
Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

# Total Cost Control in Project Management via Satisficing

Joel Goh, Nicholas G. Hall,

MANAGEMENT SCIENCE
Vol. 59, No. 6, June 2013, pp. 1354–1372
ISSN 0025-1909 (print) | ISSN 1526-5501 (online)

http://dx.doi.org/10.1287/mnsc.1120.1653




# Total Cost Control in Project Management via Satisficing

Joel Goh
NUS Business School, National University of Singapore, Singapore 119245, joelgoh@nus.edu.sg

Nicholas G. Hall
Department of Management Sciences, Fisher College of Business, The Ohio State University,
Columbus, Ohio 43210, hall.33@osu.edu

We consider projects with uncertain activity times and the possibility of expediting, or crashing, them. Activity times come from a partially specified distribution within a family of distributions. This family is described by one or more of the following details about the uncertainties: support, mean, and covariance. We allow correlation between past and future activity time performance across activities. Our objective considers total completion time penalty plus crashing and overhead costs. We develop a robust optimization model that uses a conditional value-at-risk satisficing measure. We develop linear and piecewise-linear decision rules for activity start time and crashing decisions. These rules are designed to perform robustly against all possible scenarios of activity time uncertainty, when implemented in either static or rolling horizon mode. We compare our procedures against the previously available Program Evaluation and Review Technique and Monte Carlo simulation procedures. Our computational studies show that, relative to previous approaches, our crashing policies provide both a higher level of performance, i.e., higher success rates and lower budget overruns, and substantial robustness to activity time distributions. The relative advantages and information requirements of the static and rolling horizon implementations are discussed.

*Key words*: project management; time and cost control under uncertainty; robust optimization and satisficing; linear decision rule
*History*: Received January 3, 2011; accepted July 18, 2012, by Dimitris Bertsimas, optimization. Published online in *Articles in Advance* January 8, 2013.

## 1. Introduction

The use of project management as a planning methodology has expanded greatly in recent years. The Project Management Institute (2010) has seen its membership increase from 50,000 in 1996 to over 500,000 in 2010, in over 185 countries. Most of this increase is due to modern applications, for example, information technology, with substantial further growth potential (Pells 2008). Also, the use of shorter life cycles for products and services (Value Based Management .net 2009) is leading to increased application of project management in the development of new products and services. Finally, the pace of corporate change is increasing as a result of new technologies, more intense competition, and more demanding and less predictable customers (1000ventures.com 2009); achieving effective organizational change requires professional project management.

The two most important quantifiable performance measures in project management are completion time relative to a planned schedule and cost incurred relative to a budget (Might and Fischer 1985). Hence, project success is often defined as "completion of the project on time and on budget." Yet, many projects fail to achieve this benchmark. For example, only 35% of software development projects are completed on time (The Standish Group 2006). When late project completion is a possibility, one possible solution is the expediting, or *crashing*, of activity times. Crashing is usually accomplished through the commitment of additional resources, such as personnel, equipment, and budget, to individual activities of the project. This results in a financial trade-off between the project completion time and the cost of crashing (Klastorin 2004, Kerzner 2009). If activity times are deterministic or highly predictable, then crashing decisions can be determined using optimization models.

However, in most projects, random variations in activity times cause delays to the overall project schedule. Indeed, failure to compensate adequately for activity time variation is a prominent cause of project failure (Hughes 1986). Herroelen and Leus (2005) provided a comprehensive survey of various types of approaches to scheduling under uncertainty. A traditional, and still widely used (White and Fortune 2002), approach for planning projects in the presence of uncertainty in activity times is the Program Evaluation and Review Technique (PERT) (U.S. Navy 1958). The PERT model requires decision makers to estimate optimistic, pessimistic, and most likely durations for each activity. To build these random activity times into a simple probabilistic model of overall project completion time, PERT makes three important assumptions regarding the project's activities and its precedence network, as follows.

Assumption 1. *The uncertain activity times are probabilistically independent.*

Assumption 2. *A path with the longest expected length is still critical after realization of the activity times.*

Assumption 3. *Based on the central limit theorem, the total length of the critical path approximately follows the normal distribution.*

Each of these assumptions may be difficult to justify both in theory and in practice. First, if two activities use the same resources, then their time and cost performance is likely to be positively correlated (Hendrickson and Au 2000, Regnier 2005); hence, Assumption 1 is unlikely to hold. Second, it is frequently the case that a path that is not critical based on expected activity times becomes critical once activity times are realized (Williams 2003, Kwak and Ingall 2007); hence, Assumption 2 does not hold, and PERT typically underestimates project duration. Finally, if there are fewer than approximately 30 activities in series on the critical path, then the approximation using the central limit theorem in Assumption 3 is likely to be poor (Black 2008). Depending on the precedence structure, this may require several hundred activities in the project. Still more activities are required for accurate estimation of the tail probabilities of the critical path length distribution (Barbour and Jensen 1989).

An alternative to PERT that does not require Assumptions 1–3 is Monte Carlo simulation (MCSIM), which requires the use of a probability distribution to generate activity times. A justification for Monte Carlo simulation was provided by Schonberger (1981). However, specifying a distribution can be difficult (Tavares 2002). Indeed, Zammori et al. (2009, p. 279) wrote, "modelling an ill-known duration by a probability distribution requires a great amount of information, whereas in most practical cases just a little knowledge is available concerning the processing time of each task of the project." This limited knowledge results in part from the unique nature of projects. However, the consensus in the literature is that managers can typically estimate with reasonable accuracy at least three data points: the minimum, most likely, and maximum durations (Dawson and Dawson 1998). One approach to handling this limited information is to fit particular distributional forms to these three data points. The majority of the literature favors the beta or triangular distribution (Perry and Greig 1975, Chau 1995, Dawson and Dawson 1998, Ben-Yair 2003, Kerzner 2009). However, these three data points define a very broad class of distributions. Moreover, as our computational study demonstrates, an arbitrary choice of a distribution within this class may result in poor crashing decisions and project performance. Other objections to the use of Monte Carlo simulation focus on a lack of hardware and software support (White and Fortune 2002).

Both PERT and Monte Carlo simulation are typically used descriptively, rather than prescriptively, to inform project crashing decisions. However, Bowman (1994) applied infinitesimal perturbation analysis based on simulation results to construct a heuristic for making crashing decisions. Golenko-Ginzburg and Gonik (1997) described a zero-one integer program to prioritize activities. Golenko-Ginzburg and Gonik (1998) used a chance-constrained optimization model to control the project at discrete inspection points. Gutjahr et al. (2000) applied two sampling techniques and developed an overall stochastic branch and bound approach. Mitchell and Klastorin (2007) described a heuristic to minimize the total of crash cost, expected makespan, and expected tardiness penalty costs. All of these works assumed full knowledge of the probability distributions of the arc lengths, and also required Assumption 1. A wealth of both industry and academic evidence (for example, Adler et al. 1995, Leach 1999, van Dorp and Duffey 1999, Pender 2001, Williams 2003, Herroelen and Leus 2005, Cohen et al. 2007, Kerzner 2009) questions these two assumptions. Our prescriptive methodology does not rely on these assumptions.

Robust optimization was introduced by Soyster (1973) and popularized by Ben-Tal and Nemirovski (1998), El-Ghaoui et al. (1998), and Bertsimas and Sim (2004). Its primary function is to optimize systems that are subject to uncertainties with unknown distributions, which makes it useful for our problem. Also, Ben-Tal et al. (2004) tractably extended robust optimization problems to handle *adapted* decisions, also known as recourse decisions, by restricting the structure of the adapted decisions to linear rules. This also makes robust optimization a useful approach for our problem.

We are aware of only three groups of works that apply robust optimization to the time–cost tradeoff problem of project management. Cohen et al. (2007) minimized the worst-case total cost (crash cost plus linear project completion time cost) over all activity times within an uncertainty set. They applied linear decision rules (LDRs) for crashing decisions. Chen et al. (2008) studied the crashing problem as a two-stage robust optimization problem.

They modeled uncertainties with support, mean, and covariance information. There are two key distinctions with our work. First, our crashing decisions are adapted, whereas theirs are not. Second, they considered a two-stage uncertainty model, whereas our construction is inherently multistage. Wiesemann et al. (2012a, b) considered a model with a more general resource allocation mechanism for activities, based on production functions. They used a pathwise enumeration approach to circumvent a two-stage optimization. The former work used an uncertainty set, whereas the latter work considered uncertainties described by their first two moments.

Our work makes two distinct contributions to the literature. First, we propose a model that is closer to project management practice than the previously published works. Instead of using an expected-utility or total cost objective, we employ a satisficing objective relative to a budget (Brown and Sim 2009). Specifying a budget for a project more closely resembles how projects are managed in practice, compared to the unnatural alternative of estimating risk tolerance parameters to input into an expected-utility objective. Our satisficing approach penalizes both the incidence and extent of budget overrun, two key metrics of project performance in practice. Second, we derive piecewise-linear decision rules (PWLDRs) for activity start times and crashing quantities that adapt to revealed uncertainties. Furthermore, by using these decision rules (DRs), and by describing uncertainties through descriptive statistics such as their support, mean, and covariance, we model and exploit dependence between activity times.

In this paper, we develop satisficing models for the objective of completing the project on time and on budget. We consider total project cost, consisting of a project completion time penalty plus crash and overhead cost. Linear and linear-based decision rules (Ben Tal et al. 2004, Goh and Sim 2010) are used to solve this model. Because decision rules are functions of the uncertainties in the problem, they adjust decisions as the uncertainties become realized. The decision rules use realized information to find activity start time and crashing decisions that perform well against all possible scenarios of activity time uncertainty. We allow for the modeling of correlation between previous and future performance on a single activity, and between performance on previous and subsequent activities. Our crashing strategies are implemented in both static and rolling horizon modes. We show computationally that our crashing strategies provide a higher probability of meeting the overall project budget than crashing strategies based on PERT and Monte Carlo simulation. The relative advantages of the static and rolling horizon implementations are also discussed. We also identify several managerial insights from our results.

## 2. Problem Description

We define a project to be a finite collection of $N$ individual activities, including precedence relationships between them. The project completes when every activity has completed. The project manager faces the problem of completing the project on budget, where the total project cost includes completion time penalty cost, crash cost, and overhead cost. The project completion target, $\tau \in \mathbb{R}_+$, the project budget, $B \in \mathbb{R}_+$, and a vector representing the upper limits of crash quantities, $\mathbf{u} \in \mathbb{R}_+^N$, are fixed exogenous parameters. These parameters are typically set by senior management during project planning.

### 2.1. Notation

We denote a random variable by the tilde sign, for example, $\tilde{x}$. Bold lowercase letters such as $\mathbf{x}$ represent vectors, and bold uppercase letters such as $\mathbf{A}$ represent matrices. Given a matrix $\mathbf{A}$, its transposed $i$th row is denoted by $\mathbf{a}_i$, and its $(i, j)$th element is denoted by $A_{ij}$. We denote by $\mathbf{e}$ the vector of all ones, and by $\mathbf{e}^i$ the $i$th standard basis vector. In addition, $x^+ := \max\{x, 0\}$, and $x^- := \max\{-x, 0\}$. The same notation is applied to vectors, for example, $\mathbf{y}^+$, and $\mathbf{z}^-$, which denotes that the corresponding operations are performed componentwise. For any set $S$, we denote by $\mathbb{I}_S$ the indicator function on that set. Also, we denote by $[N]$ the set of positive running indices to $N$, i.e., $[N] = \{1, \ldots, N\}$, for some positive integer $N$. For completeness, we define $[0] := \varnothing$.

### 2.2. Model of Uncertainty

The source of uncertainty in our model is the vector of uncertain activity completion times $\tilde{\mathbf{z}}$. We model $\tilde{\mathbf{z}}$ as sample elements of the measurable space $(\mathbb{R}^N, \mathscr{B}(\mathbb{R}^N))$, where $\mathscr{B}(\mathbb{R}^N)$ represents the corresponding Borel $\sigma$-algebra. On this space, there is a true measure, $\mathbb{P}$, that assigns probabilities to events in $\mathscr{B}(\mathbb{R}^N)$. However, for the reasons discussed in the introduction, we assume that the decision maker does not know $\mathbb{P}$ precisely. It is known only that $\mathbb{P}$ belongs to some family of measures $\mathbb{F}$ that is fully parameterized by a triple of properties $(\mathscr{W}, \hat{\mathscr{W}}, \mathbf{\Sigma})$, as now described.

*Support.* $\mathscr{W} \subseteq \mathbb{R}_+^N$ contains the support of $\tilde{\mathbf{z}}$. We assume that $\mathscr{W}$ is convex, bounded, full dimensional, and a tractable conic representable set, that is, $\mathscr{W}$ is a set that can be represented, exactly or approximately, by a number of linear and/or second order conic constraints that is polynomial in $N$.

*Mean.* We denote by $\bar{\mathbf{z}}$ the mean of $\tilde{\mathbf{z}}$. The mean $\bar{\mathbf{z}}$ is itself uncertain and only takes values (with positive probability) in a set $\hat{\mathscr{W}} \subseteq \mathscr{W}$. The set $\hat{\mathscr{W}}$ is assumed to be convex, bounded, and tractable conic representable. This includes the case of a known mean, if $\hat{\mathscr{W}}$ is a singleton set.

*Covariance.* We denote by $\boldsymbol{\Sigma}$ the known covariance of $\tilde{\mathbf{z}}$.

Thus, $(\mathscr{W}, \hat{\mathscr{W}}, \boldsymbol{\Sigma})$ represent exogenous data to our model. However, note that it is not necessary that all of $(\mathscr{W}, \hat{\mathscr{W}}, \boldsymbol{\Sigma})$ are known. If $\mathscr{W}$ is unknown, we replace it with some large hyperrectangle in $\mathbb{R}_+^N$. Similarly, if $\hat{\mathscr{W}}$ is unknown, we take $\hat{\mathscr{W}} = \mathscr{W}$, because $\hat{\mathscr{W}} \subseteq \mathscr{W}$ in general. If the covariance matrix $\boldsymbol{\Sigma}$ is not known, we define $\mathbb{F}$ only in terms of $\mathscr{W}$ and $\hat{\mathscr{W}}$. Formally, we define the distributional families

$$\begin{aligned} &\mathbb{F}_1 := \{\mathbb{P}\colon\ \mathbb{P}(\tilde{\mathbf{z}} \in \mathscr{W}) = 1,\ \mathrm{E}_{\mathbb{P}}(\tilde{\mathbf{z}}) \in \hat{\mathscr{W}}\} \quad \text{and} \\ &\mathbb{F}_2 := \{\mathbb{P}\colon \mathrm{Cov}_{\mathbb{P}}(\tilde{\mathbf{z}}) = \boldsymbol{\Sigma}\}. \end{aligned} \tag{1}$$

The distributional family of uncertain activities times, $\mathbb{F}$, that we use is then

$$\mathbb{F} := \begin{cases} \mathbb{F}_1 \cap \mathbb{F}_2 & \text{if } \boldsymbol{\Sigma} \text{ is known,} \\ \mathbb{F}_1 & \text{otherwise.} \end{cases} \tag{2}$$

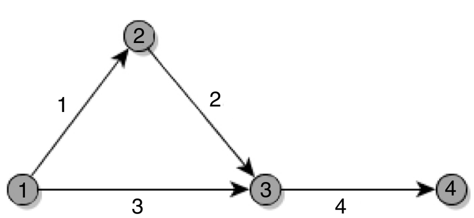
Henceforth, we adopt the notational shorthand that all (in)equalities involving uncertainties hold with probability one across all measures in $\mathbb{F}$. For example, we write $\tilde{z} \geq 0$ to mean $\mathbb{P}(\tilde{z} \geq 0) = 1$, for all $\mathbb{P} \in \mathbb{F}$. Given the above structure, this is equivalent to the collection of inequalities $z \geq 0$, $\forall z \in \mathscr{W}$.

### 2.3. Precedence Relationships

The precedence relationships between the activities of a project can be represented by a directed acyclic graph, or project network. Two widely used conventions for this network are the activity-on-node (AON) and activity-on-arc (AOA) formats (Klastorin 2004). In this paper, we adopt the AOA format. The project has $N$ activities; hence, the graph has $N$ arcs. We let $M$ represent the number of nodes in the graph. By introducing dummy activities, without loss of generality, we only consider graphs where no two arcs start and end at the same pair of nodes. For each activity $n \in [N]$, we let $i_n \in [M]$ represent the node from which this activity emanates, and $j_n \in [M]$ the node into which this activity points.

For example, in the project network illustrated in Figure 1, there are $M = 4$ nodes and $N = 4$ activities. For this graph, $(i_1, j_1) = (1, 2)$, $(i_2, j_2) = (2, 3)$, $(i_3, j_3) = (1, 3)$, and $(i_4, j_4) = (3, 4)$. Activity 2 cannot start until activity 1 has ended. Similarly, activity 4 cannot start until *each* activity 1, 2, and 3 is complete.

**Figure 1** A Small Example Project




### 2.4. Decisions

The project manager makes two sets of decisions. Taking into account the restrictions imposed by the precedence relationships, he chooses $\mathbf{x}$, with components representing the event time of each node of the project network, or, equivalently, the earliest start time of all activity arcs that emanate from that node. Also, he chooses $\mathbf{y}$, with components representing the amount of crashing for each activity.

Technically, $\mathbf{x}$ and $\mathbf{y}$ are *nonanticipative functionals* of the uncertainties that take values in $\mathbb{R}_+^M$ and $\mathbb{R}_+^N$, respectively. For now, it suffices to view $\mathbf{x}$ and $\mathbf{y}$ as vectors in $\mathbb{R}_+^M$ and $\mathbb{R}_+^N$, respectively.

For the decision $(\mathbf{x}, \mathbf{y})$ to be feasible, it is necessary that $\mathbf{0} \leq \mathbf{y} \leq \mathbf{u}$, and $\mathbf{x} \geq \mathbf{0}$. Also, for each $n \in [N]$, it is necessary that $x_{j_n} - x_{i_n} \geq (\tilde{z}_n - y_n)^+$. This set of constraints enforces the precedence relationship between activities. Thus, any activities that succeed activity $n$ can commence only after the (potentially expedited) activity time $(\tilde{z}_n - y_n)^+$ has elapsed. The $(\cdot)^+$ operator ensures that crashed activities have nonnegative activity times.

### 2.5. Information Flow

In typical projects, information about the progress of all ongoing activities within a project is updated periodically, for example, weekly. A standard methodology for reporting project progress is earned value analysis (U.S. Department of Defense 1962). Earned value analysis compares both the time used and the cost spent so far against the amount of progress that has been achieved in the project (Fleming and Koppelman 2005). The resulting differences are known as *time variance* and *cost variance*, respectively.

The uncertain activity times in our model are revealed over time by two mechanisms. First, whenever an activity is completed, its previously uncertain activity time becomes known. We use this information to form nonanticipative decision rules for crashing and start time decisions. Because these decision rules are functions of the revealed uncertainties, the actual decisions adjust to the already revealed activity times, as discussed in §4.1. This is the sole source of information for our proposed crashing policies when operated in static mode. Second, in rolling horizon mode, the project manager receives periodic updates by activity managers on the progress of each activity. We denote by $T$ the length of each time period between updates. At the end of each time period, time variance and cost variance are used to monitor the progress of each activity and the overall status of the project with respect to the schedule and budget. Each periodic update provides the manager with updated resource information (budget, schedule, and allowable crashing) and an updated uncertainty description. Using the updated information, the project manager may reallocate crashing resources between activities. We use this periodic information gain, as well as the activity completion time information described above, to solve the updated model, as discussed in §6.

## 3. Objective Function

As discussed in the introduction, the project manager faces the two, potentially conflicting, objectives of completing the project on schedule and within budget. We consider the composite objective of minimizing the total project cost, denoted by $\tilde{V}$, which we define to include completion time penalty cost, overhead cost, and crashing cost. We describe these three cost components in turn.

The first cost component is a contract penalty that is incurred when the project completion time, $\tilde{T}$, exceeds its deadline. We model this penalty as a piecewise-linear nondecreasing convex function, $p(\cdot)$, and represent it as $p(\tilde{T}-\tau) := \max_{k\in[K]}\{p_k(\tilde{T}-\tau)+q_k\}$, for some $K \in \mathbb{N}$, and exogenous parameters $\{p_k, q_k\}_{k=1}^K$. The monotonicity and convexity properties are typical of business practice (Fulkerson 1961, Kapur 1973, Kerzner 2009). The second cost component is an overhead cost component that is linear in the project completion time, $\tilde{T}$. Overhead cost models the general costs of the organization, such as administrative support, power, and security, which are incurred during the project and allocated to it (Klastorin 2004, Kerzner 2009). Without loss of generality, we rearrange terms and incorporate the overhead cost within $p(\cdot)$, for notational convenience. We explicitly discuss overhead cost because of its practical importance, even though we account for it analytically within the penalty cost. The third cost component is the sum of the crashing cost of each activity, which is assumed to be individually piecewise linear, nondecreasing, and convex. These properties are also typical of business practice (Kelley and Walker 1959), where each piece of the cost function represents a different type of crashing resource with its own linear cost rate. Formally, we denote the cost of crashing activity $n \in [N]$ by an amount $y$ as $c_n(y)$ and represent it as $c_n(y) := \max_{\ell\in[L]}\{c_{n\ell}y + d_{n\ell}\}$ for some $L \in \mathbb{N}$ and exogenous parameters $\{c_{n\ell}, d_{n\ell}\}_{\ell=1}^L$. Without loss of generality, we assume that each activity has the same number, $L$, of piecewise-linear components, that the parameters $p_k$ are nondecreasing in $k$, and $c_{n\ell}$ are nondecreasing in $\ell$ for each $n$. For both the first and third cost components, the piecewise-linearity assumption enables us to approximate arbitrary convex functions. Combining these cost components, the total cost $\tilde{V}$ is given by the expression

$$\begin{aligned}\tilde{V} &= p(\tilde{T}-\tau) + \sum_{n=1}^N c_n(y)\\ &= \max_{k\in[K]}\{p_k(\tilde{T}-\tau)+q_k\} + \sum_{n=1}^N \max_{\ell\in[L]}(c_{n\ell}y + d_{n\ell}).\end{aligned}$$

The project manager's objective function should incorporate two features. First, it should include all three cost components described above. Second, it should consider both the probability of project completion within the budget, and also the extent of tail risk above the budget. A class of objective functions that has these features and other useful properties is the class of quasiconcave satisficing measures (QSMs), introduced by Brown and Sim (2009). Brown et al. (2012) showed that generalizations of QSMs resolve some classical behavioral paradoxes that arise in expected utility theory (Allais 1953, Ellsberg 1961). The *conditional value-at-risk* (*CVaR*) *satisficing measure* is an example of a QSM and is defined as follows.

Definition 1. Given a random variable $\tilde{V}$ representing uncertain expenditure and a budget $B$, the *CVaR satisficing measure* is defined by

$$\rho_B(\tilde{V}) = \begin{cases}\sup\{\beta\in(0,1)\colon \mu_\beta(\tilde{V})\le B\} & \text{if feasible,}\\ 0 & \text{otherwise,}\end{cases}$$

where $\mu_\beta$ is the $\beta$-conditional value at risk (Rockafellar and Uryasev 2000) given by

$$\mu_\beta(\tilde{V}) = \inf_{v\in\mathbb{R}}\left\{v + \frac{1}{1-\beta}\mathrm{E}_{\mathbb{P}}(\tilde{V}-v)^+\right\}.$$

Our justifications for using the CVaR satisficing measure are as follows, and are proven by Brown and Sim (2009). First, if $\tilde{V}$ always falls below the budget, $B$, then $\mu_\beta(\tilde{V}) \le B$ for all $\beta\in(0,1)$, and hence $\rho_B(\tilde{V}) = 1$. Second, if $\tilde{V}$ always exceeds the budget, then $\mu_\beta(\tilde{V}) > B$, and hence $\rho_B(\tilde{V}) = 0$. Third, among all QSMs, $\rho_B(\tilde{V})$ is the largest lower bound for the success probability, $\mathbb{P}(\tilde{V}\le B)$. Fourth, the solution obtained by optimizing the satisficing level typically provides a high probability of project success. Our computational study in §7 verifies this point. In particular, if uncertainties are normally distributed, then optimal solutions found under the CVaR satisficing measure are also optimal under maximization of the probability of project success. Finally, if $\mathbb{P}(\tilde{V}<B) > 0$, then

$$\rho_B(\tilde{V}) = \sup_{\alpha\ge 0}\mathrm{E}_{\mathbb{P}}(\min\{1, \alpha(B-\tilde{V})\}) > 0. \tag{3}$$

This model reflects indifference toward costs that fall within the budget by a margin of at least $1/\alpha$, and increasing aversion toward costs that exceed the budget by larger amounts.

## 4. Robust Optimization Model

As discussed in the introduction, the project manager rarely has full knowledge of the probability distribution, $\mathbb{P}$. Because decision makers are typically averse to ambiguity (Ellsberg 1961), we evaluate the worst-case objective over the family of distributions $\mathbb{F}$ containing $\mathbb{P}$, that is,

$$Z_0 = \inf_{\mathbb{P}\in\mathbb{F}} \sup_{\alpha\geq 0} \mathrm{E}_{\mathbb{P}}(\min\{1, \alpha(B-\tilde{V})\}).$$

It is convenient to represent $Z_0$ in an alternative manner by reversing the order of the inf and sup operators. We show this in the following proposition, under mild conditions on $\mathbb{F}$.

Proposition 1. *If $\exists\, \mathbb{P}\in\mathbb{F}$ such that $\mathbb{P}(\tilde{V} < B) > 0$, and $\tilde{V}$ is bounded w.p.1 $\forall\, \mathbb{P}\in\mathbb{F}$, then*

$$Z_0 = \sup_{\alpha\geq 0} \inf_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(\min\{1, \alpha(B-\tilde{V})\}). \tag{4}$$

Proof. See the appendix.

The first assumption (on $\mathbb{F}$) would fail to hold only if all feasible crashing strategies have zero probability of meeting the budget, for all probability measures in $\mathbb{F}$. In this case, the project has to fail with probability 1 for all feasible crashing strategies, which is admittably extreme. The boundedness condition on $\tilde{V}$ follows from the boundedness of $\mathscr{W}$. The use of this objective function not only provides robust worst-case performance, but as our computational results in §7 confirm, also excellent typical performance.

### 4.1. Nonanticipative Decision Rules

Ideally, we would like the crashing and activity start time decisions to be DRs, i.e., measurable functions of the underlying uncertainties. As the project progresses, we impose *nonanticipativity* requirements on the DRs, which require that they depend only on previously revealed uncertainties. We model the nonanticipativity requirements using an information index set $I \subseteq [N]$. Each DR depends only on the components of the uncertainty vector with indices within the DR's corresponding information index set. Specifically, we define the parametrized set of functions

$$\mathscr{Y}(D, N, I) := \left\{ \mathbf{f}\colon \mathbb{R}^N \to \mathbb{R}^D\colon \mathbf{f}\left(\mathbf{z} + \sum_{i\notin I} \lambda_i \mathbf{e}^i\right) = \mathbf{f}(\mathbf{z}), \forall\, \lambda \in \mathbb{R}^N \right\}.$$

The first two parameters specify the dimensions of the codomain and domain of the functions, respectively, whereas the final parameter enforces the nonanticipativity requirement. For example, if we have $\mathbf{w} \in \mathscr{Y}(3, 4, \{2, 4\})$, then the DR $\mathbf{w}(\cdot)$ is a function characterized by $\mathbf{w}(\tilde{\mathbf{z}}) = \mathbf{w}(\tilde{z}_2, \tilde{z}_4), \forall\, \tilde{\mathbf{z}} \in \mathbb{R}^4$. Because 1 and 3 are not included in the information index set of this DR, we cannot use $\tilde{z}_1$ or $\tilde{z}_3$ in the DR.

The nonanticipativity requirements for the start time and crashing decisions are

$$x_m \in \mathscr{Y}(1, N, I_m^x), \quad \forall\, m \in [M],$$
$$y_n \in \mathscr{Y}(1, N, I_n^y), \quad \forall\, n \in [N],$$

where $\{I_m^x\}_{m=1}^M$, $\{I_n^y\}_{n=1}^N$ depend on the structure of the project network. To illustrate how information index sets are constructed from the project network, consider the small project in Figure 1. Node 1 has no preceding activities, and so $I_1^x = \varnothing$. Also, activities 1, 2, and 3 precede node 3; hence, $I_3^x = \{1, 2, 3\}$. The information index sets for crashing decisions are constructed analogously. Activity 2 is preceded by activity 1, and thus $I_2^y = \{1\}$. Activity 4 is preceded by activities 1, 2, and 3, and so $I_4^y = \{1, 2, 3\}$. We obtain the other information index sets analogously.

Using general DRs for robust optimization typically results in computational intractability (Ben-Tal et al. 2004). LDRs are a more tractable, but also more restrictive, class of decision rules that are affine functions of the uncertainties. We define this class of decision rules as the parametrized set of affine functions

$$\mathscr{L}(D, N, I) := \left\{ \mathbf{f}\colon \mathbb{R}^N \to \mathbb{R}^D\colon \exists\, (\mathbf{y}^0, \mathbf{Y}) \in \mathbb{R}^D \times \mathbb{R}^{D\times N}\colon \begin{array}{l} \mathbf{f}(\mathbf{z}) = \mathbf{y}^0 + \mathbf{Y}\mathbf{z} \\ \mathbf{Y}\mathbf{e}^i = \mathbf{0}, \forall\, i \notin I \end{array} \right\}. \tag{5}$$

Using LDRs, the nonanticipativity requirements take the explicit form of a set of linear equality constraints. For example, if $\mathbf{w} \in \mathscr{L}(3, 4, \{2, 4\})$, then the LDR $\mathbf{w}(\tilde{\mathbf{z}})$ can be represented as the affine function $\mathbf{w}(\tilde{\mathbf{z}}) = \mathbf{w}^0 + \mathbf{W}\tilde{\mathbf{z}}$, with the additional constraints that the first and third columns of the $3 \times 4$ matrix $\mathbf{W}$ must be zero. An equivalent representation is $\mathbf{w}(\tilde{\mathbf{z}}) = \mathbf{w}^0 + \tilde{z}_2\mathbf{w}^2 + \tilde{z}_4\mathbf{w}^4$. Observe that the actual numerical value of $\mathbf{w}(\tilde{\mathbf{z}})$ is known only after the uncertainties $(\tilde{z}_2, \tilde{z}_4)$ have been revealed.

### 4.2. Project Crashing Model

We assume without loss of generality that the project commences at node 1 and terminates at the final node $M$. Hence, the project completion time is $\tilde{T} = x_M(\tilde{\mathbf{z}})$. Because the uncertain expenditure $\tilde{V}$ in (4) is the sum of the total crashing and penalty costs, the project crashing model is

$$\begin{aligned}
Z_0^* := \max_{\mathbf{x}, \mathbf{y}, \alpha} \; & \inf_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}\left( \min\left\{ 1, \alpha B - \alpha \sum_{n=1}^N c_n(y_n(\tilde{\mathbf{z}})) - \alpha p(x_M(\tilde{\mathbf{z}}) - \tau) \right\} \right) \\
\text{s.t.} \; & x_{j_n}(\tilde{\mathbf{z}}) - x_{i_n}(\tilde{\mathbf{z}}) \geq (\tilde{z}_n - y_n(\tilde{\mathbf{z}}))^+, \quad \forall\, n \in [N], \\
& \mathbf{0} \leq \mathbf{y}(\tilde{\mathbf{z}}) \leq \mathbf{u}, \\
& \mathbf{x}(\tilde{\mathbf{z}}) \geq \mathbf{0}, \\
& x_m \in \mathscr{Y}(1, N, I_m^x), \quad \forall\, m \in [M], \\
& y_n \in \mathscr{Y}(1, N, I_n^y), \quad \forall\, n \in [N], \\
& \alpha \geq 0,
\end{aligned} \tag{6}$$

where $\tilde{z}_n$ denotes the uncertain time for activity $n$. The solution $(\mathbf{x}(\cdot), \mathbf{y}(\cdot))$ to problem (6) is a set of decision rules that prescribes the start time of each node and the amount of crashing for each activity.

## 5. Approximate Solution

Because of the intractability of the optimal crashing model (6), we construct tractable approximations for (6) that yield feasible decisions. It is useful to express (6) less concisely. First, we convert the maximization problem into a minimization problem, as is conventional in convex programming. Second, we substitute the piecewise-linear crashing and penalty cost functions above into their respective terms in (6). Third, we linearize the objective by introducing auxilliary DRs $r, \mathbf{s}$, and $t$, which are epigraph DRs on the objective, the crashing cost, and the penalty cost, respectively, and move the terms from the objective into the constraints. This yields the equivalent optimization model

$$
\begin{aligned}
Z_0^* = 1 - \min_{\mathbf{x},\mathbf{y},\mathbf{s},r,t,\alpha} \; & \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(r(\tilde{\mathbf{z}}))\\
\text{s.t.}\;\; & 1-\alpha B+\sum_{n=1}^{N} s_n(\tilde{\mathbf{z}})+t(\tilde{\mathbf{z}})\le r(\tilde{\mathbf{z}}),\\
& \alpha c_{n\ell} y_n(\tilde{\mathbf{z}})+\alpha d_{n\ell}\le s_n(\tilde{\mathbf{z}}), && \forall \ell\in[L], \forall n\in[N],\\
& \alpha p_k x_M(\tilde{\mathbf{z}})+\alpha q_k-\alpha p_k\tau\le t(\tilde{\mathbf{z}}), && \forall k\in[K],\\
& x_{j_n}(\tilde{\mathbf{z}})-x_{i_n}(\tilde{\mathbf{z}})\ge \tilde{z}_n-y_n(\tilde{\mathbf{z}}), && \forall n\in[N],\\
& x_{j_n}(\tilde{\mathbf{z}})-x_{i_n}(\tilde{\mathbf{z}})\ge 0, && \forall n\in[N],\\
& \mathbf{0}\le\mathbf{y}(\tilde{\mathbf{z}})\le\mathbf{u},\quad \mathbf{x}(\tilde{\mathbf{z}})\ge\mathbf{0},\quad r(\tilde{\mathbf{z}})\ge 0,\quad \alpha\ge 0,\\
& x_m\in\mathscr{Y}(1,N,I_m^x), && \forall m\in[M],\\
& y_n\in\mathscr{Y}(1,N,I_n^y), && \forall n\in[N],\\
& \mathbf{s}\in\mathscr{Y}(N,N,[N]),\\
& r,t\in\mathscr{Y}(1,N,[N]).
\end{aligned}
\tag{7}
$$

### 5.1. Linear Decision Rule Approximations

We begin with a linear approximation of problem (6), where we restrict all decision rules to be LDRs. Notationally, this involves approximating the larger spaces of generic DRs in (7), denoted by $\mathscr{Y}$, with the spaces of affine functions $\mathscr{L}$ as defined in (5), and using the nonanticipativity constraints to reflect the precedence structure of the project network. Then, we formulate problem (7) as a robust optimization problem, using two transformations. First, consider a scalar-valued LDR $x$ with a generic information set $I\subseteq[N]$. As discussed in §4.1, $x$ can be represented by decision variables $x^0$ and $\mathbf{x}$, as $x(\tilde{\mathbf{z}})=x^0+\mathbf{x}'\tilde{\mathbf{z}}$, with $x^i=0$, for $i\notin I$. Now, the constraint $x(\tilde{\mathbf{z}})\ge 0$ can be written as $x^0+\mathbf{x}'\tilde{\mathbf{z}}\ge 0$, which is equivalent to $x^0+\mathbf{x}'\mathbf{z}\ge 0$, for all $\mathbf{z}\in\mathscr{W}$. Second, because $x_M(\tilde{\mathbf{z}})$ has representation $x_M(\tilde{\mathbf{z}})=x_M^0+\mathbf{x}_M'\tilde{\mathbf{z}}$, we have a nonlinear term $\alpha x_M^0$. To linearize this term, we define the product $\alpha x_M^0$ as a new single decision variable.

Applying these transformations iteratively to the LDRs, the robust optimization problem becomes

$$
\begin{aligned}
Z_{\mathrm{LDR}} := 1-\min \; & \sup_{\tilde{\mathbf{z}}\in\hat{\mathscr{W}}}\{r^0+\mathbf{r}'\tilde{\mathbf{z}}\}\\
\text{s.t.}\;\; & 1-\alpha B+t^0-r^0+(\mathbf{t}-\mathbf{r})'\mathbf{z}\\
&\quad +\sum_{n=1}^{N} s_n^0+\mathbf{s}_n'\mathbf{z}\le 0, && \forall \mathbf{z}\in\mathscr{W},\\
& c_{n\ell}y_n^0+\alpha d_{n\ell}-s_n^0+(c_{n\ell}\mathbf{y}_n-\mathbf{s}_n)'\mathbf{z}\le 0,\\
& && \forall \mathbf{z}\in\mathscr{W}, \forall \ell\in[L], \forall n\in[N],\\
& p_k x_M^0+\alpha q_k-\alpha p_k\tau-t^0+(p_k\mathbf{x}_M-\mathbf{t})'\mathbf{z}\le 0,\\
& && \forall \mathbf{z}\in\mathscr{W}, \forall k\in[K],\\
& x_{j_n}^0-x_{i_n}^0+\mathbf{x}_{j_n}'\mathbf{z}-\mathbf{x}_{i_n}'\mathbf{z}\ge\alpha z_n-y_n^0-\mathbf{y}_n'\mathbf{z},\\
& && \forall \mathbf{z}\in\mathscr{W}, \forall n\in[N],\\
& x_{j_n}^0-x_{i_n}^0+\mathbf{x}_{j_n}'\mathbf{z}-\mathbf{x}_{i_n}'\mathbf{z}\ge 0, && \forall \mathbf{z}\in\mathscr{W}, \forall n\in[N],\\
& 0\le y_n^0+\mathbf{y}_n'\mathbf{z}\le\alpha u_n, && \forall \mathbf{z}\in\mathscr{W}, \forall n\in[N],\\
& x^0+\mathbf{x}_m'\mathbf{z}\ge 0, && \forall \mathbf{z}\in\mathscr{W}, \forall m\in[M],\\
& r^0+\mathbf{r}'\mathbf{z}\ge 0, && \forall \mathbf{z}\in\mathscr{W},\\
& \alpha\ge 0,\\
& x_{mj}=0, && \forall m\in[M], \forall j\notin I_m^x,\\
& y_{nj}=0, && \forall n\in[N], \forall j\notin I_n^y.
\end{aligned}
\tag{8}
$$

where the minimization is taken with respect to $\alpha, t^0, \mathbf{t}, r^0, \mathbf{r}, \{x_i^0, \mathbf{x}_i\}_{i=1}^M, \{y_n^0, \mathbf{y}_n, s_n^0, \mathbf{s}_n\}_{i=1}^N$.

Problem (8) can be solved using robust counterpart techniques from the literature of robust optimization (Ben-Tal and Nemirovski 1998, Bertsimas and Sim 2004). If $\mathscr{W}$ and $\hat{\mathscr{W}}$ are polytopes, then model (8) reduces to a simple linear program. The next two results summarize the relationship between this linear approximation and the original problem.

Theorem 1. *If model* (8) *has a feasible solution where* $\alpha>0$, *then*

$$
\begin{aligned}
x_i^*(\tilde{\mathbf{z}}) &= \frac{1}{\alpha}(x_i^0+\mathbf{x}_i'\tilde{\mathbf{z}}), && \forall i\in[M],\\
y_n^*(\tilde{\mathbf{z}}) &= \frac{1}{\alpha}(y_n^0+\mathbf{y}_n'\tilde{\mathbf{z}}), && \forall n\in[N],\\
s_n^*(\tilde{\mathbf{z}}) &= s_n^0+\mathbf{s}_n'\tilde{\mathbf{z}}, && \forall n\in[N],\\
r^*(\tilde{\mathbf{z}}) &= r^0+\mathbf{r}'\tilde{\mathbf{z}},\\
t^*(\tilde{\mathbf{z}}) &= t^0+\mathbf{t}'\tilde{\mathbf{z}},\\
\alpha^* &= \alpha
\end{aligned}
$$

*is a feasible solution to* (7).

Proof. By assumption, $\mathbf{x}^*(\tilde{\mathbf{z}})$ and $\mathbf{y}^*(\tilde{\mathbf{z}})$ are affine functions of $\tilde{\mathbf{z}}$. From the last two constraints of model (8), and recalling the definition (5) of $\mathcal{L}(m,N,I)$, we have

$$x_m^* \in \mathcal{L}(1,N,I_m^x) \subset \mathcal{Y}(1,N,I_m^x), \quad \forall\, m \in [M],$$
$$y_n^* \in \mathcal{L}(1,N,I_n^y) \subset \mathcal{Y}(1,N,I_n^y), \quad \forall\, n \in [N].$$

The linear inequality constraints in model (7) are directly satisfied by the corresponding linear inequality constraints in model (8), by substitution. □

It follows from Theorem 1 that the robust optimization problem (8) provides a feasible, but generally suboptimal, approximation to the intractable problem (6).

Corollary 1. *The following inequality holds:* $Z_0^* \geq Z_{\mathrm{LDR}}$.

Thus, by solving a tractable convex program, we use Theorem 1 to construct an approximate feasible solution to the original intractable problem (6). We refer to the decision rules constructed by this method as the *LDR approximation* to the project crashing problem. We next develop a closer approximation.

### 5.2. Piecewise-Linear Decision Rule Approximation

We now construct DRs for the robust crashing problem that are more flexible than LDRs. We start from an LDR approximation of the problem and replace the explicit constraints by weighted penalty terms in the objective (Chen et al. 2008, Goh and Sim 2010). The solution to the resulting problem provides piecewise-linear decision rules by adding nonlinear terms to the LDR. Our approach exploits the special structure of the project network constraints. This approach defines the following problem:

$$
\begin{aligned}
&Z^*_{\mathrm{PWLDR}}\\
&:= 1 - \min_{\mathbf{x},\mathbf{y},\mathbf{s},r,t,\alpha} \Bigg\{ \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((r(\tilde{\mathbf{z}}))^+)\\
&\quad + \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((1-\alpha B + \mathbf{e}'\mathbf{s}(\tilde{\mathbf{z}}) + t(\tilde{\mathbf{z}}) - r(\tilde{\mathbf{z}}))^+)\\
&\quad + \sum_{n=1}^{N}\sum_{\ell=1}^{L} \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((c_{n\ell}y_n(\tilde{\mathbf{z}}) + \alpha d_{n\ell} - s_n(\tilde{\mathbf{z}}))^+)\\
&\quad + \sum_{k=1}^{K} \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((p_k x_M(\tilde{\mathbf{z}}) + \alpha q_k - \alpha p_k \tau - t(\tilde{\mathbf{z}}))^+)\\
&\quad + p_K \sum_{n=1}^{N} \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((\alpha\tilde{z}_n - x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}) - y_n(\tilde{\mathbf{z}}))^+)\\
&\quad + p_K \sum_{n=1}^{N} \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((-x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}))^+)\\
&\quad + \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(\mathbf{c}_L'(\mathbf{y}(\tilde{\mathbf{z}}))^- + (p_K - \mathbf{c}_1')^+(\mathbf{y}(\tilde{\mathbf{z}}) - \alpha\mathbf{u})^+)\\
&\quad + p_K \mathbf{e}' \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((\mathbf{x}(\tilde{\mathbf{z}}))^-) \Bigg\}\\
&\text{s.t. } x_m \in \mathcal{L}(1,N,I_m^x), \quad \forall\, m\in[M],\\
&\qquad y_n \in \mathcal{L}(1,N,I_n^y), \quad \forall\, n\in[N],\\
&\qquad \mathbf{s} \in \mathcal{L}(N,N,[N]),\\
&\qquad r,t \in \mathcal{L}(1,N,[N]),
\end{aligned} \tag{9}
$$

where we define $\mathbf{e} := (1,\ldots,1)$, $\mathbf{c}_1 := (c_{11},\ldots,c_{N1})$, and $\mathbf{c}_L := (c_{1L},\ldots,c_{NL})$.

We show that if a solution exists for (9), then we can use the solution to construct piecewise-linear DRs that are better than the LDR approximation. Before constructing these decision rules, we first define the following sets. For any $j\in[M]$, let $\mathcal{P}(j) := \{i\in[M]\colon (i_n,j_n) = (i,j) \text{ for some } n\in[N]\}$, and

$$\mathcal{N}(j) := \{j\} \cup \bigcup_{i\in\mathcal{P}(j)} \mathcal{N}(i), \quad \text{and}$$
$$\mathcal{A}(j) := \{n\in[N]\colon j_n \in \mathcal{N}(j)\}.$$

Here, $\mathcal{P}(j)$ represents the set of nodes that are immediate predecessors of $j$, whereas the recursively defined $\mathcal{N}(j)$ represents the set of predecessor nodes of $j$. Similarly, $\mathcal{A}(j)$ represents the set of arcs that enter into all the predecessor nodes of $j$. We now relate problem (9) to the original problem (6).

Theorem 2. *If model* (9) *has a feasible solution* $(\alpha,\mathbf{x}(\tilde{\mathbf{z}}),\mathbf{y}(\tilde{\mathbf{z}}),\mathbf{s}(\tilde{\mathbf{z}}),r(\tilde{\mathbf{z}}),t(\tilde{\mathbf{z}}))$ *where* $\alpha>0$, *then for each* $m\in[M]$,

$$
\begin{aligned}
x_m^*(\tilde{\mathbf{z}}) &= \frac{1}{\alpha}\Bigg(x_m(\tilde{\mathbf{z}}) + \sum_{i\in\mathcal{N}(m)} (x_i(\tilde{\mathbf{z}}))^- + \sum_{n\in\mathcal{A}(m)} (y_n(\tilde{\mathbf{z}}) - \alpha u_n)^+\Bigg)\\
&\quad + \frac{1}{\alpha}\Bigg(\sum_{n\in\mathcal{A}(m)} (\alpha\tilde{z}_n - y_n(\tilde{\mathbf{z}}) - x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}))^+\\
&\qquad + (-x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}))^+\Bigg),
\end{aligned}
$$

*and*

$$\mathbf{y}^*(\tilde{\mathbf{z}}) = \min\left\{\max\left\{\frac{1}{\alpha}\mathbf{y}(\tilde{\mathbf{z}}), \mathbf{0}\right\}, \mathbf{u}\right\}$$

*is a feasible solution to* (6).

Proof. See the appendix.

Theorem 2 shows that the PWLDRs are feasible decision rules. The next theorem compares the strength of the LDR and PWLDR bounds.

Theorem 3. *The following inequality holds:* $Z_0^* \geq Z^*_{\mathrm{PWLDR}} \geq Z_{\mathrm{LDR}}$.

Proof. See the appendix.

Theorem 3 shows that piecewise-linear decision rules provide a tighter approximation to the optimal objective $Z_0^*$ than LDRs. Solving problem (9) is more difficult than solving (8), because of the piecewise-linear terms within the expectations, $\sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((\cdot)^+)$. Furthermore, evaluating this worst-case expectation is typically an intractable problem (Bertsimas and Popescu 2005); hence, we use an upper bound.

Proposition 2 (Goh and Sim 2010). *Let $\mathbb{F}$ be defined as in* (2). *Additionally defining*

$$\pi(y^0,\mathbf{y}) := \min_{x^0,\mathbf{x}} \pi^1(y^0 - x^0, \mathbf{y}-\mathbf{x}) + \pi^2(x^0,\mathbf{x}),$$

*where*

$$\pi^1(y^0,\mathbf{y}) := \inf_{\mathbf{s}\in\mathbb{R}^N}\left(\sup_{\hat{\mathbf{z}}\in\hat{\mathscr{W}}}\{\mathbf{s}'\hat{\mathbf{z}}\} + \sup_{\mathbf{z}\in\mathscr{W}}(\max\{y^0+\mathbf{y}'\mathbf{z}-\mathbf{s}'\mathbf{z}, -\mathbf{s}'\mathbf{z}\})\right),$$

*and*

$$\pi^2(y^0,\mathbf{y}) := \begin{cases} \sup_{\hat{\mathbf{z}}\in\hat{\mathscr{W}}}\left\{\frac{1}{2}(y^0+\mathbf{y}'\hat{\mathbf{z}}) + \frac{1}{2}\sqrt{(y^0+\mathbf{y}'\hat{\mathbf{z}})^2 + \mathbf{y}'\boldsymbol{\Sigma}\mathbf{y}}\right\} & \\ \qquad \text{if } \boldsymbol{\Sigma} \text{ is known}, & \\ 0 \qquad \text{if } \boldsymbol{\Sigma} \text{ is unknown and } (y^0,\mathbf{y}) = (0,\mathbf{0}), & \\ +\infty \qquad \text{if } \boldsymbol{\Sigma} \text{ is unknown and } (y^0,\mathbf{y}) \neq (0,\mathbf{0}), & \end{cases}$$

*we have*

$$\sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((y^0+\mathbf{y}\tilde{\mathbf{z}})^+) \le \pi(y^0,\mathbf{y}).$$

Applying Proposition 2, we use $\pi(\cdot,\cdot)$ functions to bound each of the $\sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((\cdot)^+)$ terms in the objective of (9) separately and obtain the following approximation:

$$\begin{aligned} &Z_{\text{PWLDR}} \\ &:= 1 - \min\Bigg\{\pi(r^0,\mathbf{r}) \\ &\quad + \pi\left(1-\alpha B + \sum_{n=1}^N s_n^0 + t^0 - r^0, \sum_{n=1}^N \mathbf{s}_n + \mathbf{t} - \mathbf{r}\right) \\ &\quad + \sum_{n=1}^N\sum_{\ell=1}^L \pi(c_{n\ell}y_n^0 + \alpha d_{n\ell} - s_n^0, c_{n\ell}\mathbf{y}_n - \mathbf{s}_n) \\ &\quad + \sum_{k=1}^K \pi(p_k x_M^0 + \alpha q_k - \alpha p_k\tau - t^0, p_k\mathbf{x}_M - \mathbf{t}) \\ &\quad + p_K\sum_{n=1}^N \pi(-x_{j_n}^0 + x_{i_n}^0 - y_n^0, \alpha\mathbf{e}^n - \mathbf{x}_{j_n} + \mathbf{x}_{i_n} - \mathbf{y}_n) \\ &\quad + p_K\sum_{n=1}^N \pi(-x_{j_n}^0 + x_{i_n}^0, -\mathbf{x}_{j_n} + \mathbf{x}_{i_n}) \\ &\quad + p_K\sum_{m=1}^M \pi(-x_m^0, -\mathbf{x}_m) + \sum_{n=1}^N c_{nL}\pi(-y_n^0, -\mathbf{y}_n) \\ &\quad + (p_K - c_{n1})^+\pi(y_n^0 - \alpha u_n, \mathbf{y}_n)\Bigg\}, \\ &\text{s.t.}\quad x_{mj} = 0, \quad \forall\, m\in[M],\ \forall\, j\notin I_m^x, \\ &\qquad\ \ y_{nj} = 0, \quad \forall\, n\in[N],\ \forall\, j\notin I_n^y, \end{aligned} \tag{10}$$

where the minimization is taken with respect to $\alpha$, $t^0$, $\mathbf{t}$, $r^0$, $\mathbf{r}$, $\{x_m^0, \mathbf{x}_m\}_{m=1}^M$, $\{y_n^0, \mathbf{y}_n, s_n^0, \mathbf{s}_n\}_{n=1}^N$.

The function values $\pi(\cdot,\cdot)$ are determined by the model parameters, i.e., the sets $\mathscr{W}$, $\hat{\mathscr{W}}$ and, if available, the covariance matrix $\boldsymbol{\Sigma}$. We now show that problem (10) can be represented as a tractable second-order conic program (Nesterov and Nemirovski 1994). We introduce auxilliary decision variables to linearize the objective by bounding the individual nonlinear $\pi(\cdot,\cdot)$ terms. The transformed problem (10) has a linear objective, with nonlinear constraints. For example, if the auxiliary variable $w$ is introduced to bound $\pi(r^0,\mathbf{r})$ in the objective of (10), the associated nonlinear constraint becomes $\pi(r^0,\mathbf{r}) \le w$. After substituting the definition of $\pi(\cdot,\cdot)$, the constraint can be reformulated as the set of robust constraints:

$$\begin{aligned} \mathbf{s}'\hat{\mathbf{z}} + (r^0 - x^0) + (\mathbf{r}-\mathbf{x})'\mathbf{z} - \mathbf{s}'\mathbf{z} \le w, &\quad \forall\,\mathbf{z}\in\mathscr{W},\ \hat{\mathbf{z}}\in\hat{\mathscr{W}}, \\ \mathbf{s}'\hat{\mathbf{z}} - \mathbf{s}'\mathbf{z} \le w, &\quad \forall\,\mathbf{z}\in\mathscr{W},\ \hat{\mathbf{z}}\in\hat{\mathscr{W}}, \\ x^0 + \mathbf{x}'\hat{\mathbf{z}} \le v, &\quad \forall\,\hat{\mathbf{z}}\in\hat{\mathscr{W}}, \\ \tfrac{1}{2}v + \tfrac{1}{2}\sqrt{v^2 + \mathbf{x}'\boldsymbol{\Sigma}\mathbf{x}} \le w, & \end{aligned}$$

with auxilliary variables $(\mathbf{s}, \mathbf{x}, x^0, v)$. These constraints are second-order conic representable (Ben-Tal and Nemirovski 1998, Bertsimas and Sim 2004).

Thus, from any solution of (10), we can construct LDRs that are feasible in (9) and then use the construction in Theorem 2 to obtain a piecewise-linear DR that is feasible in (4). This approximation is referred to as the *PWLDR approximation*. The next result compares the LDR and PWLDR bounds.

Theorem 4. *The following inequality holds*: $Z_0^* \ge Z_{\text{PWLDR}}^* \ge Z_{\text{PWLDR}} \ge Z_{\text{LDR}}$.

Proof. See the appendix.

Theorem 4 implies that we can obtain decision rules with stronger bounds than LDRs. The bound improvement arises from two sources. First, the piecewise-linear rules explore a larger space of possible decisions. Second, by using the piecewise-linear rules and related bounds, we incorporate correlation information between different activities, through the covariance matrix $\boldsymbol{\Sigma}$ in $\pi^2(\cdot)$.

## 6. Rolling Horizon Crashing Model

To make the most effective use of periodic information updates as well as activity completion time information, we develop a rolling horizon crashing model. As described in §2.5, an updated project report is received in each period. Then, the project manager updates the model parameters based on the realized uncertainties and solves problem (6).

The solution follows an iterative procedure. The iteration number is denoted by $t \ge 0$. The vector of unfinished fraction of each activity in the $t$th period is $\mathbf{f}^{(t)} \in [0,1]^N$. Also, we denote by $\mathbf{g}^{(t)} \in [0,1]^N$ the vector of fractional work completed for each activity in period $t$. Accordingly, we have

$$\mathbf{g}^{(t)} = \mathbf{f}^{(t-1)} - \mathbf{f}^{(t)}. \tag{11}$$

Furthermore, let $\mathbf{D}_\mathbf{f}^{(t)}$ denote the $N \times N$ diagonal matrix with $\mathbf{f}^{(t)}$ on the main diagonal, and let $\mathbf{D}_\mathbf{g}^{(t)}$ denote the $N \times N$ diagonal matrix with $\mathbf{g}^{(t)}$ on the main diagonal. The solution procedure uses the algorithm CrashPlan, described below. For simplicity, the algorithm is described for the case that the mean of uncertainties is a singleton, i.e., $\hat{\mathcal{W}} = \{\boldsymbol{\mu}\}$ for some known $\boldsymbol{\mu} \in \mathcal{W}$. The extension to the general case is straightforward.

**Algorithm** (CrashPlan)

*Step* 1. Initialize $t = 0$, $\mathbb{F}^{(0)} = \mathbb{F}$, $\mathbf{f}^{(0)} = \mathbf{e}$, $\mathbf{u}^{(0)} = \mathbf{u}$, $\tau^{(0)} = \tau$, $B^{(0)} = B$.

*Step* 2. Solve the following problem:

$$\begin{aligned} &Z_0(t)^* \\ &\quad = \max_{\mathbf{x}^{(t)},\,\mathbf{y}^{(t)},\,\alpha} \left\{ \inf_{\mathbb{P}\in\mathbb{F}^{(t)}} \mathrm{E}_\mathbb{P}\left( \min\left\{ 1, \alpha B^{(t)} - \alpha \sum_{n=1}^{N} c_n(y_n^{(t)}(\tilde{\mathbf{z}})) \right.\right.\right. \\ &\qquad\qquad \left.\left.\left. - \alpha p(x_M^{(t)}(\tilde{\mathbf{z}}) - \tau^{(t)}) \right\} \right) \right\} \end{aligned}$$

$$\begin{aligned} \text{s.t.}\quad & x_{j_n}^{(t)}(\tilde{\mathbf{z}}) - x_{i_n}^{(t)}(\tilde{\mathbf{z}}) \ge (\tilde{z}_n - y_n^{(t)}(\tilde{\mathbf{z}}))^+, \quad \forall\, n \in [N], \\ & \mathbf{0} \le \mathbf{y}^{(t)}(\tilde{\mathbf{z}}) \le \mathbf{u}^{(t)}, \\ & \mathbf{x}^{(t)}(\tilde{\mathbf{z}}) \ge \mathbf{0}, \\ & x_m^{(t)} \in \mathcal{Y}(1, N, I_m^x), \quad \forall\, m \in [M], \\ & y_n^{(t)} \in \mathcal{Y}(1, N, I_n^y), \quad \forall\, n \in [N],\ \alpha \ge 0 \end{aligned} \tag{12}$$

to obtain new decision rules $(\mathbf{x}^{(t)}(\cdot), \mathbf{y}^{(t)}(\cdot))$ for start times and crashing amounts, which are implemented in period $t+1$.

*Step* 3. Wait to receive the $t$th periodic project report.

*Step* 4. Obtain $\mathbf{f}^{(t)}$ from the report. Compute $\mathbf{g}^{(t)}$ from (11). Also, compute the matrices $\mathbf{D}_\mathbf{f}^{(t)}$ and $\mathbf{D}_\mathbf{g}^{(t)}$. If $\mathbf{f}^{(t)} = \mathbf{0}$, then stop. Otherwise, continue with Step 5.

*Step* 5. Update the family of uncertainty distributions using

$$\begin{aligned} \mathbb{F}^{(t+1)} = \big\{ \mathbb{P} : \mathrm{E}_\mathbb{P}(\tilde{\mathbf{z}}) = \mathbf{D}_\mathbf{f}^{(t)}\boldsymbol{\mu},\ \mathrm{E}_\mathbb{P}(\tilde{\mathbf{z}}\tilde{\mathbf{z}}') = \mathbf{D}_\mathbf{f}^{(t)}(\boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}')\mathbf{D}_\mathbf{f}^{(t)}, \\ \mathbb{P}(\tilde{\mathbf{z}} \in \mathcal{W}^{(t+1)}) = 1 \big\}, \end{aligned}$$

where

$$\mathcal{W}^{(t+1)} = \left\{ \boldsymbol{\zeta} \in \mathbb{R}^N \colon \zeta_n = \frac{z_n}{f_n} \mathbb{1}_{\{f_n > 0\}},\ \forall\, n \in [N],\ \mathbf{z} \in \mathcal{W} \right\}. \tag{13}$$

*Step* 6. Update the vector of remaining crash limits using

$$\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} - \mathbf{D}_\mathbf{g}^{(t)}\mathbf{y}^{(t)}(\tilde{\mathbf{z}}). \tag{14}$$

*Step* 7. Update the remaining time and budget, respectively, using

$$\begin{aligned} &\tau^{(t+1)} = \tau - tT \quad \text{and} \\ &B^{(t+1)} = B - \sum_{n=1}^{N} c_n(y_n^{(t)}(\tilde{\mathbf{z}}) \cdot (1 - f_n^{(t)})). \end{aligned} \tag{15}$$

*Step* 8. If $\mathbf{f}^{(t)} = \mathbf{0}$, then stop. Otherwise, set $t = t+1$, and go to Step 2.

In Step 5, the family of uncertainty distributions $\mathbb{F}^{(t+1)}$ is updated using a heuristic that proportionately scales the distributional properties. In particular, when $\mathcal{W} = \{\mathbf{z} \in \mathbb{R}^N \colon \mathbf{z}_l \le \mathbf{z} \le \mathbf{z}_u\}$ for vectors of lower and upper limits $\mathbf{z}_l$ and $\mathbf{z}_u$, the support update simplifies to

$$\mathcal{W}^{(t+1)} = \{\boldsymbol{\zeta} \in \mathbb{R}^N \colon {\mathbf{D}_\mathbf{f}}^{(t)}\mathbf{z}_l \le \boldsymbol{\zeta} \le {\mathbf{D}_\mathbf{f}}^{(t)}\mathbf{z}_u\}.$$

Regarding consumption of the budget by crashing cost, we assume that crashing activity and the accrual of crashing cost occur at a constant rate within each period. This assumption is consistent with crashing by assigning additional resources to an activity, where resources have constant usage and cost rates throughout the period, as in earned value analysis (Klastorin 2004). The amount of crashing used in the $t$th period can be written as $\mathbf{D}_\mathbf{g}^{(t)}\mathbf{y}^{(t)}(\tilde{\mathbf{z}})$, which explains the updating of $\mathbf{u}^{(t)}$ in Step 6.

We note that $\{I_n^y\}_{n=1}^N$ is constructed such that at any update period $t$, if $g_n^{(t)} > 0$ for some $n \in [N]$, the crashing decision $y_n^{(t)}(\tilde{\mathbf{z}})$ is known, based on the activity times that are revealed by period $t$. Consequently, in (14) and (15), the crash limits and budget are only updated with known quantities.

Because the rolling horizon approach uses more information than model (6), we expect it to perform better. In §7, we examine the extent of this improvement.

# 7. Computational Study

## 7.1. Description

In this section, we summarize the results of our computational study on the crashing strategies described above. Following the guidelines of Hall and Posner (2001), (a) we generate a wide range of parameter specifications, (b) the data generated are representative of real-world scenarios, and (c) the experimental design varies only the parameters that may affect the analysis. Also, we use the procedure described by Hall and Posner (2001), within each project network, to generate precedence constraints that have an expected density, i.e., the ratio of the number of immediate precedence relationships in the graph to the maximum possible, that meets a prespecified target and is uniform across the network.

We use the random network generator *RanGen*1 of Demeulemeester et al. (2003) to generate networks with a wide range of network topologies. The 500 project instances generated contain 10 activities each, with a density of 0.5. For each network, we simulate 100 sets of activity times for each data set, varying the simulation parameters across data sets, as discussed below. This gives us a total of 50,000 activity time vectors in each data set. We further study 20-activity networks with 200 project instances in each data set, again with 100 sets of activity times per network.

We then apply a series of transformations to each network to fit it into our modeling framework. First, we convert the generated networks from the AON convention to the AOA convention, using the technique of Cohen and Sadeh (2007). Second, to introduce stochasticity into the project networks, we use the activity times generated by *RanGen*1 as mean activity times. We assume that each activity time has support on the interval $[0.2t, 1.8t]$, where $t$ represents the mean activity time of a given activity. For simplicity, we consider linear crashing costs, penalty costs that are 0 for on-time completion, and linear penalties per unit of schedule overrun. For each project network, the project completion time $\tau$ and project budget $B$ are chosen so that if the PERT assumptions hold, then the project is delayed with probability 0.90 and overruns its budget with probability 0.70 if no crashing is used. In rolling horizon mode, we use an update period of $T = 0.7\tau$ for our rolling horizon LDR (RHLDR) and rolling horizon PWLDR (RHPWLDR) crashing policies, such that each network receives approximately two updates.

For each crashing policy, we consider two performance metrics, (a) the probability of project success and (b) the expected loss conditional on project failure. Formally, if we let $\tilde{V}$ represent the total project expenditure in any given run, the project success probability can be computed as $p_S = \mathrm{E}(\mathbb{1}_{\{\tilde{V} \le B\}})$, whereas the conditional expected loss can be computed as $L = \mathrm{E}(\tilde{V} - B \mid \tilde{V} > B)$. In all studies, we report the conditional expected loss as a percentage of the budget $B$ for the project.

### 7.2. Benchmarks

We compare our LDR and PWLDR crashing policies with existing heuristic crashing policies, namely, a crashing policy based on PERT assumptions (PERT), and crashing policies based on MCSIM. As discussed in the introduction, these are the two main approaches used by project managers for project scheduling under uncertainty.

PERT bases crashing decisions on expected activity times $\bar{z}$. Thus, our PERT heuristic finds event times $\mathbf{x}$ and crash amounts $\mathbf{y}$ to minimize the total cost of the expected problem, i.e.,

$$\begin{aligned} Z_{\text{PERT}} := \min_{\mathbf{x},\,\mathbf{y}} \; & \left\{ \sum_{n=1}^{N} c_n y_n + p(x_M - \tau) \right\} \\ \text{s.t.} \;\; & x_{j_n} - x_{i_n} \ge (\bar{z}_n - y_n)^+, \quad \forall\, n \in [N], \\ & \mathbf{0} \le \mathbf{y} \le \mathbf{u}, \\ & \mathbf{x} \ge \mathbf{0}. \end{aligned} \tag{16}$$

In the MCSIM heuristic, we simulate $R = 100$ sets of activity times from an assumed probability distribution. In §7.3, we study the robustness of MCSIM to this distributional assumption, and in §7.4, we optimistically assume that the manager knows the true distribution. In either case, crashing decisions $\mathbf{y}$ are made based on the simulated activity times. The matrix $\mathbf{X} \in \mathbb{R}^{M \times R}$ is a matrix of decisions, with columns that represent the end times of the activities in each run.

In the standard MCSIM case, we consider a total expected cost objective, namely,

$$\begin{aligned} Z_{\text{MCSIM}} := \min_{\mathbf{X},\,\mathbf{y}} & \left\{ \sum_{n=1}^{N} c_n y_n + \frac{1}{R} \sum_{r=1}^{R} p(X_{Mr} - \tau) \right\} \\ \text{s.t.} \;\; & X_{j_n r} - X_{i_n r} \ge (Z_{nr} - y_n)^+, \quad \forall\, n \in [N],\; r \in [R], \\ & \mathbf{0} \le \mathbf{y} \le \mathbf{u}, \\ & \mathbf{X} \ge \mathbf{0}. \end{aligned} \tag{17}$$

To provide a more thorough comparison, we also present results from two alternative MCSIM heuristics, MCSIM-EO and MCSIM-CV. In these heuristics, we solve (17) using different objective functions. In MCSIM-EO, we use the expected budget overage as the objective, i.e., $\mathrm{E}((\tilde{V} - B)^+)$, where $\tilde{V}$ represents the total project expenditure; in MCSIM-CV, we use the CVaR satisficing measure (3) as the objective. For the rolling horizon computations, to provide a fair comparison, we use the same update period, $T = 0.7\tau$, as in our crashing policies.

Finally, we report three clairvoyant benchmarks, CLVT, CLVT-EO, and CLVT-CV. In these benchmarks, we use the same samples of uncertain activity times as MCSIM, but here we assume that all realized activity times are known *before* crashing decisions are made. Consequently, the crashing problems are deterministic optimization problems. CLVT, CLVT-EO, and CLVT-CV, respectively, use total cost, total overage, and the CVaR satisficing measure as objectives for their optimizations, and are upper bounds on the performance of each network.

**Table 1 Robustness Study: Static Mode**

| Success percentage | | | | | | | |
|---|---|---|---|---|---|---|---|
| Crash method | Assumed distribution | R0 $\beta(4,4)$ | R1 $\beta(2,2)$ | R2 $\beta(1,1)$ | R3 $\beta(0.5,0.5)$ | R4 Bernoulli(0.5) | Min |
| MCSIM | $\beta(4,4)$ | 96.77 (0.79) | 86.56 (1.53) | 71.39 (2.02) | 55.38 (2.22) | 29.70 (2.04) | 29.70 (2.04) |
| | Uniform | 97.91 (0.64) | 97.12 (0.75) | 91.86 (1.22) | 79.93 (1.79) | 42.93 (2.21) | 42.93 (2.21) |
| | Bernoulli(0.5) | 83.80 (1.65) | 84.20 (1.63) | 85.80 (1.56) | 86.96 (1.51) | 85.29 (1.58) | 83.80 (1.65) |
| MCSIM-EO | $\beta(4,4)$ | 90.27 (1.33) | 85.76 (1.56) | 81.60 (1.73) | 75.85 (1.91) | 48.72 (2.24) | 48.72 (2.24) |
| | Uniform | 87.59 (1.47) | 85.98 (1.55) | 85.16 (1.59) | 78.02 (1.85) | 54.12 (2.23) | 54.12 (2.23) |
| | Bernoulli(0.5) | 71.20 (2.03) | 70.60 (2.04) | 73.20 (1.98) | 77.60 (1.86) | 67.42 (2.10) | 67.42 (2.10) |
| MCSIM-CV | $\beta(4,4)$ | 99.85 (0.17) | 98.55 (0.53) | 93.94 (1.07) | 85.87 (1.56) | 60.30 (2.19) | 60.30 (2.19) |
| | Uniform | 97.97 (0.63) | 98.23 (0.59) | 96.76 (0.79) | 91.57 (1.24) | 60.29 (2.19) | 60.29 (2.19) |
| | Bernoulli(0.5) | 83.80 (1.65) | 84.20 (1.63) | 85.60 (1.57) | 86.80 (1.51) | 85.60 (1.57) | 83.80 (1.65) |
| LDR | — | 90.35 (1.32) | 89.17 (1.39) | 91.57 (1.24) | 91.39 (1.25) | 91.01 (1.28) | 89.17 (1.39) |

| Conditional expected loss | | | | | | | |
|---|---|---|---|---|---|---|---|
| Crash method | Assumed distribution | R0 $\beta(4,4)$ | R1 $\beta(2,2)$ | R2 $\beta(1,1)$ | R3 $\beta(0.5,0.5)$ | R4 Bernoulli(0.5) | Max |
| MCSIM | $\beta(4,4)$ | 30.91 (0.70) | 48.02 (0.50) | 68.70 (0.45) | 94.41 (0.45) | 170.34 (0.51) | 170.34 (0.51) |
| | Uniform | 11.14 (0.28) | 20.69 (0.60) | 37.34 (0.52) | 50.42 (0.44) | 93.67 (0.38) | 93.67 (0.38) |
| | Bernoulli(0.5) | 17.73 (0.19) | 18.29 (0.18) | 17.01 (0.19) | 18.25 (0.23) | 18.03 (0.22) | 18.29 (0.18) |
| MCSIM-EO | $\beta(4,4)$ | 1.59 (0.12) | 10.23 (0.29) | 34.03 (0.46) | 60.56 (0.53) | 110.12 (0.50) | 110.12 (0.50) |
| | Uniform | 1.74 (0.06) | 1.76 (0.10) | 5.59 (0.19) | 14.59 (0.28) | 49.30 (0.32) | 49.30 (0.32) |
| | Bernoulli(0.5) | 9.97 (0.13) | 9.83 (0.12) | 9.01 (0.12) | 10.56 (0.16) | 7.46 (0.10) | 10.56 (0.16) |
| MCSIM-CV | $\beta(4,4)$ | 31.57 (3.13) | 32.75 (1.18) | 40.60 (0.67) | 56.81 (0.58) | 90.09 (0.45) | 90.09 (0.45) |
| | Uniform | 10.88 (0.25) | 16.21 (0.67) | 29.74 (0.72) | 38.91 (0.61) | 61.09 (0.36) | 61.09 (0.36) |
| | Bernoulli(0.5) | 17.96 (0.19) | 18.40 (0.18) | 16.86 (0.18) | 18.27 (0.22) | 16.88 (0.18) | 18.40 (0.18) |
| LDR | — | 17.86 (0.24) | 16.04 (0.18) | 17.67 (0.23) | 19.60 (0.29) | 20.95 (0.28) | 20.95 (0.28) |

### 7.3. Robustness Study

In our first computational study, we compare crashing policies obtained from MCSIM against our robust LDR policy. In particular, we investigate the sensitivity of MCSIM to the assumed shape of the activity time distribution, in networks with 10 activities.

We consider activity times drawn from five different distributions (data sets R0 through R4, described in the header row of Table 1), all of which have the same support and mean, but different distributional shapes. Specifically, in each data set, the activity times are first drawn from the specified distribution, all of which have support on the unit interval. They are then scaled and shifted so that they have support on $[0.2t, 1.8t]$, as described above. We note that the $\beta(4,4)$ distribution used in data set R0 is exactly the distribution that is assumed in PERT. For each data set, we compute the optimal crashing policy using various MCSIM benchmarks (i.e., problem (17)) by drawing $R$ samples from an assumed activity time distribution, which may differ from the true distribution of activity times in the data set. We compare the results against those from the LDR policy, which ignores the covariance matrix of the distributions used, to provide a fair comparison. We also repeat the computations in rolling horizon mode for MCSIM, assuming a uniform distribution. The unit crashing costs for each activity are drawn from a uniform distribution between 0 and 1, and the unit penalty cost is fixed at 8.

Table 1 tabulates the success percentages and conditional expected losses of the different crashing policies, implemented in static mode, with standard errors shown in parentheses. We also report the minimum success percentage and maximum conditional expected losses across all data sets, which represent the worst-case values of these respective metrics across the data sets. Our computational results demonstrate that the performance of MCSIM, using any objective, is typically sensitive to the choice of a particular distribution. Specifically, when the $\beta(4,4)$ or uniform distribution is assumed, there is a sharp degradation of success probabilities and conditional losses in data sets R3 and R4. However, for the distributions in these data sets, probability mass is concentrated around the extreme points of their support, which may not be representative of distributions encountered in practice. Our use of these data sets merely serves to illustrate the nonrobustness of MCSIM to arbitrarily chosen distributional assumptions. Conversely, our robust LDR policy maintains a high level of project success and low

**Table 2 Robustness Study: Rolling Horizon Mode**

| | | Success percentage | | | | | |
|---|---|---|---|---|---|---|---|
| Crash method | Assumed distribution | R0 $\beta(4,4)$ | R1 $\beta(2,2)$ | R2 $\beta(1,1)$ | R3 $\beta(0.5,0.5)$ | R4 Bernoulli(0.5) | Min |
| MCSIM | Uniform | 98.04 (0.62) | 98.49 (0.55) | 97.09 (0.75) | 93.13 (1.13) | 68.58 (2.08) | 68.58 (2.08) |
| MCSIM-EO | Uniform | 83.99 (1.64) | 86.56 (1.53) | 87.26 (1.49) | 79.75 (1.80) | 53.07 (2.23) | 53.07 (2.23) |
| MCSIM-CV | Uniform | 97.82 (0.65) | 98.35 (0.57) | 98.22 (0.59) | 96.51 (0.82) | 84.18 (1.63) | 84.18 (1.63) |
| RHLDR | — | 90.35 (1.32) | 89.14 (1.39) | 91.47 (1.25) | 91.39 (1.25) | 90.97 (1.28) | 89.14 (1.39) |
| | | Conditional expected loss | | | | | |
| Crash method | Assumed distribution | R0 $\beta(4,4)$ | R1 $\beta(2,2)$ | R2 $\beta(1,1)$ | R3 $\beta(0.5,0.5)$ | R4 Bernoulli(0.5) | Max |
| MCSIM | Uniform | 7.70 (0.19) | 12.68 (0.30) | 16.92 (0.38) | 25.56 (0.42) | 40.83 (0.27) | 40.83 (0.27) |
| MCSIM-EO | Uniform | 1.18 (0.04) | 1.24 (0.05) | 3.05 (0.10) | 6.49 (0.15) | 18.36 (0.16) | 18.36 (0.16) |
| MCSIM-CV | Uniform | 7.64 (0.15) | 10.82 (0.24) | 15.83 (0.43) | 24.66 (0.63) | 31.55 (0.36) | 31.55 (0.36) |
| RHLDR | — | 17.87 (0.24) | 16.10 (0.18) | 17.52 (0.23) | 19.56 (0.29) | 18.72 (0.25) | 19.56 (0.29) |

conditional losses across a range of distributions. For MCSIM runs that assume a Bernoulli distribution, performance is generally more stable across data sets R0 through R4, but success probabilities are uniformly poorer than for our LDR policy.

We also note that for the MCSIM policies computed under different objectives, MCSIM-EO tends to have lower success probabilities than MCSIM and MCSIM-CV, but also much lower conditional losses, which makes comparison with MCSIM and MCSIM-CV difficult. Between MCSIM-CV and MCSIM, it is clear that MCSIM-CV performs better, with higher success rates and lower conditional losses. This effect is most prominent under the $\beta(4,4)$ and uniform distributions. This is as expected, because the CVaR satisficing measure explicitly incorporates the budget $B$ into the objective, whereas MCSIM, which minimizes total expected cost, does not.

Table 2 shows that rolling horizon versions of the MCSIM policies generally improve their robustness, but there is still a degradation of performance depending on the choice of distribution. The LDR policy in rolling horizon mode, denoted by RHLDR, again maintains a more robust performance across different data sets than other policies. In particular, although MCSIM-EO achieves a modest improvement in conditional loss relative to RHLDR, it has a much lower success percentage.

### 7.4. Cross-Sectional Study

In our second test, we compare our proposed crashing policies, LDR and PWLDR, against the established crashing policies PERT and MCSIM, in both static and rolling horizon modes. To study the value of adaptability in our policies, we also compute performance for the NOADAPT policy, where model (8) is solved by replacing its last two constraints with $x_{mj} = 0$, $\forall m \in [M]$, $\forall j \in [N]$ and $y_{nj} = 0$, $\forall n \in [N]$, $\forall j \in [N]$. Consequently, in NOADAPT, all decisions are here-and-now decisions that do not depend on any uncertainty. We also report results from clairvoyant policies, where decisions are made after all uncertainties are revealed, as upper bounds on performance.

The data that we use are simulated under the structural assumption of a linear factor model (McCullagh and Nelder 1989). For each network, we draw 100 vectors of uncertain activity times, $\tilde{\mathbf{z}}$, from a distribution with structure

$$\tilde{\mathbf{z}} = \tilde{\boldsymbol{\epsilon}} + \mathbf{A}\tilde{\boldsymbol{\zeta}}, \tag{18}$$

where $\tilde{\boldsymbol{\epsilon}} \in \mathbb{R}^N$, $\tilde{\boldsymbol{\zeta}} \in \mathbb{R}^{N_d}$ are independent random vectors, and $\mathbf{A} \in \mathbb{R}^{N \times N_d}$ is a random coefficient matrix that stays fixed for each of the 100 draws. Each component of $\tilde{\boldsymbol{\epsilon}}$ is drawn from the distribution $Uni[0.2t, 0.8t]$, where $t$ is the deterministic activity time obtained from *RanGen*1, whereas each component of $\tilde{\zeta}$ takes value $\pm 1$ with equal probability. The coefficient matrix $\mathbf{A} = \text{diag}(\mathbf{t})\mathbf{S}$, where $\mathbf{t}$ is the vector of activity times from *RanGen*1, and $\mathbf{S}$ is a sparse matrix, with nonzero entries that are $\pm 1$ with equal probability. The premultiplication of $\mathbf{S}$ by $\text{diag}(\mathbf{t})$ ensures that each output activity time is appropriately scaled. The lower the sparsity of $\mathbf{S}$, the smaller the extent of correlation between activity times. In our tests, we use $N_d = 5$, and we vary the sparsity of the matrix $\mathbf{S}$ across data sets to study the effect of various extents of correlation between activities. The sparsity used in data sets C0 through C3 is respectively 0.0, 0.2, 0.5, and 0.8. The unit crashing costs are uniformly generated between 0 and 1, and a fixed unit penalty cost of 10 is used.

In the PERT policy and our LDR and PWLDR policies, we only assume that we have accurate information about the support, mean, and, for PWLDR policies, the covariance matrix of the activity times. Whereas, to compute the MCSIM benchmarks, we

**Table 3** Cross-Sectional Study: 10-Activity Networks in Static Mode

| Crash method | C0 | C1 | C2 | C3 |
|---|---|---|---|---|
| | Success percentage | | | |
| PERT | 35.27 (2.14) | 31.81 (2.08) | 28.76 (2.02) | 26.75 (1.98) |
| MCSIM | 99.18 (0.40) | 92.85 (1.15) | 91.12 (1.27) | 90.01 (1.34) |
| MCSIM-EO | 99.63 (0.27) | 91.49 (1.25) | 89.87 (1.35) | 91.53 (1.24) |
| MCSIM-CV | 99.99 (0.04) | 97.44 (0.71) | 96.58 (0.81) | 95.72 (0.91) |
| NOADAPT | 100.00 (0.00) | 82.55 (1.70) | 70.70 (2.04) | 71.40 (2.02) |
| LDR | 100.00 (0.00) | 88.97 (1.40) | 77.23 (1.88) | 72.76 (1.99) |
| PWLDR | 100.00 (0.00) | 90.26 (1.33) | 80.33 (1.78) | 74.91 (1.94) |
| | Conditional expected loss | | | |
| PERT | 136.06 (0.51) | 138.31 (0.52) | 139.98 (0.52) | 140.96 (0.52) |
| MCSIM | 26.77 (1.28) | 41.20 (0.67) | 43.09 (0.61) | 44.65 (0.60) |
| MCSIM-EO | 12.72 (1.02) | 8.50 (0.34) | 11.39 (0.36) | 17.36 (0.48) |
| MCSIM-CV | 18.07 (12.48) | 32.65 (0.99) | 36.48 (0.90) | 39.01 (0.88) |
| NOADAPT | —[a] (—)[a] | 17.30 (0.18) | 22.73 (0.19) | 26.46 (0.24) |
| LDR | —[a] (—)[a] | 16.73 (0.21) | 19.26 (0.16) | 19.40 (0.15) |
| PWLDR | —[a] (—)[a] | 16.45 (0.21) | 19.00 (0.18) | 18.48 (0.15) |

[a]The empirical conditional expected loss is undefined for these entries because the success percentage is 100%.

**Table 4** Cross-Sectional Study: 10-Activity Networks in Rolling Horizon Mode

| Crash method | C0 | C1 | C2 | C3 |
|---|---|---|---|---|
| | Success percentage | | | |
| PERT | 54.41 (2.23) | 48.14 (2.23) | 44.80 (2.22) | 41.99 (2.21) |
| MCSIM | 99.98 (0.06) | 97.47 (0.70) | 96.01 (0.88) | 95.15 (0.96) |
| MCSIM-EO | 99.89 (0.15) | 90.36 (1.32) | 89.47 (1.37) | 89.57 (1.37) |
| MCSIM-CV | 100.00 (0.00) | 98.05 (0.62) | 96.66 (0.80) | 96.17 (0.86) |
| RHNOADAPT | 100.00 (0.00) | 82.51 (1.70) | 71.70 (2.01) | 73.23 (1.98) |
| RHLDR | 100.00 (0.00) | 88.88 (1.41) | 77.26 (1.87) | 72.90 (1.99) |
| RHPWLDR | 100.00 (0.00) | 90.12 (1.33) | 80.28 (1.78) | 74.80 (1.94) |
| CLVT | 100.00 (0.00) | 100.00 (0.02) | 99.94 (0.11) | 99.79 (0.20) |
| CLVT-EO | 98.48 (0.55) | 95.13 (0.96) | 92.47 (1.18) | 90.15 (1.33) |
| CLVT-CV | 100.00 (0.00) | 100.00 (0.02) | 99.94 (0.11) | 99.79 (0.20) |
| | Conditional expected loss | | | |
| PERT | 63.88 (0.28) | 71.59 (0.32) | 70.77 (0.32) | 72.16 (0.33) |
| MCSIM | 23.25 (7.55) | 19.28 (0.60) | 22.73 (0.53) | 26.15 (0.62) |
| MCSIM-EO | 2.08 (0.27) | 3.82 (0.13) | 6.33 (0.18) | 8.93 (0.26) |
| MCSIM-CV | —[a] (—)[a] | 13.16 (0.48) | 16.07 (0.49) | 20.58 (0.65) |
| RHNOADAPT | —[a] (—)[a] | 17.69 (0.19) | 21.53 (0.19) | 23.61 (0.24) |
| RHLDR | —[a] (—)[a] | 16.59 (0.21) | 19.22 (0.16) | 19.12 (0.14) |
| RHPWLDR | —[a] (—)[a] | 16.29 (0.21) | 18.56 (0.17) | 18.05 (0.14) |
| CLVT | —[a] (—)[a] | 32.77 (0.00) | 33.08 (5.80) | 36.14 (3.15) |
| CLVT-EO | 0.00 (0.00) | 0.00 (0.00) | 0.23 (0.06) | 0.77 (0.10) |
| CLVT-CV | —[a] (—)[a] | 6.26 (0.00) | 28.19 (5.77) | 36.13 (3.15) |

[a]The empirical conditional expected loss is undefined for these entries because the success percentage is 100%.

assume full knowledge of the true distribution and of the structural equation (18), which are highly optimistic and unrealistic assumptions in practice.

The results of our computational study for the various crashing policies in 10-activity networks using static and rolling horizon modes, respectively, are shown in Tables 3 and 4. Using both success probability and conditional loss metrics, our LDR and PWLDR policies improve substantially over the PERT-based policies. As in the robustness study, the MCSIM-CV policy tends to perform better across all data sets than MCSIM. In static mode, our policies are comparable with MCSIM policies. Specifically, MCSIM policies typically incur more than twice the conditional losses of our policies, whereas our policies yield lower success probabilities than the MCSIM policies. In rolling horizon mode, our polices perform worse than MCSIM on both metrics, because adaptability to uncertainties is not well exploited in small 10-activity networks. This is further substantiated by the fact that the NOADAPT policies perform similarly to the LDR. We hypothesize that this will not occur in larger networks. The presence of correlation degrades the average performance of all crashing policies, in both static and rolling horizon modes. In the static case, the PWLDR policy performs better than its LDR counterpart, because the PWLDR policy incorporates covariance information, whereas the LDR policy does not.

We repeat our computational tests using 20-activity networks. We use a similar data generation procedure. However, we use a unit penalty cost of 20, because we expect larger projects to have a proportionately larger penalty per unit of delay, and we scale the coefficient matrix **A** by a factor of 2. We summarize our findings in Tables 5 and 6.

In both static and rolling horizon modes, all policies including the clairvoyant CLVT policy decline in performance as correlation increases. Our policies

**Table 5** Cross-Sectional Study: 20-Activity Networks in Static Mode

| Crash method | C0 | C1 | C2 | C3 |
|---|---|---|---|---|
| | Success percentage | | | |
| PERT | 24.04 (3.02) | 13.11 (2.39) | 6.52 (1.75) | 4.76 (1.51) |
| MCSIM | 98.51 (0.86) | 95.86 (1.41) | 88.12 (2.29) | 77.06 (2.97) |
| MCSIM-EO | 93.45 (1.75) | 97.39 (1.13) | 87.20 (2.36) | 76.00 (3.02) |
| MCSIM-CV | 99.98 (0.11) | 98.47 (0.87) | 90.28 (2.09) | 78.09 (2.93) |
| NOADAPT | 100.00 (0.00) | 88.89 (2.22) | 47.89 (3.53) | 27.41 (3.15) |
| LDR | 100.00 (0.00) | 98.56 (0.84) | 85.59 (2.48) | 69.50 (3.26) |
| PWLDR | 100.00 (0.00) | 99.01 (0.70) | 89.98 (2.12) | 77.48 (2.95) |
| | Conditional expected loss | | | |
| PERT | 147.13 (0.77) | 165.89 (0.82) | 174.08 (0.78) | 179.35 (0.77) |
| MCSIM | 28.11 (1.54) | 30.75 (0.96) | 46.72 (0.89) | 58.87 (0.78) |
| MCSIM-EO | 0.27 (0.10) | 28.75 (1.10) | 42.22 (0.84) | 56.35 (0.76) |
| MCSIM-CV | 22.63 (6.63) | 28.54 (1.52) | 48.47 (1.02) | 59.77 (0.81) |
| NOADAPT | —[a] (—)[a] | 48.89 (0.87) | 79.87 (0.60) | 106.84 (0.61) |
| LDR | —[a] (—)[a] | 35.14 (1.79) | 52.39 (0.92) | 66.46 (0.73) |
| PWLDR | —[a] (—)[a] | 33.20 (1.99) | 49.72 (1.04) | 62.24 (0.81) |

[a]The empirical conditional expected loss is undefined for these entries because the success percentage is 100%.

**Table 6** Cross-Sectional Study: 20-Activity Networks in Rolling Horizon Mode

| Crash method | C0 | C1 | C2 | C3 |
|---|---|---|---|---|
| | | Success percentage | | |
| PERT | 40.95 (3.48) | 27.63 (3.16) | 17.81 (2.71) | 13.88 (2.44) |
| MCSIM | 99.99 (0.07) | 98.56 (0.84) | 90.55 (2.07) | 78.39 (2.91) |
| MCSIM-EO | 90.43 (2.08) | 97.58 (1.09) | 89.58 (2.16) | 77.67 (2.95) |
| MCSIM-CV | 100.00 (0.05) | 98.89 (0.74) | 91.25 (2.00) | 78.75 (2.89) |
| RHNOADAPT | 100.00 (0.00) | 89.75 (2.14) | 51.88 (3.53) | 31.94 (3.30) |
| RHLDR | 100.00 (0.00) | 98.79 (0.77) | 89.10 (2.20) | 76.39 (3.00) |
| RHPWLDR | 100.00 (0.00) | 99.06 (0.68) | 90.59 (2.06) | 78.25 (2.92) |
| CLVT | 100.00 (0.00) | 99.69 (0.39) | 95.75 (1.43) | 87.23 (2.36) |
| CLVT-EO | 99.91 (0.21) | 93.09 (1.79) | 79.12 (2.87) | 65.90 (3.35) |
| CLVT-CV | 100.00 (0.00) | 99.69 (0.39) | 95.75 (1.43) | 87.23 (2.36) |
| | | Conditional expected loss | | |
| PERT | 72.79 (0.46) | 83.56 (0.52) | 96.47 (0.57) | 107.80 (0.61) |
| MCSIM | 12.02 (5.55) | 30.47 (1.52) | 49.21 (1.06) | 60.27 (0.82) |
| MCSIM-EO | 0.02 (0.01) | 14.27 (1.08) | 44.74 (0.98) | 58.37 (0.80) |
| MCSIM-CV | 0.13 (0.00) | 26.63 (1.92) | 50.85 (1.11) | 60.94 (0.83) |
| RHNOADAPT | —[a] (—)[a] | 57.75 (1.24) | 84.29 (0.73) | 106.39 (0.68) |
| RHLDR | —[a] (—)[a] | 33.39 (1.81) | 47.10 (1.00) | 58.68 (0.79) |
| RHPWLDR | —[a] (—)[a] | 33.78 (2.26) | 50.38 (1.09) | 62.11 (0.84) |
| CLVT | —[a] (—)[a] | 27.04 (2.95) | 47.54 (1.51) | 55.53 (1.00) |
| CLVT-EO | 0.00 (0.00) | 1.19 (0.20) | 9.67 (0.43) | 20.79 (0.50) |
| CLVT-CV | —[a] (—)[a] | 27.04 (2.95) | 47.54 (1.51) | 55.53 (1.00) |

[a]The empirical conditional expected loss is undefined for these entries because the success percentage is 100%.

perform similarly to MCSIM policies in both modes, even though MCSIM has full distributional information. Compared with the CLVT and CLVT-CV clairvoyant policies, even on the most challenging data set, C3, our rolling horizon RHLDR and RHPWLDR policies register only about a 10% reduction in success probability, and comparable conditional losses. In static mode, the LDR policy performs worse than the PWLDR policy, especially as correlation increases, which is as expected. In rolling horizon mode, the RHLDR policy remains worse than the RHPWLDR policy, albeit by a smaller margin. As we hypothesize above, for these larger 20-activity networks, NOADAPT performs poorly in static mode. In rolling horizon mode, NOADAPT improves, but does not perform at the level of LDR or PWLDR. Finally, PERT is uniformly dominated by our policies in all data sets and in both operating modes.

The results from the robustness and cross-sectional studies provide several insights, which we now briefly discuss. First, using the probability of project success and conditional expected losses as performance metrics, the CVaR satisficing measure is a better objective than total cost. Second, our policies are significantly more robust to the true distribution of uncertainties than Monte Carlo simulation methods are. Third, even in the highly optimistic scenario where the Monte Carlo simulation method somehow guesses the right distribution, the ability of our policies to adapt crashing decisions to previously revealed uncertainties compensates for the difference in distributional knowledge, thereby performing similarly to the MCSIM policies on the above two performance metrics. Moreover, the relative value of adapted crashing decisions improves as the size of the network increases. Fourth, between our policies, the PWLDR typically performs better than the LDR, and especially so when the activity times are highly correlated, because the PWLDR policy incorporates covariance information, whereas the LDR policy does not.

## 8. Concluding Remarks

In this paper, we study the problem of project planning with crashing under uncertainty. We consider a satisficing objective that models penalties for late completion as well as crashing and overhead cost, relative to an overall project budget. We describe a linear decision rule and a piecewise-linear decision rule that are computed at the project planning stage and that use updated information about activity completion times during project execution. We also describe a rolling horizon procedure that uses periodic information updates during the execution of activities to find new decision rules. Our computational studies demonstrate that our LDR and PWLDR crashing policies have success probabilities and budget overruns that strongly outperform PERT. Our policies are also competitive with MCSIM policies that use the very strong assumption of full distributional knowledge. Moreover, the performance of our policies even approaches that from a clairvoyant policy, where decisions are made after all uncertainties are revealed. Our policies have the further advantage of robustness against specific activity time distributions, a feature that is typically absent from MCSIM policies.

The data requirements of our models are not excessive or unusual. We assume only that the activity times come from a partially specified distribution within a family of distributions that is described by any one or more of the following details about the uncertainties: support, mean, and covariance. Covariance information is implied by common resource usage between projects, which in practice is typically identified during preliminary project evaluation. Furthermore, we do not rely on knowledge of an accurate probability distribution for uncertainties, as is needed in Monte Carlo methods.

The contributions of our work for managers, relative to current practice, are as follows. Currently, most companies use one of three methodologies to estimate project duration: (a) PERT, possibly with a locally estimated adjustment factor to allow for the underestimation problem; (b) Monte Carlo simulation; or (c) informal planning based on a deterministic schedule from project management software, possibly with an adjustment factor to allow for uncertainty. Crashing decisions are often made ad hoc, based on project progress and resource availability. We propose an alternative methodology that does not rely on knowledge of probability distributions for activity times, or a crude approximation to them. Our methodology incorporates correlations that are well known to exist, but rarely considered, in current decision-making practice. Although the above theoretical justification for our methodology is in part quite complex, a simple implementation methodology is described by Goh and Sim (2011) using the software package ROME. Most importantly, the recommended crashing decisions lead to a much higher probability of project success, and a much lower expected budget overrun, than the currently used methodologies.

Several extensions of the problem we consider are relevant for future research. The first extension is the modeling of discrete crashing resources, for example, the outsourcing of an entire activity, within a robust optimization framework. Related to this is the modeling of discrete crash cost functions for activities, for example, when adding an extra production shift to meet a project deadline. Another practical generalization is the modeling of discrete penalty functions for project completion time, such as sometimes arise in large public construction projects (Philips 2005). Unfortunately, many projects suffer from a loss of resources during their execution stage, due to changing senior management priorities. Hence, another useful generalization is the modeling of uncertainty about the availability of resources for crashing. Still more generally, the study of multiple projects that share crashing resources would be a valuable contribution. In conclusion, we hope that our work will encourage the development of effective robust optimization approaches to these important project management planning problems.

## Acknowledgments

The authors thank department editor Dimitris Bertsimas, the anonymous associate editor, and three referees for their insightful comments and suggestions on this paper. Their input led to significant improvements in the authors' work. The authors also thank Melvyn Sim (National University of Singapore) for substantial discussions regarding development of the models presented. Helpful comments on an earlier draft were provided by Zhuoyu Long (National University of Singapore) and Jin Qi (National University of Singapore).

## Appendix

Proof of Proposition 1. We demonstrate the sufficient conditions of Sion's (1958) minimax theorem to justify the interchange of the $\inf(\cdot)$ and $\sup(\cdot)$ operators. To begin, define $\tilde{X} := B - \tilde{V}$ and

$$f(\alpha, \mathbb{P}) := \mathrm{E}_{\mathbb{P}}(\min\{1, \alpha\tilde{X}\}).$$

First, choose some $\mathbb{P} \in \mathbb{F}$ such that $\epsilon := \mathrm{E}_{\mathbb{P}}(\tilde{X}^-) > 0$. Suppose for a contradiction that $\alpha > 2\epsilon^{-1}$. Then,

$$\begin{aligned}\inf_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(\min\{1, \alpha\tilde{X}\}) &\le \mathrm{E}_{\mathbb{P}}(\min\{1, \alpha\tilde{X}\})\\ &= \mathbb{P}(\alpha\tilde{X} > 1) + \mathrm{E}_{\mathbb{P}}(\alpha\tilde{X}\mathbb{1}_{\{0<\alpha\tilde{X}\le 1\}})\\ &\quad + \alpha\mathrm{E}_{\mathbb{P}}(\tilde{X}\mathbb{1}_{\{\alpha\tilde{X}\le 0\}})\\ &\le 2 - \alpha\epsilon\\ &< 0.\end{aligned}$$

Since choosing $\alpha = 0$ gives $f(0, \mathbb{P}) = 0$ for any $\mathbb{P} \in \mathbb{F}$, we may require $0 \le \alpha \le 2\epsilon^{-1}$. Hence, $\alpha$ is optimized over a compact set. Second, fixing any $\mathbb{P} \in \mathbb{F}$, $f(\cdot, \mathbb{P})$ is easily seen to be concave and continuous. Third, viewing $\mathbb{P}$ as an element of the linear space of all finite measures on $(\mathbb{R}^N, \mathscr{B}(\mathbb{R}^N))$, $f(\alpha, \cdot)$ is linear and thus convex. Moreover, for any feasible $\alpha$, note that by assumption there exists some $\underline{X} \in \mathbb{R}$ such that $\tilde{X} \ge \underline{X}$ a.s., for any $\mathbb{P} \in \mathbb{F}$. This implies that for each $\mathbb{P} \in \mathbb{F}$, $f$ has a more explicit representation,

$$f(\alpha, \mathbb{P}) = \mathrm{E}_{\mathbb{P}}(\max\{\min\{1, \alpha\tilde{X}\}, \alpha\underline{X}\}).$$

Hence, the integrand is continuous and bounded in $\tilde{X}$. Thus, if we have a sequence of measures $\{\mathbb{P}_n\}_{n\in\mathbb{N}}$ such that $\mathbb{P}_n$ converges to $\mathbb{P}$ in total variation, then $\mathbb{P}_n$ converges to $\mathbb{P}$ weakly, which in turn implies that $f(\alpha, \mathbb{P}_n) \to f(\alpha, \mathbb{P})$, i.e., $f(\alpha, \cdot)$ is continuous. □

Proof of Theorem 2. By construction, the decision rules $\mathbf{x}^*$ and $\mathbf{y}^*$ satisfy the nonanticipativity requirements. Formally, for each $m \in [M]$, $n \in [N]$, we have $x^*_m \in \mathscr{Y}(1, N, I^x_m)$, $y^*_n \in \mathscr{Y}(1, N, I^y_n)$.

The second constraint of model (6) is satisfied directly from the construction of $\mathbf{y}^*(\tilde{\mathbf{z}})$. The third constraint is satisfied because for each $m \in [M]$, we have $m \in \mathscr{N}(m)$, and hence $x^*_m(\tilde{\mathbf{z}}) \ge \alpha^{-1}x_m(\tilde{\mathbf{z}}) + \alpha^{-1}(x_m(\tilde{\mathbf{z}}))^- = \alpha^{-1}(x_m(\tilde{\mathbf{z}}))^+ \ge 0$.

To show that the first constraint of model (6) is satisfied, we consider an arbitrary arc $n \in [N]$ on the network. Writing $(i, j) = (i_n, j_n)$, we first observe that $\mathscr{N}(i) \subset \mathscr{N}(j)$, and $\mathscr{A}(i) \subset \mathscr{A}(j)$. We have

$$\begin{aligned}&\alpha(x^*_j(\tilde{\mathbf{z}}) - x^*_i(\tilde{\mathbf{z}}) + y^*_n(\tilde{\mathbf{z}}))\\ &\quad\ge x_j(\tilde{\mathbf{z}}) - x_i(\tilde{\mathbf{z}}) + (y_n(\tilde{\mathbf{z}}) - \alpha u_n)^+\\ &\qquad + (\alpha\tilde{z}_n - y_n(\tilde{\mathbf{z}}) - x_j(\tilde{\mathbf{z}}) + x_i(\tilde{\mathbf{z}}))^+ + \alpha y^*_n(\tilde{\mathbf{z}})\\ &\quad= x_j(\tilde{\mathbf{z}}) - x_i(\tilde{\mathbf{z}}) + (y_n(\tilde{\mathbf{z}}) - \alpha u_n)^+\\ &\qquad + (\alpha\tilde{z}_n - y_n(\tilde{\mathbf{z}}) - x_j(\tilde{\mathbf{z}}) + x_i(\tilde{\mathbf{z}}))^+\\ &\qquad + y_n(\tilde{\mathbf{z}}) + (y_n(\tilde{\mathbf{z}}))^- - (y_n(\tilde{\mathbf{z}}) - \alpha u_n)^+\\ &\quad\ge \alpha\tilde{z}_n + (x_j(\tilde{\mathbf{z}}) - x_i(\tilde{\mathbf{z}}) + y_n(\tilde{\mathbf{z}}) - \alpha\tilde{z}_n)\\ &\qquad + (\alpha\tilde{z}_n - y_n(\tilde{\mathbf{z}}) - x_j(\tilde{\mathbf{z}}) + x_i(\tilde{\mathbf{z}}))^+\\ &\quad= \alpha\tilde{z}_n + (x_j(\tilde{\mathbf{z}}) - x_i(\tilde{\mathbf{z}}) + y_n(\tilde{\mathbf{z}}) - \alpha\tilde{z}_n)^+\\ &\quad\ge \alpha\tilde{z}_n.\end{aligned} \tag{19}$$

The first inequality results from the inclusions $\mathscr{N}(i) \subset \mathscr{N}(j)$ and $\mathscr{A}(i) \subset \mathscr{A}(j)$, which cancel out many of the common terms. The remainder of the terms are nonnegative and give rise to the inequality. The final equation follows from $x + x^- \equiv x^+$. Similarly, we have

$$\begin{aligned}\alpha(x_j^*(\tilde{\mathbf{z}}) - x_i^*(\tilde{\mathbf{z}})) &\geq x_j(\tilde{\mathbf{z}}) - x_i(\tilde{\mathbf{z}}) + (-x_j(\tilde{\mathbf{z}}) + x_i(\tilde{\mathbf{z}}))^+ \\ &= (x_j(\tilde{\mathbf{z}}) - x_i(\tilde{\mathbf{z}}))^+ \\ &\geq 0. \end{aligned} \tag{20}$$

Together, (19) and (20) imply the first constraint of (6), $x_j^*(\tilde{\mathbf{z}}) - x_i^*(\tilde{\mathbf{z}}) \geq (\tilde{z}_n - y_n^*(\tilde{\mathbf{z}}))^+$. Because all the constraints are satisfied, the piecewise-linear decision rules $\mathbf{x}^*$ and $\mathbf{y}^*$ are feasible in (6). □

Proof of Theorem 3. We prove a preliminary result.

Lemma 1. *Any convex piecewise-linear function of a scalar-valued decision rule, with any information index set $I$, i.e., $x \in \mathcal{Y}(1, N, I)$, can be equivalently expressed as*

$$\max_{k\in[K]}\{a_k x(\tilde{\mathbf{z}}) + b_k\} = \min_{t\in\mathcal{Y}(1,N,[N])} t(\tilde{\mathbf{z}}) + \sum_{k=1}^{K}(a_k x(\tilde{\mathbf{z}}) + b_k - t(\tilde{\mathbf{z}}))^+.$$

Proof of Lemma 1. Fix any $\mathbf{z} \in \mathcal{W}$ and $x \in \mathcal{Y}(1, N, I)$. It suffices to prove the lemma pointwise, i.e., that for each $v \in \mathbb{R}$ we have

$$\max_{k\in[K]}\{a_k v + b_k\} = \min_{t\in\mathbb{R}}\left\{t + \sum_{k=1}^{K}(a_k v + b_k - t)^+\right\}. \tag{21}$$

Fix any $v \in \mathbb{R}$. Let $k^* := \arg\max_{k\in[K]}\{a_k v + b_k\}$. For any $t \in \mathbb{R}$, $t + \sum_{k=1}^{K}(a_k v + b_k - t)^+ \geq t + (a_{k^*} v + b_{k^*} - t)^+ \geq a_{k^*} v + b_{k^*}$. Hence, in (21), the left-hand side is less than or equal to the right-hand side. Conversely, suppose for a contradiction that the optimal $t^*$ on the right-hand side is $t^* := a_{k^*} v + b_{k^*} + \eta$ for some $\eta > 0$. The point $t := a_{k^*} v + b_{k^*} + \eta/2$ has a smaller objective, establishing the contradiction. □

To prove Theorem 3, we note that given a feasible piecewise-linear decision rule $(\mathbf{x}^*, \mathbf{y}^*)$ obtained from model (9), Theorem 2 shows that it is also feasible in (6). Under this decision rule, the objective becomes

$$\begin{aligned} 1 - Z_0^* &= \min \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}\Bigg(\Bigg(1 - \alpha B + \alpha \sum_{n=1}^{N} \max_{\ell\in[L]}\{c_{n\ell} y_n^*(\tilde{\mathbf{z}}) + d_{n\ell}\} \\ &\qquad + \alpha \max_{k\in[K]}\{p_k x_M^*(\tilde{\mathbf{z}}) - p_k \tau + q_k\}\Bigg)^+\Bigg) \\ &\leq \min \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}\Bigg(\Bigg(1 - \alpha B + \sum_{n=1}^{N} \max_{\ell\in[L]}\{c_{n\ell} y_n(\tilde{\mathbf{z}}) + \alpha d_{n\ell}\} \\ &\qquad + \max_{k\in[K]}\{p_k x_M(\tilde{\mathbf{z}}) - \alpha p_k \tau + \alpha q_k\} \\ &\qquad + \sum_{n\in\mathcal{A}(M)} c_{nL}(y_n(\tilde{\mathbf{z}}))^- \\ &\qquad + \sum_{n\in\mathcal{A}(M)} (p_K - c_{k1})^+ (y_n(\tilde{\mathbf{z}}) - \alpha u_n)^+ \\ &\qquad + p_K \sum_{n\in\mathcal{A}(M)} (\alpha\tilde{z}_n - y_n(\tilde{\mathbf{z}}) - x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}))^+ \\ &\qquad + p_K \sum_{n\in\mathcal{A}(M)} (-x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}))^+ \\ &\qquad + p_K \sum_{m\in\mathcal{N}(M)} (x_m(\tilde{\mathbf{z}}))^-\Bigg)^+\Bigg) \\ &= \min \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}\Bigg(\Bigg(1 - \alpha B + \sum_{n=1}^{N} s_n(\tilde{\mathbf{z}}) + t(\tilde{\mathbf{z}}) \\ &\qquad + \sum_{n=1}^{N}\sum_{\ell=1}^{L}(c_{n\ell} y_n(\tilde{\mathbf{z}}) + \alpha d_{n\ell} - s_n(\tilde{\mathbf{z}}))^+ \\ &\qquad + \sum_{k=1}^{K}(p_k x_M(\tilde{\mathbf{z}}) - \alpha p_k \tau + \alpha q_k - t(\tilde{\mathbf{z}}))^+ \\ &\qquad + \sum_{n=1}^{N} c_{nL}(y_n(\tilde{\mathbf{z}}))^- \\ &\qquad + \sum_{n=1}^{N} (p_K - c_{n1})^+ (y_n(\tilde{\mathbf{z}}) - \alpha u_n)^+ \\ &\qquad + p_K \sum_{n=1}^{N} (\alpha\tilde{z}_n - y_n(\tilde{\mathbf{z}}) - x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}))^+ \\ &\qquad + p_K \sum_{n=1}^{N} (-x_{j_n}(\tilde{\mathbf{z}}) + x_{i_n}(\tilde{\mathbf{z}}))^+ \\ &\qquad + p_K \sum_{m=1}^{M} (x_m(\tilde{\mathbf{z}}))^-\Bigg)^+\Bigg) \\ &\leq 1 - Z_{\mathrm{PWLDR}}^*. \end{aligned}$$

The first inequality results from applying the definitions of $(\mathbf{x}^*, \mathbf{y}^*)$, the assumptions $p_K \geq p_k$, $\forall\, k \in [K]$ and $c_{nL} \geq c_{n\ell} \geq c_{n1}$, $\forall\, \ell \in [L]$, $n \in [N]$, and the inequality $(p_K - c_{n1})^+ \geq (p_K - c_{n1})$. The second equality follows from Lemma 1, and uses auxiliary decision rules $\mathbf{s} \in \mathcal{Y}(N, N, [N])$ and $t \in \mathcal{Y}(1, N, [N])$. Also, we use the fact that, because $M$ is the terminal node, $\mathcal{N}(M) = [M]$ and $\mathcal{A}(M) = [N]$, which simplifies the summations. Finally, the last inequality results from adding and subtracting another auxiliary decision rule $r \in \mathcal{Y}(1, N, [N])$ and applying subadditivity of the $(\cdot)^+$ and $\sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(\cdot)$ operators.

We now show that $Z_{\mathrm{PWLDR}}^* \geq Z_{\mathrm{LDR}}$. For any feasible solution to (8), we form the following affine functions for each $m \in [M]$ and $n \in [N]$: $x_m(\tilde{\mathbf{z}}) = x_m^0 + \mathbf{x}_m'\tilde{\mathbf{z}}$, $y_n(\tilde{\mathbf{z}}) = y_n^0 + \mathbf{y}_n'\tilde{\mathbf{z}}$, $s_n(\tilde{\mathbf{z}}) = s_n^0 + \mathbf{s}_n'\tilde{\mathbf{z}}$, $r(\tilde{\mathbf{z}}) = r^0 + \mathbf{r}'\tilde{\mathbf{z}}$, and $t(\tilde{\mathbf{z}}) = t^0 + \mathbf{t}'\tilde{\mathbf{z}}$. These functions are feasible for (9). Moreover, because the inequalities of (8) are satisfied by a feasible solution, the nonlinear terms in the objective of (9) vanish, and the expression for the objective coincides with (8). Hence, we have $Z_0{}^* \geq Z_{\mathrm{PWLDR}}^* \geq Z_{\mathrm{LDR}}$, as required. □

Proof of Theorem 4. We establish two preliminary results.

Lemma 2. *Consider a scalar-valued LDR with an arbitrary information index set, $y \in \mathcal{L}(1, N, I)$.*

1. *If $y(\mathbf{z}) = y^0 + \mathbf{y}'\mathbf{z} \leq 0$, $\forall\, \mathbf{z} \in \mathcal{W}$, then $\pi^1(y^0, \mathbf{y}) \leq 0$.*
2. *If $y(\mathbf{z}) = y^0 + \mathbf{y}'\mathbf{z} \geq 0$, $\forall\, \mathbf{z} \in \mathcal{W}$, then $\pi^1(y^0, \mathbf{y}) \leq \sup_{\hat{\mathbf{z}}\in\hat{\mathcal{W}}}\{y^0 + \mathbf{y}'\hat{\mathbf{z}}\}$.*

Proof. Applying the definition of $\pi^1(y^0, \mathbf{y})$, the assumption $y(\mathbf{z}) \leq 0$, $\forall\, \mathbf{z} \in \mathcal{W}$, implies that $\pi^1(y^0, \mathbf{y}) = \inf_{\mathbf{s}\in\mathbb{R}^N}(\sup_{\hat{\mathbf{z}}\in\hat{\mathcal{W}}}\{\mathbf{s}'\hat{\mathbf{z}}\} + \sup_{\mathbf{z}\in\mathcal{W}}\{-\mathbf{s}'\mathbf{z}\})$. Because $\mathbf{s} = \mathbf{0}$ is feasible, we have $\pi^1(y^0, \mathbf{y}) \le 0$. Similarly, the assumption $y(\mathbf{z}) \ge 0,\ \forall \mathbf{z} \in \mathcal{W}$ implies that $\pi^1(y^0, \mathbf{y}) = \inf_{\mathbf{s}\in\mathbb{R}^N} (\sup_{\hat{\mathbf{z}}\in\hat{\mathcal{W}}}\{\mathbf{s}'\hat{\mathbf{z}}\} + \sup_{\mathbf{z}\in\mathcal{W}}\{y^0 + \mathbf{y}'\mathbf{z} - \mathbf{s}'\mathbf{z}\})$. Choosing $\mathbf{s} = \mathbf{y}$ yields $\pi^1(y^0, \mathbf{y}) \le \sup_{\hat{\mathbf{z}}\in\hat{\mathcal{W}}}\{y^0 + \mathbf{y}'\hat{\mathbf{z}}\}$. □

Lemma 3. *Consider a scalar-valued LDR with an arbitrary information index set, $y \in \mathcal{L}(1, N, I)$. If $y(\mathbf{z}) = y^0 + \mathbf{y}'\mathbf{z} \le 0$, $\forall \mathbf{z} \in \mathcal{W}$, then $\pi(y^0, \mathbf{y}) = 0$.*

Proof. Note that $y(\mathbf{z}) \le 0,\ \forall \mathbf{z} \in \mathcal{W}$ implies that $\sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((y^0 + \mathbf{y}'\tilde{\mathbf{z}})^+) = 0$. Using Proposition 2, we have $\pi(y^0, \mathbf{y}) \ge 0$. Moreover,

$$\begin{aligned}\pi(y^0, \mathbf{y}) &= \min_{x^0, \mathbf{x}} \pi^1(x^0, \mathbf{x}) + \pi^2(y^0 - x^0, \mathbf{y} - \mathbf{x})\\ &\le \pi^1(y^0, \mathbf{y}) + \pi^2(0, \mathbf{0}) \quad [\text{choosing } (x^0, \mathbf{x}) = (y^0, \mathbf{y})]\\ &= \pi^1(y^0, \mathbf{y}) \quad [\text{by positive homogeneity of } \pi^2(\cdot)]\\ &\le 0\end{aligned}$$

(by Lemma 2, part 1). Hence, we have $\pi(y^0, \mathbf{y}) = 0$. □

To prove Theorem 4, consider a feasible solution to (8). Observe that this solution is also feasible in (10). The objective of (10) is simplified by iteratively applying Lemma 3 to each constraint of model (8), which causes all the $\pi(\cdot)$ terms except the first to vanish. Thus, we have

$$\begin{aligned}1 - Z_{\mathrm{PWLDR}} &\le \pi(r^0, \mathbf{r})\\ &\le \pi^1(r^0, \mathbf{r}) \quad (\text{from the proof of Lemma 3})\\ &\le \sup_{\hat{\mathbf{z}}\in\hat{\mathcal{W}}}\{r^0 + \mathbf{r}'\hat{\mathbf{z}}\} \quad (\text{by Lemma 2, part 2})\\ &= 1 - Z_{\mathrm{LDR}}.\end{aligned}$$

Therefore, we have $Z_{\mathrm{PWLDR}} \ge Z_{\mathrm{LDR}}$. The second inequality $Z^*_{\mathrm{PWLDR}} \ge Z_{\mathrm{PWLDR}}$ follows, first from applying subadditivity of the $\sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(\cdot)$ operator to obtain

$$\begin{aligned}&\sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(\mathbf{c}_L'(\mathbf{y}(\tilde{\mathbf{z}}))^- + (p_K - \mathbf{c}_1')^+(\mathbf{y}(\tilde{\mathbf{z}}) - \alpha\mathbf{u})^+)\\ &\quad\le \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}(\mathbf{c}_L'(\mathbf{y}(\tilde{\mathbf{z}}))) + \sup_{\mathbb{P}\in\mathbb{F}} \mathrm{E}_{\mathbb{P}}((p_K - \mathbf{c}_1')^+(\mathbf{y}(\tilde{\mathbf{z}}) - \alpha\mathbf{u})^+)\end{aligned}$$

in the objective of (9), and second from iteratively applying Proposition 2 to bound the respective terms in the objective of (9) from above. The result then follows from Theorem 3. □

## References

Adler PS, Mandelbaum A, Nguyen V, Schwerer E (1995) From project to process management: An empirically-based framework for analyzing product development time. *Management Sci.* 41(3):458–484.

Allais M (1953) Le comportement de l'homme rationnel devant le risque, critique des postulats et axiomes de l'école Américaine. *Econometrica* 21(4):503–546.

Barbour AD, Jensen JL (1989) Local and tail approximations near the Poisson limit. *Scandinavian J. Statist.* 16(1):75–87.

Ben-Tal A, Nemirovski A (1998) Robust convex optimization. *Math. Oper. Res.* 23(4):769–805.

Ben-Tal A, Goryashko A, Guslitzer E, Nemirovski A (2004) Adjustable robust solutions of uncertain linear programs. *Math. Programming* 99(2):351–376.

Ben-Yair A (2003) Justification of using beta-distribution in project management. *17th IPMA World Congress on Project Management (IPMA-2003), June 3–6, Moscow*, 293–298.

Bertsimas D, Popescu I (2005) Optimal inequalities in probability theory: A convex optimization approach. *SIAM J. Optim.* 15(3):780–804.

Bertsimas D, Sim M (2004) The price of robustness. *Oper. Res.* 52(1):35–53.

Black K (2008) *Business Statistics: Contemporary Decision Making*, 5th ed. (Wiley, Hoboken, NJ).

Bowman RA (1994) Stochastic gradient-based time-cost tradeoffs in PERT networks using simulation. *Ann. Oper. Res.* 53(1):533–551.

Brown DB, Sim M (2009) Satisficing measures for analysis of risky positions. *Management Sci.* 55(1):71–84.

Brown DB, De Giorgi E, Sim M (2012) Aspirational preferences and their representation by risk measures. *Management Sci.* 58(11): 2095–2113.

Chau KW (1995) The validity of the triangular distribution assumption in Monte Carlo simulation of construction costs: Empirical evidence from Hong Kong. *Construction Management Econom.* 13(1):15–21.

Chen X, Sim M, Sun P, Zhang J (2008) A linear decision-based approximation approach to stochastic programming. *Oper. Res.* 56(2):344–357.

Cohen I, Golany B, Shtub A (2007) The stochastic time-cost tradeoff problem: A robust optimization approach. *Networks* 49(2):175–188.

Cohen Y, Sadeh A (2007) A new approach for constructing and generating AOA networks. *J. Engrg. Comput. Architecture* 1(1):1–15.

Dawson RJ, Dawson CW (1998) Practical proposals for managing uncertainty and risk in project planning. *Internat. J. Project Management* 16(5):299–310.

Demeulemeester E, Vanhoucke M, Herroelen W (2003) A random network generator for activity-on-the-node networks. *J. Scheduling* 6(1):13–34.

El-Ghaoui L, Oustry F, Lebret H (1998) Robust solutions to uncertain semidefinite programs. *SIAM J. Optim.* 9(1):33–52.

Ellsberg D (1961) Risk, ambiguity, and the savage axioms. *Quart. J. Econom.* 75(4):643–669.

Fleming Q, Koppelman J (2005) *Earned Value Project Management*, 3rd ed. (Project Management Institute, Newtown Square, PA).

Fulkerson D (1961) A network flow computation for project cost curves. *Management Sci.* 7(2):167–178.

Goh J, Sim M (2010) Distributionally robust optimization and its tractable approximations. *Oper. Res.* 58(4, part 1):902–917.

Goh J, Sim M (2011) Robust optimization made easy with ROME. *Oper. Res.* 59(4):973–985.

Golenko-Ginzburg D, Gonik A (1997) Stochastic network project scheduling with non-consumable limited resources. *Internat. J. Production Econom.* 48(1):29–37.

Golenko-Ginzburg D, Gonik A (1998) High performance heuristic algorithm for controlling stochastic network projects. *Internat. J. Production Econom.* 54(3):235–245.

Gutjahr WJ, Strauss C, Wagner E (2000) A stochastic branch-and-bound approach to activity crashing in project management. *INFORMS J. Comput.* 12(2):125–135.

Hall NG, Posner ME (2001) Generating experimental data for computational testing with machine scheduling applications. *Oper. Res.* 49(6):854–865.

Hendrickson C, Au T (2000) *Project Management for Construction*, 2nd ed. (Carnegie Mellon University, Pittsburgh).

Herroelen W, Leus R (2005) Project scheduling under uncertainty: Survey and research potentials. *Eur. J. Oper. Res.* 165(2):289–306.

Hughes MW (1986) Why projects fail: The effects of ignoring the obvious. *Indust. Engrg.* 18(1):14–18.

Kapur KC (1973) An algorithm for project cost-duration analysis problems with quadratic and convex cost functions. *AIIE Trans.* 5(4):314–322.

Kelley JE Jr, Walker MR (1959) Critical-path planning and scheduling. *Proc. Eastern Joint Computer Conf., December 1–3, Boston*, 160–173.

Kerzner H (2009) *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 10th ed. (Wiley, Hoboken, NJ).

Klastorin TD (2004) *Project Management: Tools and Trade-Offs*, 1st ed. (Wiley, Hoboken, NJ).

Kwak YH, Ingall L (2007) Exploring Monte Carlo simulation applications for project management. *Risk Management* 9(1):44–57.

Leach LP (1999) Critical chain project management improves project performance. *Project Management J.* 30(2):39–51.

McCullagh P, Nelder JA (1989) *Generalized Linear Models*, 2nd ed. (Chapman & Hall, London).

Might RJ, Fischer WA (1985) The role of structural factors in determining project management success. *IEEE Trans. Engrg. Management* 32(2):71–77.

Mitchell G, Klastorin T (2007) An effective methodology for the stochastic project compression problem. *IIE Trans.* 39(10): 957–969.

Nesterov Y, Nemirovski A (1994) *Interior-Point Polynomial Algorithms in Convex Programming*, 1st ed. (SIAM, Philadelphia).

Pells DL (2008) Seven good reasons for the rapid growth of project management in IT and why that trend will continue. *PM World Today* 10(6):1–14.

Pender S (2001) Managing incomplete knowledge: Why risk management is not sufficient. *Internat. J. Project Management* 19(2):79–87.

Perry C, Greig ID (1975) Estimating the mean and variance of subjective distributions in PERT and decision analysis. *Management Sci.* 21(12):1477–1480.

Philips P (2005) Lessons for post-Katrina reconstruction: A high-road vs. low-road recovery. Briefing Paper 166, Economic Policy Institute, Washington, DC.

Project Management Institute (2010) Last accessed December 5, 2012, http://www.pmi.org/.

Regnier E (2005) Hidden assumptions in project management tools. Newsletter, Issue 11, Defense Resources Management Institute, Naval Postgraduate School, Monterey, CA.

Rockafellar RT, Uryasev S (2000) Optimization of conditional value-at-risk. *J. Risk* 2(3):21–41.

Schonberger RJ (1981) Why projects are "always" late: A rationale based on manual simulation of a PERT/CPM network. *Interfaces* 11(5):66–70.

Sion M (1958) On general minimax theorems. *Pacific J. Math.* 8(1):171–176.

Soyster AL (1973) Convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* 21(5):1154–1157.

Standish Group, The (2006) Chaos report. The Standish Group, Boston.

Tavares LV (2002) A review of the contribution of operational research to project management. *Eur. J. Oper. Res.* 136(1):1–18.

U.S. Department of Defense (1962) DOD and NASA guide: PERT cost systems design. U.S. Department of Defense, Washington, DC.

U.S. Navy (1958) PERT summary report, phase I. Defense Technical Information Center, AD-735 902, Special Projects Office, Bureau of Naval Weapons, Washington, DC.

Value Based Management.net (2009) Product life cycle—Industry maturity stages. Last accessed December 5, 2012, http://www.valuebasedmanagement.net/methods_product_life_cycle.html.

van Dorp JR, Duffey MR (1999) Modeling statistical dependence in risk analysis for project networks. *Internat. J. Production Econom.* 58(1):17–29.

White D, Fortune J (2002) Current practice in project management—An empirical study. *Internat. J. Project Management* 20(1):1–11.

Wiesemann W, Kuhn D, Rustem B (2012a) Multi-resource allocation in stochastic project scheduling. *Ann. Oper. Res.* 193(1):193–220.

Wiesemann W, Kuhn D, Rustem B (2012b) Robust resource allocations in temporal networks. *Math. Programming* 135(1–2): 437–471.

Williams T (2003) The contribution of mathematical modeling to the practice of project management. *IMA J. Management Math.* 14(1):3–30.

Zammori FA, Braglia M, Frosolini M (2009) A fuzzy multi-criteria approach for critical path definition. *Internat. J. Project Management* 27(3):278–291.

1000ventures.com (2009) Change management. Last accessed December 5, 2012, http://1000ventures.com/business_guide/crosscuttings/change_management.html.