Vietnam Journal of Mathematics **35**:4(2007) 387–398

Vietnam Journal
of
MATHEMATICS

Models and Algorithms for Robust PERT Scheduling with Time-dependent Task Durations

Michel Minoux

University Paris 6, France

Dedicated to Professor Hoang Tuy on the occasion of his 80th-birthday

Received July 17, 2007

Abstract. This paper investigates models and solution algorithms for solving PERT scheduling problems under uncertainty on task processing times. The proposed uncertainty model is very general in that it is capable of representing situations in which various distinct sources of uncertainty have to be considered ; and realistic in that it is consistent with the idea that situations featuring simultaneous occurrence of worst-case values for all (or a majority of) the task processing times do not have to be taken into account in the robustness analysis. From the computational point-of-view, a major interest of this uncertainty model is to provide very compact representations of potentially huge uncertainty sets, thus leading to efficient solution algorithms. We address both the standard case (where processing times do not depend on actual starting dates) and the case of time-dependent task durations.

1. Introduction

Uncertainty in scheduling problems is a major concern with respect to practical applications, and everyday life provides a wealth of observed situations where anticipated deadlines have to be significantly revised due to the occurrence of unfavourable events of various types (disease of an employee, breakdown of a manufacturing device, traffic jams delaying a product delivery, bad weather conditions, etc...).

In this paper we investigate robust extensions to the well-known PERT scheduling problem (i.e. the special subclass of scheduling problems with precedence constraints only) both in its basic version and in its time-dependent version (i.e. the case where task durations depend on the actual time instant at which processing the task is started). Considering first the classical (time-indepedent) version of the problem, we introduce in Sec. 2, a very general uncertainty model based on a compact (implicit) representation of uncertainty sets of potentially huge cardinality. This is obtained by assuming that each possible occurrence of uncertainty consumes a fraction of a given global amount of resources called "uncertainty budget", and by considering that the only allowed combinations of uncertain events to be taken into account in the model are those which do not exceed the global available uncertainty budget. It is then shown how the determination of a robust earliest termination date for a project under such an uncertainty model can be efficiently carried out by means of a dynamic programming-type recursion, providing a pseudo-polynomial solution algorithm. Also NP-completeness of the problem of determining an earliest robust termination date is shown, thus providing evidence of a fundamental difference between our approach and the one proposed by Bertsimas & Sim [2, 3] for the robust shortest path problem (which leads to polynomial time algorithms).

In Sec. 3 the above robust PERT scheduling model is further extended to handle the time-dependent version of the problem, i.e. the case where, for each task, processing time depends on the actual time instant at which task processing is started. To the best of our knowledge, this problem does not seem to have been addressed before in the literature. For this more general case, we also describe an efficient (pseudo-polynomial) algorithm to compute a robust earliest termination date.

The models and solution algorithms discussed in the present paper are described assuming a circuitless graph representation in which the tasks correspond to nodes, and precedence constraints correspond to arcs. An alternative way of stating a PERT scheduling problem is to use a circuitless graph representation in which the tasks correspond to arcs. We observe that the latter representation can easily be reduced to the former by using the so-called *line-graph* of the given graph (the nodes of the line-graph correspond to the arcs of the given graph and two nodes u and u' in the line-graph are connected with an arc (u, u') iff, in the given graph, the terminal endpoint of arc u and the initial endpoint of arc u' coincide). In view of this, it is clear that the robust PERT scheduling models and solution algorithms proposed in the present paper apply, irrespective of which initial graph representation is used to describe the problem under consideration.

2. Robust PERT Scheduling in the Time-independent Case

In this section we introduce the basic robust PERT scheduling model which will be used in Sec. 3 to handle the more complicated case of time-dependent task durations.

2.1. Problem Statement

Consider a PERT network represented as a directed circuitless graph $G = [\mathcal{N}, \mathcal{U}]$ in which the nodes correspond to tasks (the tasks are numbered $i = 1, \ldots, n$, the node set of G is thus $\mathcal{N} = \{1, \ldots, n\}$) and there is an arc $(i, j) \in \mathcal{U}$ with length d_i equal to the duration of task i whenever there is a precedence constraint stating that processing of task j should not start before completion of task i. In the sequel it will be assumed that the nodes of G are numbered according to a *topological ordering* of G, i.e. that $(i, j) \in \mathcal{U} \Rightarrow i < j$. Thus node 1, which has no predecessor, represents the initialization of the project, and node n, which has no successor, represents termination of the project. When all the task durations d_i are perfectly known (i.e. are not subject to fluctuations) the problem of finding the earliest termination date $\theta^*(n)$ amounts to finding a longest path from node 1 to node n in G, and it is well-known that this is efficiently done (in linear time $\mathcal{O}(m)$, where $m = |\mathcal{U}|$ is the number of arcs in G) via the dynamic programming-type recursion

$$\begin{cases} \theta^*(1) \leftarrow 0; \\ \text{for } j = 2 \text{ to } n: \\ \quad \theta^*(j) \leftarrow \text{Max}_{i \in \Gamma_j^{-1}} \{\theta^*(i) + d_i\}; \\ \text{end for.} \end{cases}$$

(In the above Γ_j^{-1} denotes the set of nodes i such that $(i, j) \in \mathcal{U}$, i.e. the set of immediate predecessors of j).

Now we are interested in the case where the task durations are *subject to uncertainty* and we want to determine an earliest termination date under the condition that it is achievable for any realization of uncertainties which might occur.

Such a value will be called an *optimal robust termination date*. It is readily seen that, in order to properly state and solve the problem of determining an optimal robust termination date, we have to define precisely the set $\mathcal{D}$ of all the possible occurrences of uncertainty for the task durations, which we call the *uncertainty set*.

A first simple idea in this respect is to take $\mathcal{D}$ as a product of real intervals $\left[d_1^0, d_1^+\right] \times \left[d_2^0, d_2^+\right] \times \cdots \times \left[d_{n-1}^0, d_{n-1}^+\right]$. Under this simple uncertainty model, each task i can feature any possible duration in the interval $\left[d_i^0, d_i^+\right]$, irrespective to what takes place for the other tasks (a kind of independence-type assumption). In that case it is easily seen that an optimal robust solution is simply obtained by solving the longest path problem on G with all task durations set to the worst case value d_i^+. Clearly, for many applications, such a solution would appear to be (far) too conservative. A reason for this is that, in most real applications, it would not be realistic to include in the uncertainty set $\mathcal{D}$ the case where all task durations take their worst-case values *simultaneously*. Indeed a situation where 10 % to 20% of the tasks have their duration only moderately increased (as compared with the nominal value d_i^0), and 2% to 5% of the task only have their duration significantly increased, would be much more typical of many real applications.

We therefore propose an alternative uncertainty model sufficiently general and flexible to handle a huge variety of structural features (such as the typical ones mentioned above) leading to define the uncertainty set $\mathcal{D}$ as the solution set of a well-defined auxiliary system of constraints. This is done as follows.

Each task i $(i = 1, \ldots, n-1)$ [1] doesn't have a well-defined associated processing time, but it will be assumed instead that its processing time can take on any value from

[1] We exclude task n because it is a dummy task with associated processing time zero, representing the termination of the project.

a given set of values $D_i = \left\{d_i^1, d_i^2, \ldots, d_i^\nu\right\}$ (for the sake of notational simplicity, we assume that all the sets D_i have equal cardinality ν, but of course the proposed model is more general and readily extends to the case of sets of nonuniform cardinalities). Associated with each d_i^k values in D_i, we assume that we are given a p-component integer vector $w_i^k \geq 0$ representing what will be referred to as the "uncertainty profile" of the corresponding realization d_i^k for the duration of task i. The various components of the w_i^k vectors may be interpreted for instance as corresponding to various sources of uncertainty (weather conditions, failure of some manufacturing equipment, etc.) and for a given realization d_i^k in D_i, the i^{th} component of w_i^k is 1 (or more generally a strictly positive integer) if the i^{th} source of uncertainty is a factor contributing to the outcome d_i^k, and 0 otherwise.

In addition to the above, a nonnegative p-component vector B is given (global "uncertainty budget"), the uncertainty set $\mathcal{D}$ for the task durations being defined as the set of all $(n-1)$-tuples of the form $(d_i^{k_1}, d_i^{k_2}, \ldots d_{n-1}^{k_{n-1}})$ satisfying

$$\sum_{i=1}^{n-1} w_i^{k_i} \leq B. \tag{1}$$

We note that such a model is sufficiently general to handle situations for which a given realization d_i^k is the result of joint influences of several distinct sources of uncertainty: the corresponding w_i^k vector in such cases will have several components equal to 1 (or, more generally, non zero). Also we note that in most applications, the largest among the d_i^k values in D_i (those related to the most unfavourable situations with respect to finding an earliest termination date) will tend to correspond to the w_i^k vectors of largest "weight" (as measured e.g. in terms of number of non zero components, or in terms of L_1 norm).

Also worth mentioning is another special case of the above general model, potentially useful in applications, where there are several (p) independent sources of uncertainty acting on *disjoint subsets of tasks* $U_1, U_2, \ldots, U_p$. Then, for each $i \in U_j$, the corresponding w_i^k vectors will have all components 0 except the j^{th} component which can be non zero. In such a case, the global uncertainty budget constraint just amounts to imposing one uncertainty budget constraint for each subset of tasks separately.

We note that, for practical applicability, the main limitation of the above model will be that the components of the w_i^k vectors and of the B vector be sufficiently small integers, in order that the quantity $\prod_{i=1}^{p}(B_i + 1)$ remains sufficiently small (typically less than 10^3 to 10^4). However, even with this restriction, an attractive feature of the model is its capability of providing very compact (implicit) representations of uncertainty sets with potentially huge cardinalities. To be convinced of this, just consider a fairly moderate size instance with $n = 50$ tasks, $p = 1$, $B = 10$, and, for each task i, only two possible w_i^k values $w_0^1 = 0$, $w_i^2 = 1$ (thus $\nu = 2$). The number of distinct $(n-1)$-tuples satisfying (1) is $\begin{pmatrix} 1 \\ 49 \end{pmatrix} + \begin{pmatrix} 2 \\ 49 \end{pmatrix} + \cdots + \begin{pmatrix} 10 \\ 49 \end{pmatrix}$ which is greater than 10^{12}.

2.2. A Solution Algorithm for the Basic Robust PERT Scheduling Problems

We now investigate a solution algorithm for determining an optimal robust termination date under the above-defined uncertainty set for the task durations. To that aim, we first observe that the problem may be viewed as a special instance out of a whole family of problem instances, denoted by $P[i, \sigma]$, associated with each node $i \in \mathcal{N}$, and each integer valued p-component vector $\sigma = \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_p \end{pmatrix} \geq 0$. $\theta^*[i, \sigma]$, the optimal solution value of the problem $P[i, \sigma]$, is defined as the earliest termination date for the problem of scheduling the subset of tasks $\{1, 2, \ldots, i\}$ which is robust against the uncertainty set $\mathcal{D}^{\sigma}_{[i]}$ formed by all vectors $\left(d_j^{k_j}\right)_{j=1,\ldots,i}$ verifying:

$$\sum_{j=1}^{i} w_j^{k_j} \leq \sigma.$$

In other words, $\theta^*[i, \sigma]$ is the optimal robust solution value of the subproblem restricted to the subset of tasks $\{1, 2, \ldots, i\}$, given the uncertainty budget σ. In this context the problems that we originally want to solve is just $P[n, B]$, the corresponding robust earliest termination date being $\theta^*[n, B]$.

For a given node i, we will refer to $\theta^*[i, \sigma]$ as the optimal value for node i when node i takes *uncertainty status* σ. We denote by Σ the set of all possible values for σ, and we observe that $|\Sigma| = (B_i + 1) \times (B_2 + 1) \times \cdots \times (B_q + 1)$. The following result shows how the $\theta^*[j, \sigma]$ values for all $\sigma \in \Sigma$ can be computed, assuming that all the $\theta^*[i, \sigma]$ values for $i \in \Gamma_j^{-1}$ have previously been obtained.

Proposition 1. *The $\theta^*[j, \sigma]$ values can be computed via the following recursion:*

$$\theta^*[j, \sigma] = \mathrm{Max}_{i \in \Gamma_j^{-1}} \left\{ \mathrm{Max}_{k \in K_i^{\sigma}} \left\{ d_i^k + \theta^* \left[i, \sigma - w_i^k \right] \right\} \right\}, \tag{2}$$

$$\theta^*[1, \sigma] = 0, \quad \forall \sigma \in \Sigma, \tag{3}$$

where K_i^{σ} denotes the set of indices $k \in \{1, 2, \ldots, \nu\}$ such that $w_i^k \leq \sigma$. Moreover, if we denote

$$\varphi^*[j, \sigma] = \mathrm{argmax}_{i \in \Gamma_j^{-1}} \left\{ \mathrm{Max}_{k \in K_i^{\sigma}} \left\{ d_i^k + \theta^* \left[i, \sigma - w_i^k \right] \right\} \right\}, \tag{4}$$

then $\varphi^*[j, \sigma]$ can be interpreted as the immediate predecessor of j on an optimal robust longest path from node 1 to node j with the uncertainty budget σ.

Proof. The proof is by induction, assuming the result is true for all $i = 1, \ldots, j-1$, and all $\sigma \in \Sigma$.

Suppose $i \in \Gamma_j^{-1}$ is the predecessor of j on an optimal robust longest path from 1 to j with uncertainty budget σ. We know that, due to uncertainty, the length of arc (i, j) can take on any value $d_i^k \in D_i$ which is compatible with the allowed budget σ, i.e. such that $w_i^k \leq \sigma$. In order to ensure the existence of a feasible schedule in all the above possible occurrences of uncertainty w.r.t. task i, the earliest termination date $\theta^*[j, \sigma]$ for the subset of tasks $1, 2, \ldots, j$ should not be less than the worst-case (i.e. maximum) value among the set of values: $d_i^k + \theta^*[i, \sigma - w_i^k]$ for all $k \in K_i^{\sigma}$.

So, if i, the immediate predecessor of node j on a robust optimal path from 1 to j with uncertainty budget σ is known, the earliest termination date for all tasks up to task j in the worst-case is exactly

$$\psi_i[j,\sigma] = \mathrm{Max}_{k \in K_i^\sigma} \left\{ d_i^k + \theta^* \left[i, \sigma - w_i^k \right] \right\}.$$

Now since, again, we want to guarantee the existence of a feasible schedule terminating not later that $\theta^*[j,\sigma]$ in all situations corresponding to the uncertainty set $\mathcal{D}_{[j]}^\sigma$, the latter value should correspond to the $\psi_{i'}[j,\sigma]$ values over all $i' \in \Gamma_j^{-1}$. The proof of this is readily obtained by contradiction: if $\psi_i[j,\sigma] < \psi_{i'}[j,\sigma]$ for some $i' \in \Gamma_j^{-1}$, $i' \neq i$, then we can find a possible set of assignments in the uncertainty set $\mathcal{D}_{[j]}^\sigma$ such that there is a path from 1 to j through i' having duration greater than $\psi_i[j,\sigma]$, which proves that the project cannot be completed within the time limit $\psi_i[j,\sigma]$ for all occurrences of uncertainty in the uncertainty set $\mathcal{D}_{[j]}^\sigma$. Thus the predecessor of node j on an optimal robust longest paths from 1 to j with uncertainty budget σ should be that node i achieving the maximum of the $\psi_{i'}[j,\sigma]$ values over all $i' \in \Gamma_j^{-1}$, which is exactly what corresponds to relations (2)–(4). ∎

Based on the above result, we now state the complexity of the procedure computing $\theta^*[n,B]$.

Proposition 2. *The computational complexity of applying the recursion* (2)–(4) *to determine* $\theta^*[n,B]$ *is* $\mathcal{O}(\nu m|\Sigma|)$.

Proof. For each node j, the computation of the $|\Sigma|$ values $\theta^*[j,\sigma]$ requires $\mathcal{O}(\nu|\Sigma|d^-(j))$ elementary operations, where $d^-(j)$ denotes the in-degree of node j. The result then follows from the identity $\sum_{j=1}^n d^-(j) = m = |\mathcal{U}|$. ∎

In view of this, it is seen that Proposition 1 provides a *pseudo-polynomial algorithm* for solving the problem of computing an earliest robust termination date under the uncertainty model of Subsec. 2.1.

It is worth observing here that NP-completeness of the problem can easily be deduced by reduction from KNAPSACK problem. Indeed, let us consider an arbitrary instance of KNAPSACK: we are given a set of n items $I = \{1, 2, \ldots, n\}$, each with size $a_i \in \mathbb{Z}_+$ and value $v_i \in \mathbb{Z}_+$, and two positive integers $b > 0$ and $K > 0$. The question is to decide whether there exists a subset $S \subseteq I$ such that $\sum_{i\in S} a_i \leq b$ and $\sum_{i\in S} v_i \geq K$ or not. This can be converted into an instance of the earliest robust termination date problem as follows.

With each item $i \in I$, we let correspond a task with uncertain duration $d_i \in \{0, v_i\}$. We add a dummy task $n+1$ with duration 0 (without uncertainty) to represent termination of the project, and we take $p = 1$. Also we agree that, for all i, the value 0 for d_i has uncertainty profile 0, and the value v_i has uncertainty profile a_i. The total available uncertainty budget is $B = b$, and the associated circuitless graph representing the precedence constraints contains the n arcs $(i, i+1)$ for $i = 1, \ldots, n$.

Now it is readily seen that the answer to the Knapsack decision problem is YES if and only if the project cannot be terminated earlier than time K for all possible occurrences of uncertainty in the corresponding robust PERT scheduling problem.

Actually, the main interest of the above NP-completeness property for the robust optimization problem addressed in the present paper is that it makes immediately clear that our approach is different from the one proposed by Bertsimas & Sim [2, 3], leading to *polynomial algorithms for robust shortest path problems*. Indeed, it is easily realized that solving our problem does not simply reduce to considering longest paths instead of shortest paths (i.e. changing Min to Max) in Bertsimas & Sim's approach (see Minoux [5] for further details on the differences with Bertsimas & Sim's approach). Though the basic ideas of the underlying uncertainty models have some common flavours, our model here is different since it gives rise to an NP-complete problem. Also worth pointing out is that Bertsimas & Sim's model corresponds to the case $p = 1$ in our model (i.e. a single uncertainty budget constraint) while we allow the possibility of considering several uncertainty budget constraints simultaneously.

3. Robust PERT Scheduling with Time-dependent Task Durations

We now proceed to show how the basic robust PERT scheduling model introduced in Sec. 2 can be extended to handle the more complex situation where the time needed to process a task depends on the time instant at which processing is started.

In the absence of uncertainty, we only have to assume that, for each task i, we are given a nondecreasing function $h_i : \mathbb{R}^+ \to \mathbb{R}^+$, defined as follows: $\forall t_i \geq 0$, $h_i(t_i)$ is the earliest termination date for task i when processing of this task is started at time t_i. In that case, an earliest termination date for the whole project is simply determined via the dynamic-programming type recursion

$$\begin{cases} t_1^* \leftarrow 0\,; \\ \text{for } j = 2, \dots, n\ : \\ \quad t_j^* \leftarrow \text{Max}_{i \in \Gamma_j^{-1}} \left\{ h_i \left(t_i^* \right) \right\} \\ \text{end for.} \end{cases}$$

Since the earliest termination date t_n^* resulting from the above computation clearly depends on the choice of the functions h_i, we will use the notation $t_n^*[h_1, \dots, h_n]$ to underline this dependence.

We observe that there are many practical contexts requiring consideration of time-dependent processing times. For instance machines have to be periodically referred or require to be stopped for periodical maintenance tasks. Human workers alternate periods of work and period of rest, etc. Suppose, for instance, that a team has to carry out a task of nominal duration 7 hours. If the task execution is started on friday at 8:00 am, it can be expected to be completed on the same day 4:00 pm. However if it is started on friday at 2:00 pm, it can only be expected to be completed by next monday at 11:00 am (assuming active periods on a normal working day are 8:00 am – 12:00 am and 1:00 pm – 5:00 pm, and no activity on saturday and sunday).

Now, considering uncertainty on task processing times, we show how to extend the model introduced in Sec. 2 to the context of time-dependent processing times. To do this, we associate, with each task i (each node i in G), ν nondecreasing functions of time $h_i^1, h_i^2, \dots, h_i^\nu$, and corresponding p-component vectors (uncertainty profiles) $w_i^1, \dots, w_i^\nu \in \mathbb{N}^p$. (Note here again that, for the sake of notational simplicity, we assume that the sets of functions associated with the various tasks have the same cardinality ν, but of course this is by no means a limitation of the model). Let $B \in \mathbb{N}^p$ a p-component vector representing the total available uncertainty budget. Under the resulting uncertainty model, the possible occurrences of uncertainty correspond to selecting the functions $h_1^{k_1}, h_2^{k_2}, \ldots, h_{n-1}^{k_{n-1}}$ where the indices $k_1, k_2, \ldots, k_{n-1}$ satisfy the global uncertainty budget constraint

$$\sum_{i=1}^{n-1} w_i^{k_i} \leq B. \tag{5}$$

The problem of determining an earliest robust termination date for the project can then be formulated as

$$\text{Maximize } t_n^* \left[h_1^{k_1}, h_2^{k_2}, \ldots, h_{n-1}^{k_{n-1}}\right], \tag{6}$$

where the maximum is taken over all possible $(k_1, k_2, \ldots, k_{n-1}) \in \{1, \ldots, \nu\}^{n-1}$ satisfying (5).

The following result shows how this can be done via an extended version of the recursion (2)–(4).

Proposition 3. *The earliest robust termination date $\theta^*[n, B]$ for the time-dependent PERT scheduling problem can be determined via the recursion:*

$$\theta^*[j, \sigma] \leftarrow \left\{ \underset{i \in \Gamma_j^{-1}}{\text{Max}} \underset{k \in K_i^\sigma}{\text{Max}} \left\{ h_i^k \left(\theta^* \left[i, \sigma - w_i^k\right]\right)\right\} \right\}, \tag{7}$$

$$\theta^*[1, \sigma] \leftarrow 0 \quad (\forall \sigma \in \Sigma). \tag{8}$$

Moreover, the resulting computational complexity of $\mathcal{O}(\nu m |\Sigma|)\mathcal{O}(h)$ where $\mathcal{O}(h)$ denotes the complexity of one function evaluation.

Proof. The proof is similar to the one given for Propositions 1 and 2 in Sec. 2. ■

To illustrate the above-stated procedure, consider the PERT scheduling problem under uncertainty represented by the 7-node graph shown in Figure 1, and Table 1 which provides, for each task $i = 1, \ldots, 6$, the functions of time h_i^1, h_i^2, h_i^3 (here we take $\nu = 3$) corresponding to the three uncertainty profiles $w_i^1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $w_i^2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $w_i^3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ (thus we assume $p = 2$). The global uncertainty budget is taken to be $B = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$. Task 7 is a dummy task (of duration 0) representing the end of the project.

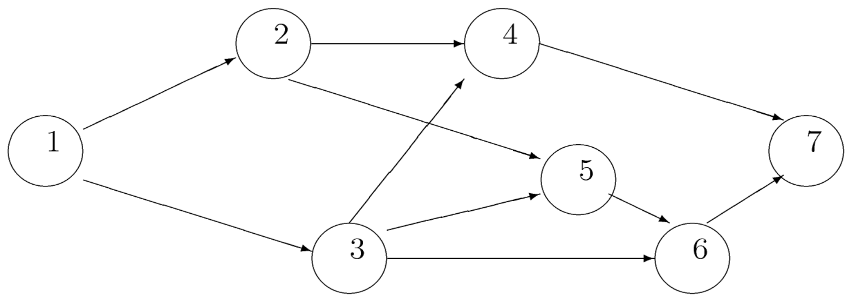

Fig. 1. A 7-node graph to illustrate the robust PERT scheduling problem with time-dependent processing times

	$w_i^1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$w_i^2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$w_i^3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
node $i = 1$	$f(\cdot, 6)$	$f(\cdot, 10)$	$f(\cdot, 14)$
node $i = 2$	$f(\cdot, 7)$	$f(\cdot, 9)$	$f(\cdot, 15)$
node $i = 3$	$f(\cdot, 10)$	$f(\cdot, 17)$	$f(\cdot, 13)$
node $i = 4$	$f(\cdot, 10)$	$f(\cdot, 15)$	$f(\cdot, 20)$
node $i = 5$	$f(\cdot, 5)$	$f(\cdot, 7)$	$f(\cdot, 7)$
node $i = 6$	$f(\cdot, 4)$	$f(\cdot, 8)$	$f(\cdot, 10)$

Table 1. For each node $i = 1, \ldots, 6$, the table provides the three functions of time h_i^1, h_i^2, h_i^3 corresponding to the various possible occurrences of uncertainty with the associated uncertainty profiles $w_i^1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $w_i^2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $w_i^3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Each entry is defined by specifying the value of the parameter d in the family of functions $f(t, d)$ given by (8).

To make the example easier to follow, all the h_i^k functions are assumed to be derived from a single function of time depending on a single parameter d and defined as follows

$$f(t, d) = 12 \left(\left\lfloor \frac{t}{12} \right\rfloor + \alpha(t) \right) + \beta(t) + d + 12 \left\lfloor \frac{\beta(t) + d}{12} \right\rfloor \tag{9}$$

with $\alpha(t) = \left\lfloor \frac{t}{12} \right\rfloor \pmod 2$ and $\beta(t) = \left(t - 12 \cdot \left\lfloor \frac{t}{12} \right\rfloor\right)(1 - \alpha(t))$ ($\lfloor x \rfloor$ denotes the largest integer not greater than x).

We note that such a function is typical of practical situations where tasks are carried out by teams working according to rosters alternating periods of work and periods of rest. Indeed the above formula (9) represents the case of alternating 12 hour work periods and 12 hours rest periods. Under such an organization for instance, a task of duration 26 hours starting at time $t = 8$ would be completed at time $f(8, 26) = 58$

(this value, given by (9) corresponds to 4 hours of work on the first day, 12 hours on the second day and 10 hours on the 3rd day).

Of course, in practical applications, things will be more complicated and we will have to take into account weekends, holidays, etc. We note that, for representing such intricate cases, a single analytic formula like (8) will probably be difficult (if not impossible) to set up, and the h_i^k functions will have to be represented as lookup tables.

The values $\theta^*[j,\sigma]$ obtained by application of the recursion (7)–(8) (Proposition 3) are shown in Table 2.

	$\sigma = \begin{pmatrix}0\\0\end{pmatrix}$	$\sigma = \begin{pmatrix}1\\0\end{pmatrix}$	$\sigma = \begin{pmatrix}0\\1\end{pmatrix}$	$\sigma = \begin{pmatrix}1\\1\end{pmatrix}$
$\theta^*[1,\sigma]$	0	0	0	0
$\theta^*[2,\sigma]$	6	10	26	-
$\theta^*[3,\sigma]$	6	10	26	-
$\theta^*[4,\sigma]$	28	35	48	55
$\theta^*[5,\sigma]$	25	29	33	49
$\theta^*[6,\sigma]$	30	35	50	55
$\theta^*[7,\sigma]$	50	57	72	79

Table 2. The result of applying the recursion (7)–(8). The earliest robust termination date for the example is $\theta^*\left[7, \begin{pmatrix}1\\1\end{pmatrix}\right] = 79$.

Let us illustrate the procedure on the computation of the value $\theta^*\left[7, \begin{pmatrix}1\\1\end{pmatrix}\right]$. We have

$$\theta^*\left[7, \begin{pmatrix}1\\1\end{pmatrix}\right] = \text{Max}\{u, v\},$$

where

$$u = \text{Max}\left\{h_4^1\left(\theta^*\left[4, \begin{pmatrix}1\\1\end{pmatrix}\right]\right) ; h_4^2\left(\theta^*\left[4, \begin{pmatrix}0\\1\end{pmatrix}\right]\right) ; h_4^3\left(\theta^*\left[4, \begin{pmatrix}1\\0\end{pmatrix}\right]\right)\right\}$$

$$v = \text{Max}\left\{h_6^1\left(\theta^*\left[6, \begin{pmatrix}1\\1\end{pmatrix}\right]\right) ; h_6^2\left(\theta^*\left[6, \begin{pmatrix}0\\1\end{pmatrix}\right]\right) ; h_6^3\left(\theta^*\left[6, \begin{pmatrix}1\\0\end{pmatrix}\right]\right)\right\}$$

Using the $\theta^*[4,\sigma]$ and $\theta^*[6,\sigma]$ values shown in Table 2 and the fact that $h_4^1(t) = f(t,10)$, $h_4^2(t) = f(t,15)$, etc. (see Table 1) we obtain

$$\begin{aligned} u &= \text{Max}\{f(55,10)\ ;\ f(48,15)\ ;\ f(35,20)\} \\ &= \text{Max}\{77\ ;\ 75\ ;\ 79\} = 79, \\ v &= \text{Max}\{f(55,4);\ f(50,8);\ f(35,10)\} \\ &= \text{Max}\{59;\ 58;\ 57\} = 59. \end{aligned}$$

So we have $\theta^*\left[7, \begin{pmatrix}1\\1\end{pmatrix}\right] = 79$ as shown in Table 2. Indeed this value represents the earliest termination date of the project which is robust against the uncertainty model under consideration.

This value can also be obtained (but of course much less efficiently) by enumerating all possible occurrences of uncertainty allowed by the uncertainty model, each time computing the earliest termination date, and taking the maximum of the resulting values (as suggested by (6)).

In the example there are 30 6-tuples $(k_1, k_2, k_3, k_4, k_5, k_6)$ with $k_i \in \{1, 2, 3\}\ \forall i$, and satisfying

$$\sum_{i=1}^{6} w_i^{k_i} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} = B.$$

For instance, taking the 6-tuple defined as: $k_1 = k_3 = k_4 = k_5 = 1$, $k_6 = 2$, $k_2 = 3$ we get an earliest termination date 58 (the corresponding critical path being $1 \to 2 \to 5 \to 6 \to 7$).

The worst-case arises with the 6-tuple for which $k_1 = k_2 = k_5 = k_6 = 1$, $k_3 = 2$, $k_4 = 3$, leading to an earliest termination date 79 (the corresponding critical path being $1 \to 3 \to 4 \to 7$).

4. Conclusion and Perspectives

Under the uncertainty model introduced in the present paper, it has been shown that robust versions of the well known PERT scheduling problem can be solved in pseudo-polynomial time. Of course the same type of uncertainty model could be used to define robust versions of many other combinatorial optimization problems. Applications involving robust multistage dynamic programming problems with uncertainty both influencing the reward function and the state-transition equations seem to be of special interest in this respect. This is left for future research.

References

1. A. Ben-Tal A., Nemirovski A. (1999). Robust solutions to uncertain programs, *Oper. Res. Letters* **25** (1999) 1–13.
2. D. Bertsimas and M. Sim, Robust discrete Optimization and network flows, *Math.. Prog. B* **98** (2003) 49–71.
3. D. Bertsimas and M. Sim , The price of robustness, *Oper. Res.* **52** (2004) 35–53.
4. P. Kouvelis and G. Yu, *Robust Discrete Optimization and its Applications*, Kluwer Academic Publisher, Boston, 1997.
5. M. Minoux, Duality, *Robustness and 2-Stage Robust Decision Models. Application to Robust PERT Scheduling*, Annales du LAMSADE, University Paris-Dauphine, France, 2007.
6. J. M. Mulvey, R. J. Vanderbei, and S. A. Zenios, Robust optimization of large Scale systems, *Oper. Res.* **43** (1995) 264–281.
7. A. L. Soyster, Convex Programming with set-inclusive constrained and applications to inexact linear programming, *Oper. Res.* **21** (1973) 1154–1157.

8. A. L. Soyster, Inexact linear programming with generalized resource sets, *EJOR* **3** (1979) 316–321.
9. G. Yu and J. Yang, On the robust shortest path problem, *Computers and Oper. Res.* **25** (1998) 457–468.