WILEY InterScience® DISCOVER SOMETHING GREAT

The Stochastic Time–Cost Tradeoff Problem: A Robust Optimization Approach

Izack Cohen, Boaz Golany, and Avraham Shtub
Faculty of Industrial Engineering and Management, The Technion, Israel Institute of Technology, Haifa, Israel 32000

We consider the problem of allocating resources to projects performed under given due dates and stochastic time–cost tradeoff settings. In particular, we show how to implement a state-of-the-art methodology known as "robust optimization" to solve the problem. In contrast to conventional approaches, the model we develop results in management *policies* rather than optimal values for the original decision variables. Hence, the project manager can postpone decisions to the point of time when they are actually required and then make them according to the optimal policy (which employs cumulative data on the project progress). The solutions are guaranteed to be robust—that is, ensuring feasibility except when the uncertain parameters assume extreme values. Still, as we demonstrate through an extensive numerical example, the price we need to pay to obtain that robustness is relatively small even for high uncertainty levels. © 2006 Wiley Periodicals, Inc. NETWORKS, Vol. 49(2), 175–188 2007

Keywords: project management; time–cost tradeoff; robust optimization; linear programming.

1. INTRODUCTION

The *time–cost tradeoff problem (TCTP)* in project management assumes that the duration of each activity is a function of the resource allocated to it. The objective of models that solve the TCTP is typically stated either as a minimization of the project's costs under a specified due-date (*the due-date problem*) or as a minimization of the project's duration under a given budget (*the budget problem*).

The TCTP has been investigated quite thoroughly during the last few decades (see, e.g., the review in [11]). Although most of the TCTP research to date assumes deterministic activity durations, in practice, these durations are uncertain, resulting in a *stochastic TCTP (STCTP).* The topic of stochastic activity networks was analyzed by Hagstrom [13], who proved that computing the cumulative distribution function, or even the mean value of the project duration is NP- hard. Hence, the relatively few publications on the STCTP, such as [12, 15], have largely adopted a minimum expected value approach (with the same objective functions as in the TCTP) and tried to solve it with heuristic procedures. This approach does not provide management with an operational solution—what decision to take under each possible realization of the random variables. Rather, it merely provides an estimate of the expected project cost (or duration). Herroelen and Leus [14] review procedures aimed at generating robust schedules for project management and suggest a framework to identify an appropriate scheduling procedure for different project scheduling environments. According to this framework, in low variability environments a baseline schedule might be enough, but as the variability increases, so does the need for a more robust baseline schedule and for adjustments during the project's execution. They also discuss multiproject environments and classify the methodologies according to two key determinants: the degree of variability in the work environment, and the degree of independence of the project. Chen et al. [10] suggest a robust approach for project management. Their approach captures asymmetrically distributed random data, but because their model is nonadaptive, it lacks a learning mechanism that might prevent overconservatism in the results (i.e., assuming that the actual durations will be long and consequently planning to perform some activities in a crash mode). The approach that we present here enhances the literature by proposing a new model that uses adaptive robust project management policies.

In general, the STCTP can be classified as a multistage decision problem. In such settings, it is natural to address the problem using *Stochastic Programming with Recourse* (SPR), which is the prevailing approach for uncertain mathematical programs (particularly, linear programs) in the OR literature (see, e.g., [21]). The SPR approach can be considered as a viable approach for some problems. Unlike stochastic dynamic programming, it is not plagued with the curse of dimensionality in the state variables. It can also

Received March 2005; accepted August 2006.
Correspondence to: B. Golany; e-mail: golany@ie.chnion.ac.il
DOI 10.1002/net.20153
Published online in Wiley InterScience (www.interscience.wiley.com).

handle global nonseparable constraints quite naturally [21]. However, its usefulness is limited to problems with very few stages, because it is adversely affected by the exponential explosion of the event tree when the number of stages increases. Furthermore, it is difficult to apply SPR whenever the decision-making process cannot be modeled according to a predefined sequence of stages. Consider the STCTP due-date problem: in each stage we are required to make decisions regarding the starting times and crashing levels of the next activities. Apart from the trivial case of a *serial* project network, it is clear that the order in which the stages are executed is not known in advance as it depends upon the realized values of predecessor activities. In such settings it would be difficult to implement SPR.

Another difficulty associated with SPR, where the typical objective is to minimize expected cost, is the need to provide the probability distribution functions of the underlying stochastic parameters. This requirement creates a heavy burden on the user, because in many real-world situations, such information is unavailable or difficult (costly) to obtain. In particular, many projects are unique by definition, and there are no data to support even the construction of empirical distribution functions for them. Thus, the need arises for a new optimization methodology that can address the uncertain nature of the problem without making specific assumptions on the probability distributions. To be useful for real-world project networks, such a methodology must be computationally tractable for problems with a large number of decision variables and stochastic parameters.

This article introduces a new model for the STCTP. Our model uses the *Affinely Adjustable Robust Counterpart* (AARC) methodology that was developed by Ben-Tal et al. [3] to determine optimal policies for the STCTP with continuous, linear time–cost curves. The AARC methodology stems from Robust Optimization (RO)—a methodology that was developed by Ben-Tal and Nemirovski [7] for solving large-scale convex optimization problems in which portions of the data are uncertain and known only to vary within given *uncertainty sets.* The RO methodology, which has been successfully applied to some large-scale and highly complex problems [4, 5, 19], is particularly useful for solving uncertain *Linear Programming (LP)* problems as it allows uncertainty in any part of the data (the activity matrix, the cost coefficients and the right-hand side vector). Thus, a STCTP with linear time–cost curves is a natural candidate for RO application.

Although RO models yield optimal *values* for the decision variables, the AARC solutions are given in terms of optimal *policies.* These policies are *adaptive* in nature because they combine information that is accumulated over time (in our context, durations of completed activities) with uncertain information (e.g., duration of future activities). The decisions given by these policies (i.e., optimal crashing levels and start times) are *robust* as they guarantee feasibility (e.g., meeting the project's due-date) under any possible realization of the uncertain data. (In some cases we are willing to take a small chance that the model would yield infeasible solutions when in most cases we would get a more efficient solution; see Section 4.2.2.)

The rest of this article is organized as follows. In the next section, we provide a literature review of the TCTP and STCTP. Then, in Section 3 we make some adjustments to the basic TCTP model and formulate an LP model for the STCTP. In Section 4 we present the principles of RO and implement the methodology by constructing a *robust counterpart* to our problem. Then, in Section 5, we present and discuss a numerical example for this model. Finally, Section 6 contains a summary and concluding remarks.

2. TCTP AND STCTP: A LITERATURE REVIEW

The TCTP assumes the existence of a cost function that determines by how much the duration of activities can be reduced (crashed) beyond their *normal* durations. Crashing may be inevitable if this is the only way to meet the project due date, but it can also be economically worthwhile when project costs include a significant fixed charge component that is linear in the project duration.

The TCTP can be represented by a directed *activity-on-arc* (*AOA*) graph $G = (V, E)$ consisting of a set of nodes $V = \{1, 2, \ldots, n\}$ and a set of activities $E = \{1, 2, \ldots, m\}$. Each activity $\{ij\}$ can be performed with a duration $t_{ij} \in \mathcal{T}_{ij}$, where $\mathcal{T}_{ij} \subseteq \Re_+$ is the set of possible durations, at a cost $c_{ij}(t_{ij})$, where c_{ij} is a nonincreasing function of t_{ij}. The TCTP usually aims at minimizing the project duration subject to an upper bound on its budget or at minimizing the project cost subject to a predetermined due date.

The most common version of the TCTP is the *linear, continuous time–cost tradeoff problem* (LTCTP; see [15], p. 493, and [20], p. 383). This model assumes that each $\mathcal{T}_{ij}$ is a closed interval $\mathcal{T}_{ij} = [M_{ij}, T_{ij}]$ and that c_{ij} is a linear and decreasing function on $\mathcal{T}_{ij}$. Linear Programming (LP) is the standard solution approach for the LTCTP (see [1, 16, 22], p. 122). Other techniques, such as the *Dantzig-Wolfe decomposition,* have also been applied to the LTCTP [22].

This model has been extended to *nonlinear, continuous time-cost tradeoff* cases where c_{ij} is assumed to be convex and nonincreasing on $\mathcal{T}_{ij}$, implying that crashing an activity becomes increasingly more expensive as its duration is reduced. Wiest and Levy [22] proposed a *piecewise linear approximation* to deal with nonlinearity in TCT curves. Deckro et al. [11] assumed a quadratic TCT relationship and applied a *quadratic programming* technique to solve the model.

Another known version of TCTP is the *discrete time–cost tradeoff problem* (DTCTP). This model assumes that $\mathcal{T}_{ij}$ consists of a discrete set of possible activity durations, and c_{ij} is decreasing on $\mathcal{T}_{ij}$. This representation is appropriate when each activity is associated with a finite number of *modes* (discrete time–cost combinations) representing different methods of executing the activity (see the review in Brucker et al. [9]).

In this article we focus on the LTCTP. Let

$$c(t) = \sum_{\{ij\} \in E} c_{ij}(t_{ij})$$

denote the total direct cost of project activities for a specific *realization* $t \in \Re_+^m$ (where t is a vector of m durations $t_{ij} \in \mathcal{T}_{ij}$), and the associated project duration (that is the duration of the longest path through the project network) is $T_{max}(t)$. We assume that the project carries a time-related "overhead" cost C, so that the sum of project costs is $Z(t) = c(t) + C \cdot T_{\max}(t)$. Then, the *due-date problem* is formulated as:

$$\min_i \{Z(t) | t_{ij} \in \mathcal{T}_{ij}, T_{\max}(t) \le D\}, \qquad (1)$$

where D is a predetermined due date.

Surprisingly, although the TCTP has been known for decades, research into its stochastic variation is just emerging. Leu et al. [18] implemented a *genetic algorithm* heuristic and Gutjahr et al. [12] used a *stochastic branch-and-bound* procedure to solve the due-date problem. Laslo [17] employed the *fractile method* to construct time–cost curves for a single activity with a stochastic duration distributed according to a known probability distribution. However, all of these models adopt an expected value approach, and hence, none of them yields an operational policy because the values given by the optimal solution under expectation would not necessarily be optimal for a random realization of the project. To be worthwhile, such a policy should be *adaptive* by specifying, at each decision epoch, the best value to select given the current status of a project. These policies should also be *robust* by guaranteeing that the project will be delivered on time under any realization of the durations for the activities that are not yet complete. Closing the gap between the existing research and these desired properties for the policies is the motivation of the present article.

3. THE STOCHASTIC, LINEAR, TIME–COST TRADEOFF PROBLEM

Consider the following LTCTP model where the cost is formulated as in (1). Substituting $t_{ij} = T_{ij} - y_{ij}$ and assuming c_{ij} is an affine function of t_{ij} we get:

LTCTP

$$\min \quad \sum_{ij} (c_{N,ij} + b_{ij} y_{ij}) + C x_n \qquad (2)$$

subject to

$$x_j - x_i + y_{ij} \ge T_{ij} \qquad \forall j, \forall i \in P_j$$

$$y_{ij} \ge 0 \qquad \forall j, \forall i \in P_j$$

$$y_{ij} \le T_{ij} - M_{ij} \qquad \forall j, \forall i \in P_j$$

$$x_1 = 0$$

$$x_n \le D$$

where x_i is the start time of node $i \in \{1, 2, \ldots, n\}$ and n is the final node (hence, if $x_1 = 0$, then x_n represents the project's duration), P_j is the set of immediate predecessors of node j, $c_{N,ij}$ is the *normal cost* of activity $\{ij\}$ associated with its *normal duration* T_{ij}, and b_{ij} is the *marginal cost* of crashing activity $\{ij\}$ by a single time unit. *Stochasticity* is introduced by the following assumptions:

1. The *normal duration* for each activity $\{ij\}$ is uncertain and bounded within a known interval $T_{ij} \in [T_{ij}^-, T_{ij}^+]$, $\forall j$, $\forall i \in P_j$ where T_{ij}^-, T_{ij}^+ can be interpreted as the optimistic and pessimistic estimates, respectively, for the normal duration. Note that we do not assume that the normal duration follows a known probability distribution. The interval can be written also as $T_{ij} \in [T'_{ij} - \varepsilon_{ij}^-, T'_{ij} + \varepsilon_{ij}^+]$, $\forall j$, $\forall i \in P_j$, where T'_{ij} is the estimated mean normal duration and ε_{ij}^-, ε_{ij}^+ are the uncertainty measures associated with the optimistic and pessimistic estimates, respectively. Finally, the duration of each activity is bounded from below by M_{ij}, which is usually rigid (e.g., a technological constraint). Hence, the inequality $T'_{ij} - \varepsilon_{ij}^- \ge M_{ij}$ (which limits the maximal possible value of ε_{ij}^-) must hold.
2. The *normal cost* of an activity is an uncertain, nondecreasing function of its uncertain normal duration denoted here as $\tilde{c}_{N,ij}$. We assume further that $\tilde{c}_{N,ij}$ is an affine function, $\tilde{c}_{N,ij} = c'_{N,ij} + q_{ij}(T_{ij} - T'_{ij})$, where $c'_{N,ij}$ is the baseline cost associated with T'_{ij} and q_{ij} is a predetermined (most often in the project contract) penalty/discount per unit time of deviations from T'_{ij}. For example, the contract may define q_{ij} as the compensation to the contractor if the scope of work is larger than expected or the award to the customer if the scope is smaller than expected [Asymmetric cases (i.e., different penalties for earliness and lateness) are also possible. But, for the sake of simplicity, we present here just the symmetric case.] Logic dictates that $q_{ij} \ge 0$. Cases where $q_{ij} = 0$ might happen when the baseline costs of the activities are fixed regardless of their actual durations.
3. The *marginal cost* b_{ij} is known and constant for every realization of T_{ij}. For example, consider the activity "reinforcing the upper fuselage structure" in an aircraft refurbishment project. Suppose that we crash the activity by allocating more man-hours to it, and its normal duration, $T_{ij} \in [T_{ij}^-, T_{ij}^+]$, is stochastic due to the possible need to repair structural damage caused by cracks, corrosion, etc., that might be revealed during that activity. This uncertainty does not affect the technological requirements for the activity nor does it change the way it is crashed, and hence, it is quite reasonable to assume that b_{ij} does not change.

Figure 1 illustrates the time–cost curves for SLTCTP, based on these three assumptions.

The implementation of the assumptions stated above in (2) leads to the following *family* of infinitely many deterministic optimization problems (where constants have been omitted from the objective):

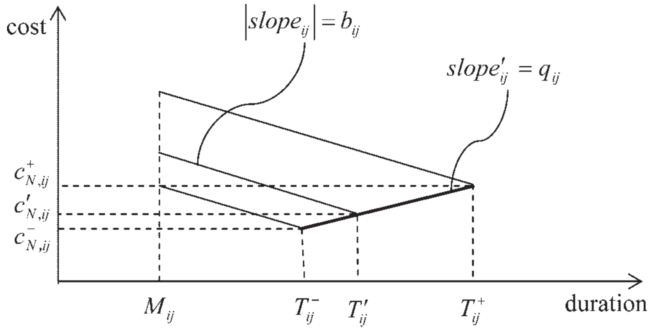

FIG. 1. Activity time–cost curves for SLTCTP.

SLTCTP

$$\left\{\begin{array}{lll} \min & \sum_{ij} q_{ij}T_{ij} + \sum_{ij} b_{ij}y_{ij} + Cx_n & \\ \text{subject to} & x_j - x_i + y_{ij} \geq T_{ij} & \forall j, \forall i \in P_j \\ & y_{ij} \geq 0 & \forall j, \forall i \in P_j \\ & y_{ij} \leq T_{ij} - M_{ij} & \forall j, \forall i \in P_j \\ & x_1 = 0 & \\ & x_n \leq D & \end{array}\right\} \quad \forall\, T_{ij} \in [T^-_{ij}, T^+_{ij}], \quad (3)$$

Note that the decisions in problem (3) amount to determining, for each activity $\{ij\}$, the crashing level y_{ij} and its start time x_i. In contrast to other research done in this area, which focuses on analyzing the throughput time of the project and the probability of finishing it on time (e.g., [8]), this article aims at optimizing the project's cost by making the optimal crashing decisions.

We shall make the decisions (i.e., level of crashing and starting time) for an activity $\{ij\}$ based upon information that is contained within certain *information sets,* I_{ij}, I_i, respectively. The most trivial assumption is that the information sets include only the past, known, durations of predecessor activities t_{kl}, $\forall\ \{kl\} \in P'_i$, where P'_i is the set of all the activities that precede node i. In some cases, we may expand the information sets to include also future durations. For example, suppose that an ad hoc evaluation is conducted just before the realization of node i and its outcome yields a fairly solid estimates on the durations of the immediate successors of node i. In this case, the estimated duration of activity $\{ij\}$ will be included in I_{ij}.

4. AARC FORMULATION OF THE SLTCTP

We begin with a brief background on the fundamentals of RO and then specialize it for problem (3). The reader is referred to Ben-Tal and Nemirovski [7] for a detailed presentation of the methodology and an up-to-date survey of its implementations.

4.1. The RO Methodology

The SLTCTP given by (3) is an instance of an *uncertain LP,* that is, a family of deterministic LPs:

$$\left\{\min_{\nu}\{w^T\nu | A\nu \leq s\}\right\}_{\upsilon \in (A,w,s) \in U} \quad (4)$$

where the data A, w, s may vary within a given uncertainty set U. The definition of U requires some effort from the user of the model. But, compared with other techniques (e.g., SPR), it requires less effort because specifying the boundaries of the region in which the data can vary is easier than characterizing the behavior of the data inside that region (through, e.g., density functions).

RO associates with the uncertain LP in (4) a Robust Counterpart (RC), which is presented as the following deterministic problem:

$$\min_{Z,\nu}\{Z | w^T\nu \leq Z, A\nu \leq s,\ \forall\ \upsilon \equiv (A,w,s) \in U\}. \quad (5)$$

The RC model (5) is a semi-infinite LP (i.e., with infinitely many linear constraints); thus, it seems computationally intractable. Nevertheless, it is shown to be solvable in polynomial time for a wide variety of convex uncertainty sets U. Moreover, the RC model is a convex mathematical problem, usually an LP or a conic quadratic problem [7]. A solution to (5), (Z^*, ν^*), is called *robust* because it satisfies the constraints for *all* possible realizations of the data in U. Such a solution also guarantees that the optimal objective value for any realization of the data will not be worse than Z^*. The optimal decision vector ν^* is fully determined before any realization of the data occurs; thus, it cannot be adjusted. However, the quality of the solutions would improve if decisions will be (at least partially) based on previous realizations of the data. For uncertain LPs with the above features, Ben-Tal and Nemirovski [7] developed the *Adjustable Robust Counterpart* (*ARC*) formulation:

$$\left\{\min_{Z,r}\{Z |\ \forall\ \upsilon \in U\colon \exists\ \nu\colon a^T\nu + g^Tr \leq Z, A\nu + Gr \leq s\}\right\}, \quad (6)$$

where ν represents the adjustable ("wait and see") decision variables and r represents the nonadjustable ("here and now") decision variables.

Clearly, the ARC model (6) is more flexible than the RC model (5), while still satisfying the constraints for all possible realizations of the data $\upsilon = (A, G, a, g, s) \in U$. The feasible set is larger, thus leading to a potentially better value for the objective function. However, the price for this added flexibility is that (6) is often computationally intractable (NP-hard) even for simple uncertainty sets (e.g., a general polyhedral set). Ben-Tal et al. [3] overcame the above intractability by restricting the adjustable decision vector ν to be *linear decisions rules* (LDR) of the uncertainty vector υ

$$\nu = L\upsilon + \lambda, \quad (7)$$

for some matrix L and vector λ with the appropriate dimensions. LDR's are quite common heuristics in many branches of science and engineering (e.g., linear feedback in control theory, linear estimators in signal processing). Applying the LDR (7) to the ARC model (6) transforms it into an *Affinely Adjustable Robust Counterpart* (AARC) model:

$$\begin{aligned}\min_{Z,r,L,\lambda}\quad & Z\\ \text{subject to}\quad & a^T(L\upsilon+\lambda)+g^Tr\le Z \quad \forall\,\upsilon\in U\\ & A(L\upsilon+\lambda)+Gr\le s \quad \forall\,\upsilon\in U\end{aligned}$$

The minimal objective value Z^* is "optimal" in the sense that no other solution to (6), with an adjustable vector ν that obeys the linear decision rule (7), can produce a better value for the objective function while satisfying the constraints for all possible realizations.

4.2. Applying the AARC Model to the SLTCTP

We start by defining the LDRs to substitute for the decision variables x_i and y_{ij} in (3):

$$x_i=\xi_i^0+\sum_{kl\in I_i}\xi_i^{kl}T_{kl} \tag{8}$$

$$y_{ij}=\eta_{ij}^0+\sum_{kl\in I_{ij}}\eta_{ij}^{kl}T_{kl}$$

where the coefficients ξ_i^0, η_{ij}^0, ξ_i^{kl}, η_{ij}^{kl} are the new decision variables and I_{ij}, I_i are the relevant information sets. We treat here all the decision variables as adjustable, but we can easily leave some variables as "here and now" decisions simply by not applying the LDRs (8) to them. Note that the initial uncertainty in t_{ij} originates from the variation in T_{ij} and that there is a linear relation between T_{ij}, t_{ij} and y_{ij}. This justifies the relations between y_{ij} and x_i to the realizations of the normal durations of the predecessor activities as given by (8) above.

By applying (8) to problem (3) we obtain the AARC representation of the SLTCTP.

<u>AARC model of the SLTCTP</u>

$$\min_{z,\xi_i^0,\xi_i^{kl},\eta_{ij}^0,\eta_{ij}^{kl}} Z \tag{9}$$

subject to

$$\sum_{ij}q_{ij}T_{ij}+\sum_{ij}b_{ij}\left(\eta_{ij}^0+\sum_{kl\in I_{ij}}\eta_{ij}^{kl}T_{kl}\right)+C\left(\xi_n^0+\sum_{kl\in I_n}\xi_n^{kl}T_{kl}\right)\le Z \qquad \forall\,T\in U_1 \tag{9.1}$$

$$\xi_j^0+\sum_{kl\in I_j}\xi_j^{kl}T_{kl}-\xi_i^0-\sum_{kl\in I_i}\xi_i^{kl}T_{kl}+\eta_{ij}^0+\sum_{kl\in I_{ij}}\eta_{ij}^{kl}T_{kl}-T_{ij}\ge 0 \qquad \forall\,j,\forall\,i\in P_j,\quad \forall\,T\in U_2 \tag{9.2}$$

$$\eta_{ij}^0+\sum_{kl\in I_{ij}}\eta_{ij}^{kl}T_{kl}\ge 0 \qquad \forall\,j,\forall\,i\in P_j,\quad \forall\,T\in U_3 \tag{9.3}$$

$$\eta_{ij}^0+\sum_{kl\in I_{ij}}\eta_{ij}^{kl}T_{kl}-T_{ij}\le -M_{ij} \qquad \forall\,j,\forall\,i\in P_j,\quad \forall\,T\in U_4 \tag{9.4}$$

$$\xi_1^0=0 \tag{9.5}$$

$$\xi_n^0+\sum_{kl\in I_n}\xi_n^{kl}T_{kl}\le D \qquad \forall\,T\in U_5 \tag{9.6}$$

where the U_r, $r = 1, 2, \ldots, 5$, are the uncertainty sets associated with the constraints.

The next step in applying AARC is to select an uncertainty set for each one of the constraints in (3). In principle, we may choose a different type of uncertainty set for each constraint. In this article, we formulate the model with either *interval-type* or *ellipsoidal-type* uncertainty sets for all the constraints. These types of uncertainty sets are discussed at length in [6] where they were shown to allow computationally tractable solutions for various problems.

A natural interpretation to our selection of these uncertainty sets is as follows. By selecting interval-type uncertainty sets we take a conservative approach, making sure that the constraints will hold for any realization of the activities, even for the most unfavorable case. On the other hand, ellipsoidal-type uncertainty sets allow a less conservative approach by eliminating realizations where all the activities' durations are at their most unfavorable values. The level of conservatism may be adjusted, as we will demonstrate shortly. In the next two subsections we provide some additional details.

4.2.1. Deriving a Computationally Tractable Problem with Interval-Type Uncertainty Sets.

First, we define an interval-type uncertainty set and then we derive an optimization problem that is equivalent to the AARC formulation (9). First, for practical reasons, it is easier to work with only one uncertainty measure ε_{ij} per activity, rather than the two original measures, ε_{ij}^- and ε_{ij}^+. To do so, we transform the original interval $T_{ij}\in[T_{ij}'-\varepsilon_{ij}^-, T_{ij}'+\varepsilon_{ij}^+]$ into an equivalent interval $T_{ij}\in[T_{ij}^*-\varepsilon_{ij}, T_{ij}^*+\varepsilon_{ij}]$ where $T_{ij}^*=(T_{ij}^-+T_{ij}^+)/2$ and $\varepsilon_{ij}=(\varepsilon_{ij}^-+\varepsilon_{ij}^+)/2$. Note that the associated normal cost is now $c_{N,ij}^*=c_{N,ij}'+q_{ij}(T_{ij}^*-T_{ij}')$ which is a known figure because $c_{N,ij}^I$, T_{ij}^-, T_{ij}^+, T_{ij}' and q_{ij} are all given as data. Also, $T_{ij}^*-\varepsilon_{ij}\ge M_{ij}$ must hold.

Thus, an interval-type uncertainty set is $U_{\text{int}}=\{T\equiv(T_1, T_2,\ldots,T_m)^T\in\Re_+^m : |T-T'|\le\varepsilon\}$, where ε and T^* are the vectors associated with the ε_{ij}'s and the T_{ij}^*'s, respectively.

Next, we derive the optimization problem. We explain this derivation through an example that represents the required transformation for all the constraints.

Consider the transformation of constraint (9.3):

$$\eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T_{kl} \geq 0 \tag{10}$$

We call this constraint "typical" because we can easily transform all the *uncertain* constraints (that depend upon the uncertain normal durations) to a similar form by introducing new decision variables that are affine combinations of the original ones.

For the interval-type uncertainty, inequality (10) is equivalent to

$$\min_{T} \left\{ \eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T_{kl} \middle| |T_{ij} - T_{ij}^{*}| \leq \varepsilon_{ij}, \quad ij = 1,2, \ldots ,m \right\} \geq 0 \tag{11}$$

The optimal solution to (11) is:

$$\eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T_{kl}^{*} - \sum_{kl \in I_{ij}} |\eta_{ij}^{kl}| \varepsilon_{kl} \geq 0 \tag{12}$$

Inequality (12) can be further expressed as the following linear system of inequalities:

$$\begin{cases} \eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T_{kl}^{*} - \sum_{kl \in I_{ij}} \tilde{\eta}_{ij}^{kl} \varepsilon_{kl} \geq 0 \\ -\tilde{\eta}_{ij}^{kl} \leq \eta_{ij}^{kl} \leq \tilde{\eta}_{ij}^{kl} \end{cases}$$

4.2.2. Deriving a Computationally Tractable Problem with Ellipsoidal-Type Uncertainty Sets. In some cases we may assume that the uncertainty affecting each one of the activities is independent of the other activities. In these cases, the *interval-type uncertainty model* seems "too conservative," as it immunizes the solution against the pernicious situation where all the data reach their most unfavorable values simultaneously. The ellipsoidal model of uncertainty $U_e = \{T \in \Re_{+}^{m} : (T - T')^T S^{-1}(T - T') \leq \Omega^2\}$ was developed especially to treat these situations.

In this model, S^{-1} is a $m \times m$ symmetric positive definite matrix, T' is a vector of the expected normal durations and $\Omega \geq 0$ is a safety parameter. When the actual durations are stochastic, a natural selection for S is the covariance matrix, i.e., $S = \mathrm{cov}(T)$.

Next, we derive the optimization problem. As before, we take constraint (9.3) to demonstrate the transformation to an equivalent tractable constraint. Assume that the normal durations are taken in U_e according to some probability function. We may express the left-hand side of the constraint as

$$a \equiv \bar{a} + \bar{\omega} = \eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T_{kl}$$

where $\bar{a}$ is the nominal value and $\bar{\omega}$ is a random perturbation with zero mean. Thus, we can treat a as a random variable with expected value

$$\eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T'_{kl}$$

and standard deviation

$$\sqrt{\sum_{\forall kl, \forall op \in I_{ij}} \eta_{ij}^{kl} \eta_{ij}^{op} S_{kl,op}}.$$

Now, adopting the conventional engineering notion that the value of a random variable (practically) never exceeds its mean plus three times the standard deviation (e.g., see [2]), we present the equivalent, "safe" version of the constraint as

$$\eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T'_{kl} - \Omega \sqrt{\sum_{\forall kl, \forall op \in I_{ij}} \eta_{ij}^{kl} \eta_{ij}^{op} S_{kl,op}} \geq 0.$$

If we assume that the normal durations $T_1, T_2, \ldots, T_m$ are independent random variables realized in the interval $[T'_{ij} - \varepsilon_{ij}^{-}, T'_{ij} + \varepsilon_{ij}^{+}]$, then to remain conservative, we must choose our uncertainty measure $\varepsilon_{ij} = \max(\varepsilon_{ij}^{+}, \varepsilon_{ij}^{-})$, $\forall ij = 1, 2, \ldots, m$, and the covariance matrix is then a diagonal matrix with entries $(\varepsilon_{ij})^2$, $\forall ij = 1, 2, \ldots, m$. In this case, this constraint is reduced to

$$\eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T'_{kl} - \Omega \sqrt{\sum_{\forall kl \in I_{ij}} (\eta_{ij}^{kl} \varepsilon_{ij})^2} \geq 0, \tag{13}$$

which is the same result obtained by using the Karush-Kuhn-Tucker conditions on the following optimization problem:

$$\min_{T} \left\{ \eta_{ij}^{0} + \sum_{kl \in I_{ij}} \eta_{ij}^{kl} T_{kl} \middle| (T - T')^T S^{-1} (T - T') \leq \Omega^2 \right\} \geq 0.$$

Ben-Tal and Nemirovski [6] proved that the probability p of violating constraint (13), for any symmetric probability distribution, is bounded by $p \leq \exp\{-\Omega^2/2\}$. For example, with $\Omega = 3$ the bound is $p \leq e^{-4.5}$, approximately 1%. It follows that when we have N uncertain constraints, the probability for a robust solution to be infeasible for the true (randomly perturbed) system of constraints is at most $N \exp\{-\Omega^2/2\}$. Thus, when Ω is fixed and N becomes large, the "reliability guarantees" deteriorate, and hence, it is reasonable to choose Ω according to $N \exp\{-\Omega^2/2\} = p_N$, with appropriately small p_N. Note, however, that the dependence of Ω on N is very rather weak (e.g., for $p_N = 0.01$ and for $N = 10$, 100, 10,000, the Ω values are 3.72, 4.29, and 5.26, respectively). Thus, in theory we can handle with a moderate Ω tens of thousands of constraints, still ensuring fairly high reliability. The actual choice depends, on one hand, on how rapidly the robust optimal value grows with Ω, and, on the other hand, on how small p_N is. The "actual reliability" of the resulting decision rule can be tested by Monte Carlo simulations.

5. A NUMERICAL EXAMPLE

In this section we demonstrate the effectiveness of the AARC model applied to the SLTCTP. First, we provide some details on the experimental environment and the basic project network. Then we continue with an overview of the experimental design and the computational results.

5.1. The Experimental Environment

The AARC model (9) was solved with the commercial code MOSEK (http://www.mosek.com). The instances that were solved involve thousands of variables and constraints. To avoid the burden of manually preparing the model for MOSEK, a program was written in Matlab to get the input data (e.g., the project data and additional parameters such as the uncertainty level, uncertainty set and the information sets) and prepare the model for MOSEK. After the model was solved, resulting in the optimal decision rules, we applied these rules to a stochastic network that was simulated by another Matlab program. Random activity durations were drawn from a chosen probability distribution.

For each one of the *cases* (we define a case as a unique combination of the following parameters: uncertainty level, information set, uncertainty set, due date, network type and the chosen probability distribution for the activities' normal durations), we generated $g = 10$ independent simulation runs. Then we calculated an α-*level* (we used $\alpha = 0.05$) *confidence interval* (CI), given by

$$\bar{S} \pm t_{g-1,1-\alpha/2} \frac{\hat{\sigma}}{\sqrt{g}},$$

where

$$\pm t_{g-1,1-\alpha/2} \frac{\hat{\sigma}}{\sqrt{g}}$$

is the *confidence interval variation* (CIV), $\bar{S}$ is the estimator for the optimal cost (based on the average of 10 independent solutions) and $\hat{\sigma}^2$ is the estimator for the variance.

$$\hat{\sigma}^2 = \frac{\sum_{r=1}^{g} (S_r - \bar{S})^2}{g - 1}.$$

We used the coefficient of variation,

$$CV = \frac{\hat{\sigma}}{\bar{S}} \cdot 100$$

in percentages, as the measure of variability.

All the experiments were conducted on a PC platform with an Intel Pentium 4 1.8GHz processor. The CPU time needed to prepare the model, solve it, and simulate it was relatively short—about 30 seconds per 10 replications of each case. Thus computation time was not an issue.

5.2. Description of the Project Network

We applied the AARC model to an AOA network that represents a *multiproject program* composed of three subprojects. The structure of the network and its characteristics (i.e., deterministic durations, costs of crashing, etc.) were taken from [23]. Project A has 11 activities, Project B has 15, and Project C has 12. The nodes are labeled with an alpha-numeric designation where the letter represents the project and the number represents the particular node within the project (X_n is a virtual node representing the completion of all three subprojects). The structure of the network defines the precedence constraints and the overall completion of the program occurs when all its underlying projects are completed. Figure 2 provides details regarding the program structure. The corresponding durations and costs of crashing are given in the Appendix. The program carries an overhead cost per month of $C = 0.305$ (in 10,000$).

5.3. The Experimental Design

The main goal of the experiment is to demonstrate the performance of the suggested *robust policies* (RP) and compare them to other policies. However, as the existing literature does not offer any alternative policy for the SLTCTP, we chose to compare our results with what we term the *utopian policy* (UP), which is the optimal policy in hindsight. (It is also called "offline optimal" in the context of online algorithms.) That is, suppose that we are told, in advance, the values of the uncertain realizations of the normal durations for all the activities of the program $T_{ij} \in [T_{ij}^-, T_{ij}^+]$, $\forall j$, $\forall i \in P_j$. This would yield a problem with a known and deterministic normal duration $\tilde{T}_{ij}^* \in [T_{ij}^-, T_{ij}^+]$ for each activity (note that almost surely $\tilde{T}_{ij}^* \neq T_{ij}^*$) and an associated deterministic normal cost $\tilde{c}_{N,ij}^*$. The optimal solution for this LTCTP can easily be found and represents the best possible policy UP. In addition, we tested the consequences of applying a deterministic policy (DP) to a stochastic project. First, we solved an LTCTP using the nominal data. Then, we applied the optimal deterministic decisions, that is, optimal crashing levels, to a stochastic project network and examined the feasibility of the model with respect to meeting the projects' due-dates.

Our problem is fairly complicated, involving a stochastic environment with many external parameters (e.g., uncertainty level, due dates, etc.) with possibly an infinite number of levels for most parameters. In these circumstances, designing a full factorial or even a fractional factorial experiment may result in an unrealistic number of experiments even for a modest number of levels for each parameter. As

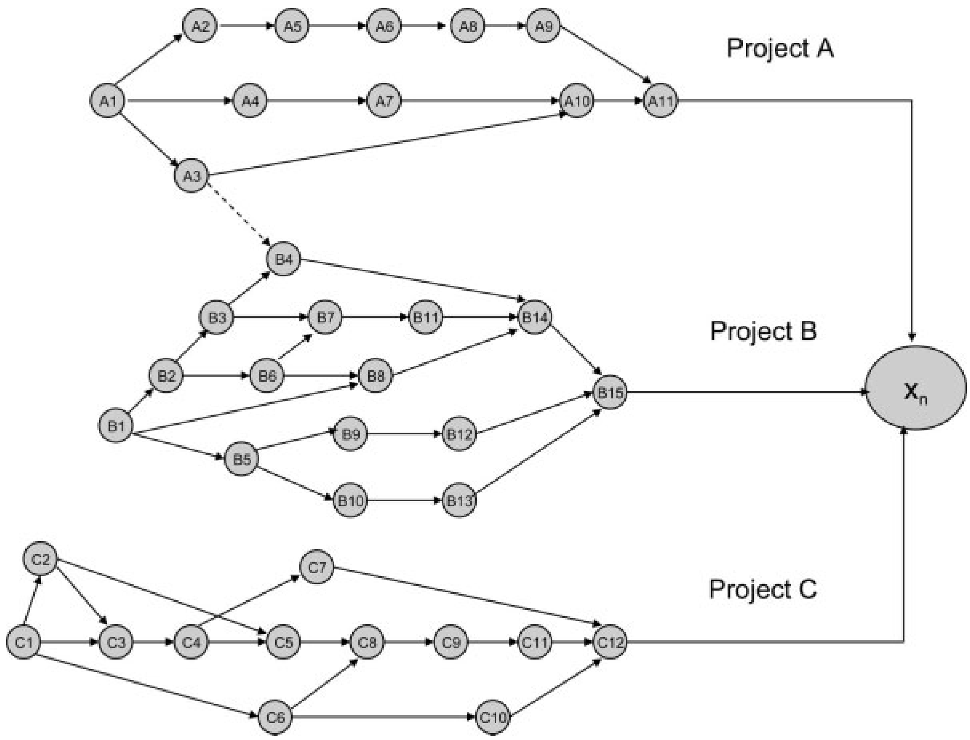

FIG. 2. The multiproject program described in [23].

the focus of this article is to introduce a new methodology and demonstrate its results we believe that a parametric experiment with respect to a predefined *base case* is the most appropriate design.

5.4. The Base-Case Description

The base case is given by Figure 2, and the data in the Appendix. For convenience, we defined R as a measure for the "looseness" of the due date with respect to the interval between the deterministic lowest bound (maximum crashing) and upper bound (no crashing). When $R = 0$, the due date equals the lowest bound, and when $R = 1$, its value is equal to the upper bound. The base case *due date* value was set to 84 months (which means $R = 0.25$), as in ([23], p. 497). The *uncertainty set* was chosen as an interval-type where we ensure that the constraints will hold even for the "worst-case" realizations. The *uncertainty level* for the base case was set to 0.7 (70% uncertainty)—which is a relatively large value when compared with the experiments in [2]. We define the uncertainty level of an activity as $\theta_{ij}^{\pm} = \varepsilon_{ij}^{\pm}/(T'_{ij}-M_{ij})$, where $\pm$ may be $+$ or $-$. We assume that at the beginning of each activity we have a fairly accurate estimate of its duration, and hence, the *information sets* I_{ij}, I_i include

TABLE 1. Comparing the performance of RP, UP, and DP with increasing levels of uncertainty.

Uncertainty level (%)	RP $\bar{S}$	RP CV (%)	UP $\bar{S}$	UP CV (%)	Price of robustness (%)	DP due-date violations (%)
5	2567.6	0.0	2547.8	0.2	0.8	80
10	2586.5	0.1	2549.3	0.3	1.5	90
15	2606.6	0.2	2553.5	0.4	2.1	90
20	2626.6	0.2	2561.9	0.8	2.5	100
25	2642.8	0.3	2553.7	1.0	3.5	80
30	2659.8	0.2	2559.5	1.3	3.9	90
40	2710.9	0.5	2565.7	1.8	5.7	90
50	2741.7	0.7	2559.5	1.6	7.1	80
60	2772.6	1.1	2567.7	2.0	8.0	90
70	2802.4	1.6	2577.2	2.1	8.7	80
80	2820.6	0.8	2580.8	2.0	9.3	90
90	2839.6	2.1	2585.4	3.0	9.8	80
100	2869.0	2.0	2627.8	3.5	9.2	100

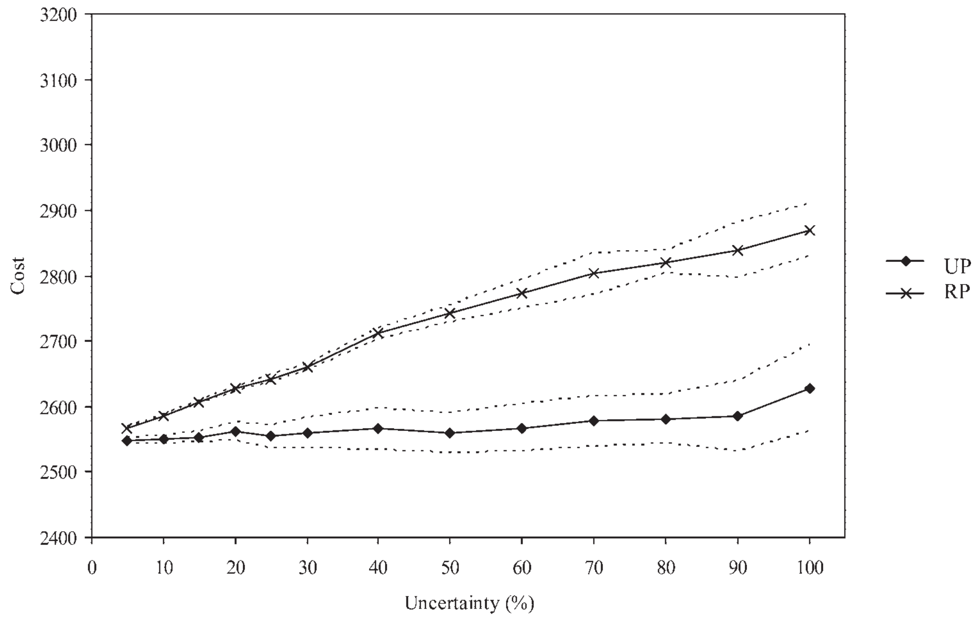

FIG. 3. Cost versus uncertainty for RP and UP.

information regarding past durations as well as the immediate successors of node *i*. The q_{ij}s were assumed to be zero, thus representing contracts where the baseline activity costs are fixed.

5.5. The Experiments

5.5.1. Experiment 1: Influence of the Uncertainty Level The uncertainty level was changed over the range of 5–100%. The uncertainty level may be higher than 100%, but we did not consider such cases in the experiment as they usually represent "one of a kind" projects where the individual activities are done for the first time and there is no valid time estimation for them.

The results are given in Table 1 and Figure 3. The price we pay for using RP is smaller than 10% (relative to the UP) for uncertainty levels up to 100%. It is interesting to note that the price of robustness is approximately linear with respect to the uncertainty level. As expected, the UP is relatively indifferent to the changes in the uncertainty level. Using the alternative DP has proved to be unfavorable as the due dates are violated in more than 80% of the cases.

5.5.2. Experiment 2: Influence of the Due-Date Values Due dates were changed according to R values in the interval [0, 1]. R values may exceed 1 as the stochastic project may finish beyond the upper bound for its deterministic counterpart but we did not include such cases in our experiment and preferred to examine due dates that require a larger amount of crashing.

Table 2 and Figure 4 demonstrate the increase in cost as the due-date looseness is decreased. For the tightest possible due date, $R = 0$, the RP fails to reach a feasible solution. However for $R \geq 0.05$, a feasible solution can be reached and for all the R values the price of robustness is below 10%. The curves in Figure 4 follow a general convex form. We suggest the following intuitive explanation for this convexity. RP and UP prefer to minimize costs by crashing activities on the critical path. For loose due dates, few critical paths exist. But, as due dates get tighter, many new critical paths may still exist even if one or more are eliminated through crashing. Consequently, the number of activ-

TABLE 2. Comparing the performance of rp, up, and dp with different due-date looseness values.

Due-date (R)	RP		UP		Price of robustness (%)	DP due-date violations (%)
	$\bar{S}$	CV (%)	$\bar{S}$	CV (%)		
69 (0)			Infeasible		—	100
72 (0.05)	3142.0	2.9	2957.0	3.8	6.3	100
75 (0.1)	3100.0	2.8	2880.6	4.2	7.6	100
84 (0.25)	2802.4	1.6	2577.2	2.1	8.7	80
99 (0.5)	2539.2	0.5	2470.0	0.7	2.8	70
114 (0.75)	2494.0	0.4	2444.4	0.6	2.0	80
129 (1)	2471.8	0.3	2427.7	0.4	1.8	80

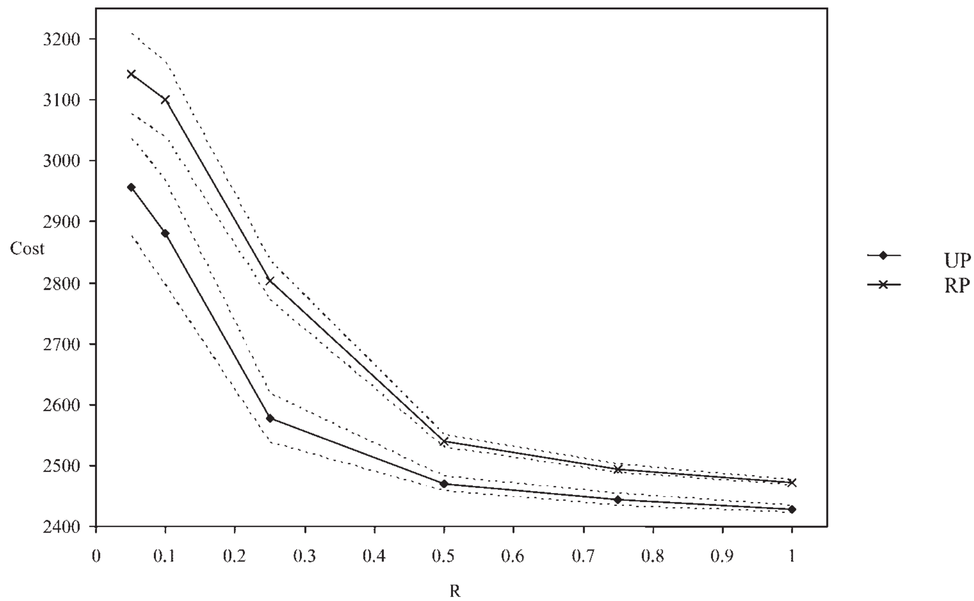

FIG. 4. Cost versus the due-date looseness (R) for RP and UP.

ities where crashing is required may increase quite drastically as due dates get tighter resulting in significantly higher costs.

5.5.3. Experiment 3: Influence of the Information Set Type We changed the base-case information set (denoted hereinafter as I^1) to include information regarding past durations only (denoted hereinafter as I^2). For tight due dates we expect that the problem, with I^2, will be infeasible even for relatively low levels of uncertainty. Hence, this experiment was done with a due-date value of 84 ($R = 0.25$) and repeated also with a due-date value of 99 ($R = 0.5$).

The influence of the information set is demonstrated through Table 3 and Figure 5. Although the UP costs are not significantly different for both information sets, the RP costs are significantly higher with the limited I^2. For example, using I^1 with $R = 0.5$ and uncertainty level of 25%, results in a reduction of 6.0% of project costs compared to the costs when using the limited I^2. As expected, we could not reach feasible solutions with I^2 even for intermediate levels of uncertainty (higher than 15% when $R = 0.25$ and higher than 30% when $R = 0.5$). This is understandable—recall that the lower bound for activity duration must hold for all normal activity realizations; thus, the amount of crashing is limited.

5.5.4. Experiment 4: Influence of the Network Topology and the Type of Uncertainty Set We used two types of uncertainty sets: an interval uncertainty set, and an uncertainty set that is an intersection between the interval and the ellipsoidal uncertainty sets. The latter enables us to control the level of conservatism in the model. For example, consider the two-dimensional cases depicted in Figure 6. The case shown on the left side of the figure, where the ellipsoid is completely contained in the rectangle, is basically an ellipsoidal type uncertainty set that is less conservative than the interval type because the extreme scenarios that occur near the four corners of the rectangle are not included. In the second case (on the right) the safety parameter increases, and hence, the ellipsoidal grows and, in fact, it goes beyond

TABLE 3. Comparing the performance of RP, UP, and DP under information set I^2 with various levels of uncertainty and two due-date values.

Due-date (R)	Uncertainty level (%)	RP $\bar{S}$	RP CV (%)	UP $\bar{S}$	UP CV (%)	Price of robustness (%)	DP due-date violations (%)
	5	2597.6	0.0	2546.0	0.2	2.0	80
84 (0.25)	10	2694.8	0.0	2553.3	0.4	5.5	100
	≥15	infeasible		—	—	—	—
	5	2477.2	0.0	2467.4	0.1	0.4	70
	10	2488.0	0.0	2467.9	0.1	0.8	70
	15	2500.0	0.1	2468.5	0.2	1.3	80
99(0.5)	20	2554.6	0.1	2467.8	0.2	3.5	80
	25	2639.3	0.1	2465.7	0.2	7.0	70
	≥30	infeasible		—	—	—	—

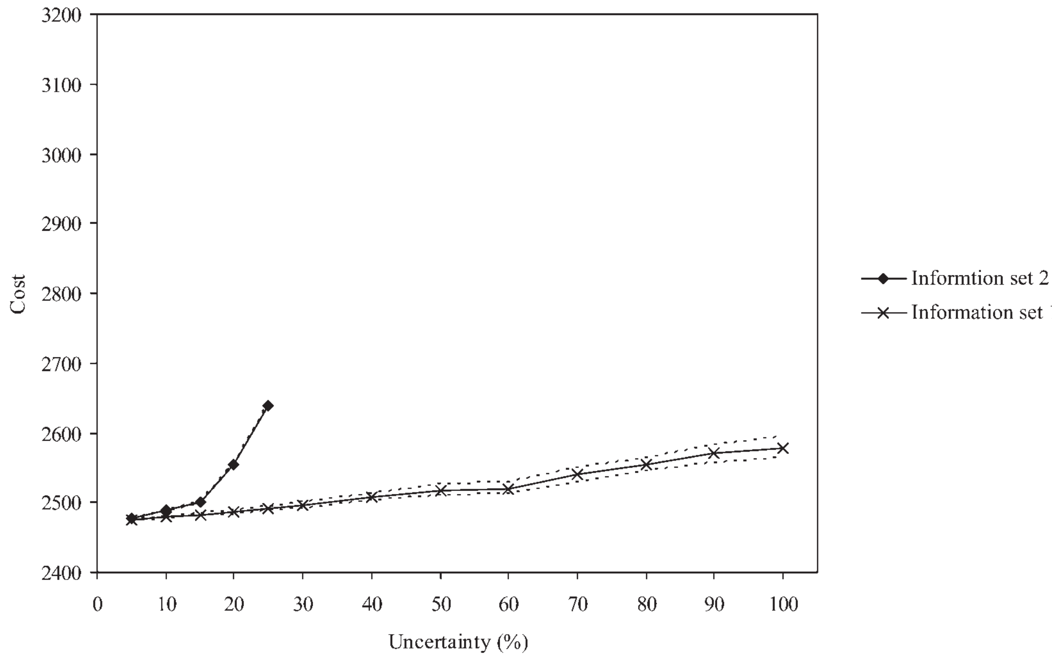

FIG. 5. Cost versus uncertainty for RP with two information sets I^1 and I^2 with due-date looseness, $R = 0.5$.

the boundaries of the rectangle. But because we take just the intersection of the two sets, we prohibit some realizations that would have been included in the original ellipsoidal set.

This experiment was conducted on both the base-case network and on a *serial network.* The serial network was constructed by arranging the original network activities in series according to the order given in the Appendix. The due date was set according to $R = 0.25$, similar to the base case. Following common engineering practice, the safety parameter for the ellipsoidal uncertainty sets was set to the conservative value of 3.

Through Table 4 we demonstrate that the use of an ellipsoidal-type uncertainty set leads to a reduction in the price of robustness even when conservative values are assigned to the safety parameter. We also notice that as the network topology becomes more serial (i.e., there are fewer predecessors per activity), the use of an ellipsoidal-type uncertainty set is more favorable. For example, for the serial network the price of robustness reduces by 53%, from 10.5% with an interval-type uncertainty set to 4.9% with an ellipsoidal-type uncertainty set. The corresponding reduction for the base-case network is 19.5%.

As expected, the project costs increased when the safety parameter was increased. For example, safety parameter values of 0.5, 1, 3, and 6 resulted in project costs of 2596.3, 2668.1, 2751.2, and 2813.2, respectively. The actual reliability was tested by a Monte Carlo simulation (over 100 simulations). The results suggested a reliability of 0.9948 for a safety factor of 0.5, and no violations of constraints for higher safety factors.

5.5.5. Experiment 5: Influence of the Probability Distribution's Shape We changed the shape of the probability distribution of the activity durations to be one of three common shapes: symmetric, skewed to the right and skewed to the left (see also [20], p. 312). Specifically, we achieved each one of the three shapes by using different parameter values in the beta probability distribution.

Table 5 presents the results. The price of robustness is about 10% for the beta probability distributions that are

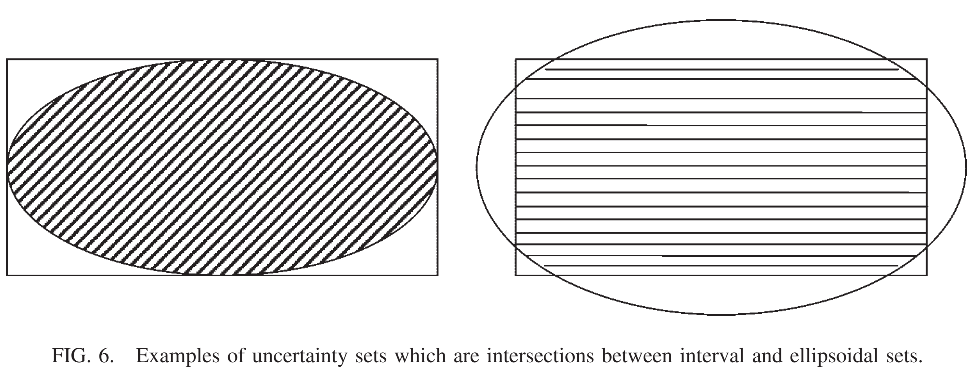

FIG. 6. Examples of uncertainty sets which are intersections between interval and ellipsoidal sets.

TABLE 4. Comparing the performance of RP, UP, and DP with interval- and ellipsoidal-type uncertainty sets for the base-case network and a serial network.

Network type	Type of uncertainty set	RP		UP		Price of robustness (%)	DP due-date violations (%)
		$\bar{S}$	CV (%)	$\bar{S}$	CV (%)		
Base-case	interval	2802.4	1.6	2577.2	2.1	8.7	80
	ellipsoidal	2751.2	1.3	2570.2	1.6	7.0	100
Serial	interval	3489.6	2.7	3157.3	2.7	10.5	50
	ellipsoidal	3264.3	2.4	3111.0	2.7	4.9	40

right skewed and symmetric and 6.5% for the left-skewed beta probability distribution. Naturally, the costs for the left-skewed distributions are the highest as the required crashing levels are the largest. As the probability distribution is more skewed to the left, the number of due-date violations caused by applying DP increases and vice versa.

5.6. An Extension: Folding Horizon Approach

The main motivation in using an adjustable model is to make better decisions by exploiting the information revealed up to the decision epoch. A very natural alternative to achieve the same result may be to solve a robust model initially and start the project. Then, we resolve a robust model at each decision epoch, using the information revealed up to that time. This approach may be described as a "folding horizon" [2].

We explored the folding horizon approach for the data in the base case. First, we solved the robust, nonadjustable model that resulted in an objective value that is 18.4% higher than the adjustable base-case model (3318.6 and 2802.4, respectively). Applying the folding horizon model yielded a cost of 2800.6, which is lower by about 0.1% compared to the base-case model (by the way, when we used an ellipsoidal uncertainty set, the cost of the adjustable model was 1.8% less than the corresponding folding horizon model). We believe that these results will hold in general—that is, we expect that the folding horizon model would give similar or even slightly better results compared to the adjustable model. However, in comparing the folding horizon model to the robust policies that were suggested earlier in this article, the folding horizon approach may prove to be very computationally demanding as it involves solving the problem a large number of times (in realistic projects it could be hundreds or even thousands of times).

6. CONCLUDING REMARKS

The main purpose of this article is to propose a new model, based on the AARC methodology, as a vehicle for determining adaptive, optimal, and robust solutions for the STCTP with continuous, linear time–cost curves and to demonstrate its potential benefits. To achieve this purpose, we first formulated a specific STCTP model and then applied the AARC principles to convert it into a computationally tractable problem that can be solved with commercial mathematical programming software.

Using an example taken from the literature, we conducted an extensive numerical experiment that generated important insights into the nature of the solutions we obtain with the proposed approach. First, we were able to get solutions that are immune to uncertainty levels that are up to 100% for a price which is higher than the best possible (utopian) solution by only 10%. This is a small price to pay in return for a guarantee that the solution will remain feasible even in the face of drastic changes in the data. If we are willing to accept rare cases of infeasibility, the price of the ellipsoidal-type model is even lower. Second, we showed that as the due date of the project becomes tighter the performance of the model deteriorates and in the extreme case when the due date is absolutely tight (i.e., a due date determined by CPM using the mean activity durations), we may not get a solution at all. However, as the due date becomes looser, the model yields the dividends described above. An important finding of the numerical experiment is the role played by the information sets. Significant reduction in costs can be achieved if we are able to get a good estimate of the length of an activity just as we start it.

We believe that the proposed approach has significant benefits over existing methods, as listed below:

TABLE 5. Comparing the performance of RP, UP, and DP with various probability distribution shapes for activity durations.

Probability distribution shape	RP		UP		Price of robustness (%)	DP due-date violations (%)
	$\bar{S}$	CV (%)	$\bar{S}$	CV (%)		
skewed to the right—Beta(1.5,3)	2739.2	0.8	2484.2	0.9	10.3	20
skewed to the left—Beta(3,1.5)	2860.0	1.1	2685.2	1.8	6.5	100
symmetric—Beta(3,3)	2802.7	0.6	2545.9	0.8	10.1	90

1. Decisions may be categorized into two groups: "here-and-now" decisions that must be taken at the beginning of the project (e.g., costs allocated for purchase of long lead items) and "wait-and-see" decisions that can be taken in a future time (e.g., allocation of costs that are associated with an available resource). "Wait-and-see" decisions can be adjusted to suit specific realizations in each project, thus avoiding unnecessary costs when possible.
2. Robust solutions: applying the AARC methodology generates a management policy that is immune to uncertainty, independent of the probability distribution type for the activity durations. The price of robustness in our experiments was fairly small—total immunity to uncertainty would cost up to 10% more than the utopian management policy for uncertainty levels up to 100%. We showed that excluding some rare events by using an ellipsoidal-type uncertainty set reduces the price of robustness even more. Moreover, the level of conservatism may be adjusted through different choices of the safety parameter.
3. User friendliness: the construction of our model may seem at first sight as a tedious task. However, this task can be easily automated as we discussed in Section 5.1. Then, we need to solve the problem just once to obtain the optimal parameters for the linear decision rules (LDR). The user, for example, the project manager, may utilize these LDRs throughout the project with minimal data requirements, such as only the durations of activities that were completed.
4. Computational tractability: AARC formulations are computationally tractable, and can be solved using commercial software. Even better, one can compare the quality of solutions with the utopian solutions.

Despite its benefits, our model is not without any limitations. Three that are worth mentioning are:

1. It does not take into account the effect that prior decisions have over future uncertainty sets. In some scenarios making some "good decisions" in the early stages of a project may reduce the uncertainty level of future activity durations. In these scenarios it could be beneficial to exploit such knowledge to reduce the value of the objective function.
2. It does not handle discrete or general nonlinear problems (apart from a conic objective function). Hence, it cannot be used to solve a discrete TCTP.
3. Our model assumes that resources can be easily obtained at short notice. Some project activities may require the reservation of resources well in advance. In such cases, the adjustable approach may be infeasible. One way to overcome such a difficulty is to add an artificial activity to represent the long lead time required for such activities.

Future research directions may include various extensions to our present work. For example, the examination of STCTP with nonlinear time–cost curves. Such research could employ the common piecewise linear approximation and then use the same approach taken here to solve the resulting model. The hope is that a small number of linear segments for each activity would provide sufficient accuracy in the results. Another possible research direction is the exploration of the DTCTP. Unlike the previous direction, that does not require any methodological extension of AARC itself, solving the DTCTP would require the enhancement of the AARC capabilities to handle integer variables.

Acknowledgments

We thank Professor Arkadi Nemirovski (Georgia Tech) and Professor Aharon Ben-Tal (The Technion, Israel Institute of Technology) for their valuable inputs to this work. Of course, the responsibility for any errors found in the article rests solely with the authors.

APPENDIX:

ACTIVITY INFORMATION FOR THE MULTIPROJECT PROGRAM DESCRIBED IN FIGURE 2

The durations T'_{ij} and M_{ij} are given in months, while the costs of crashing activity $\{ij\}$ by one month b_{ij} and the normal duration costs $c_{N,ij}$ are given in 10,000$.

i, j	Activity	T'_{ij}	M_{ij}	b_{ij}	$c_{N,ij}$	i, j	Activity	T'_{ij}	M_{ij}	b_{ij}	$c_{N,ij}$
1,2	A1,A2	3.0	1.3	15.0	71.8	18,22	B7,B11	5.3	0.5	2.3	27.7
1,4	A1,A4	4.4	1.9	2.0	15.1	19,25	B8,B14	27.4	10.8	4.5	35.2
1,3	A1,A3	20.2	14.5	17.4	73.3	20,23	B9,B12	14.0	8.0	2.4	95
2,5	A2,A5	2.8	1.7	11.1	63.1	21,24	B10,B13	18.1	11.3	1.3	17
4,7	A4,A7	11.1	7.6	1.8	19.5	22,25	B11,B14	32.7	14.6	6.3	62.7
3,10	A3,A10	26.0	21.0	1.5	17	23,26	B12,B15	21.0	6.0	1.3	69.5
3,15	A3,B4	0	0	0	0	24,26	B13,B15	15.4	0.9	0.4	37.6
5,6	A5,A6	32.8	25.0	23.7	99.4	25,26	B14,B15	3.4	1.0	0.6	25
6,8	A6,A8	27.8	23.6	16.9	73.8	27,28	C1,C2	20.3	18.4	5.2	75.7

i, j	Activity	T'_{ij}	M_{ij}	b_{ij}	$c_{N,ij}$	i, j	Activity	T'_{ij}	M_{ij}	b_{ij}	$c_{N,ij}$
7,10	A7,A10	8.1	6.3	12.6	71	27,29	C1,C3	32.5	17.9	11.3	94.4
8,9	A8,A9	13.8	13.8	0	42.2	27,32	C1,C6	24.2	13.1	4.6	36.9
9,11	A9,A11	5.9	3.7	15.8	83.6	28,29	C2,C3	2.7	0.1	1.5	53.8
10,11	A10,A11	1.5	0.5	10.3	45.6	28,31	C2,C5	19.2	9.0	23.1	95.8
12,13	B1,B2	27.1	17.4	1.7	52.2	29,30	C3,C4	25.8	21.8	1.1	34.5
12,19	B1,B8	35.4	12.6	0.7	60.1	30,31	C4,C5	16.2	2.3	1.8	39
12,16	B1,B5	2.5	0.8	8.1	42.5	30,33	C4,C7	18.9	8.4	6.0	87
13,14	B2,B3	25.3	7.6	0.3	4.6	31,34	C5,C8	9.6	0.1	1.5	8.1
13,17	B2,B6	9.0	0.1	0.6	33.7	32,34	C6,C8	21.2	17.0	7.5	64.3
14,15	B3,B4	9.1	0.4	7.3	82.4	32,36	C6,C10	28.7	17.0	5.9	61.6
14,18	B3,B7	25.5	21.9	5.1	21.7	33,38	C7,C12	2.4	0.9	3.2	23.4
15,25	B4,B14	32.7	24.4	7.5	40.5	34,35	C8,C9	9.5	5.9	1.0	8.1
16,20	B5,B9	6.2	2.8	11.1	49.4	35,37	C9,C11	6.8	3.3	1.3	9.2
16,21	B5,B10	24.8	16.3	3.4	68.5	36,38	C10,C12	2.4	0.7	12.1	73.3
17,18	B6,B7	3.3	2.5	3.2	20.8	37,38	C11,C12	28.8	13.6	1.5	17.2
17,19	B6,B8	15.6	12.3	9.8	85.6						

REFERENCES

[1] B.M. Baker, Cost/time trade-off analysis for the critical path method: A derivation of the network flow approach, J Oper Res Soc 48 (1997), 1241–1244.

[2] A. Ben-Tal, B. Golany, A. Nemirovski, and J.P. Vial, Retailer-supplier flexible commitments contracts: A robust optimization approach, Manufact Service Oper Manage 7 (2005), 248–271.

[3] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski, Adjusting robust solutions of uncertain linear programs, Math Program 99 (2004), 351–376.

[4] A. Ben-Tal, M. Kocvara, A. Nemirovski, and J. Zowe, Free material design via semidefinite programming, the multi load case with contact conditions, SIAM J Optimizat 9 (1999), 813–832.

[5] A. Ben-Tal and A. Nemirovski, Robust truss topology design via semidefinite programming, SIAM J Optimiza 7 (1997), 991–1016.

[6] A. Ben-Tal and A. Nemirovski, Robust solutions of linear programming problems contaminated with uncertain data, Math Program 88 (2000), 411–424.

[7] A. Ben-Tal and A. Nemirovski, A robust optimization—Methodology and applications, Math Program (Series B) 92 (2002), 453–480.

[8] D. Bertismas and A. Thiele, A robust optimization approach to supply chain management, Lecture Notes Comput Sci 3064 (2004), 86–100.

[9] P. Brucker, A. Drexl, R. Mohring, K. Neumann, and E. Pesch, Resource-constrained project scheduling: notation, classification, models and methods, Eur J Oper Res 112 (1999), 3–41.

[10] X. Chen, M. Sim, and P. Sun, A robust optimization perspective to stochastic models, working paper, Fuqua School of Business, Duke University (2004).

[11] R.F. Deckro, J.E. Herbert, W.A. Verdini, P.H. Grimsrud, and S. Venkateshwar, Nonlinear time/cost tradeoff models in project management, Comput Indust Eng 28 (1995), 219–229.

[12] W.J. Gutjahr, C. Strauss, and E. Wagner, A stochastic Branch-and-Bound approach to activity crashing in project management, INFORMS J Comput 12 (2000), 125–135.

[13] J.N. Hagstrom, Computational complexity of PERT problems, Networks 18 (1988), 139–147.

[14] W. Herroelen and R. Leus, Robust and reactive project scheduling: A review and classification of procedures, Int J Product Res 15 (2004), 1599–1620.

[15] F.S. Hillier and G.J. Lieberman, Introduction to operations research, 7th edition, McGraw-Hill, New York, NY, 2001.

[16] A. Kimms, Mathematical programming and financial objectives for scheduling projects, Kluwer Academic, Norwell, MA, 2001.

[17] Z. Laslo, Activity time–cost tradeoffs under time and cost chance constraints, Comput Indust Eng 44 (2003), 365–384.

[18] S. Leu, A. Chen, and C. Yang, A GA-based fuzzy optimal model for construction time–cost trade-off, Int J Project Manage 19 (2001), 47–58.

[19] O. Levinger, Robust linear programming with application to inventory management problems, Master's Thesis, Industrial Engineering and Management Faculty, Technion, Israel, 2002.

[20] A. Shtub, J.F. Bard, and S. Globerson, Project management: engineering, technology and implementation, Prentice Hall, Englewood Cliffs, NJ, 1994.

[21] C. van Delft and J.P. Vial, A practical implementation of stochastic programming: An application to the evaluation of option contracts in supply chains, Automatica 40 (2004), 743–756.

[22] J.D. Wiest and F.K. Levy, A management guide to PERT/CPM: With GERT/PDM/DCPM and other networks, Prentice-Hall, Englewood Cliffs, NJ, 1977.

[23] V.D. Wiley, R.F. Deckro, and J.A. Jackson, Optimization analysis for design and planning of multi-project programs, Eur J Oper Res 107 (1998), 492–506.