supply-side \((u)\) to a demand-side \((v)\) is equal to the length of the shortest path connecting the two nodes. Because the network has cycles and arcs with negative lengths, we need to use a Bellman-Ford algorithm to compute the lengths of the shortest paths connecting a supply-side and all demand-sides, viz. the minimum fares. If the transport volume from a supply-side \((u)\) to a demand-side \((v)\) is \( h_{u,v} \), then we transform Equation (19) in the following way,

\[
\min \phi = \sum_{u=1}^{U} \sum_{v=1}^{V} e_{u,v} h_{u,v}
\]

In all paths with the minimum fares from the supply-sides to the demand-sides, if paths \( \mu_{u-v_1}, \mu_{u-v_2}, \ldots, \mu_{u-v_v} \) pass the arc \((i,j)\), the transport volume \( g_{i,j} \) of the arc is the sum of the transport volumes of these paths, viz.

\[
g_{i,j} = \sum_{k=1}^{v} h_{i,j+k}
\]

Hence, we transform the model (Equations (10)–(14)) into the following equivalent model,

\[
\min \phi = \sum_{u=1}^{U} \sum_{v=1}^{V} e_{u,v} h_{u,v}
\]

subject to

\[
\begin{align*}
\sum_{u=1}^{U} h_{u,v} &= S_v, \quad u = 1, 2, \ldots, U \\
\sum_{v=1}^{V} h_{u,v} &= D_u, \quad v = 1, 2, \ldots, V \\
h_{u,v} &\geq 0
\end{align*}
\]

The model represents the transportation problem with balanced supply and demand.
After solving the above model using a current algorithm, such as the table-manipulation method, and obtaining the optimal solution \( h_{ij} \), we can then obtain the optimal solution \( g_{ij} \) of the model (Equations (20)–(23)). Because this is a dual model of the minimum project cost model of a project under GPRs (Equations (10)–(14)), we can obtain the optimal solution \( i^* \) of the primal model based on the primal-dual relationship. For the primal minimum project cost problem with GPRs, the optimal duration of each activity, \( k \), is 
\[
    t_k = t_{j_{k-1}} - t_{i_{k+1}},
\]
and the project duration with the minimum cost is 
\[
    T^* = t_{i_1} - t_0.
\]

**Acknowledgment**

The first author thanks the Natural Science Foundation of China under Grant 71171079 and 71271081, the Natural Science Foundation of Jiangxi Provincial Department of Science and Technology in China under Grant 20151BAB211015, and the Jiangxi Research Center of Soft Science for Water Security & Sustainable Development for supporting the work.

**References**


**Zhi-Xiong Su** is a lecturer at Nanchang Institute of Technology, China. Dr. Zhi-xiong Su received his bachelor's degree in industrial engineering from Tianjin University in 2005. He received his master's degree and Ph.D. degree in technical economic and management science from North China Electric Power University in 2008 and 2014, respectively. Since 2014, he has been an lecturer, in Nanchang Institute of Technology, China. His research interests include network planning technology, project scheduling, optimization, and software algorithm design.

Dr. Su is a member of Chinese Society of Optimization, Overall Planning and Economical Mathematics and Operations Research Society of China.

**Han-Ying Wei** is a lecturer at Nanchang Institute of Technology, and an economist at Jiangxi Nuclear Power Corporation, China. Han-Ying Wei received her bachelor's degree and master's degree in technical economic and management science from North China Electric Power University in 2007 and 2010, respectively. She joined Jiangxi Nuclear Power Corporation in 2010, and left the corporation and joined Nanchang Institute of Technology in 2014. Her research interests include Management Science, project scheduling, software algorithm design, and Finance.

**Xue-Min Yu** is a college student in the School of Government, Beijing Normal University, China. Her research interests include mathematics modeling and software algorithm design.