Project Scheduling for Maximum NPV with Variable Activity Durations and Uncertain Activity Outcomes

Stefan Creemers
Roel Leus
Bert De Reyck
Marc Lambrecht

Abstract - **The literature on project scheduling with uncertain activity durations is still in its burn-in phase. We examine project scheduling with net-present-value objective and exponential activity durations by means of a backward stochastic dynamic programming recursion. We examine the particular setting in which the individual activities carry a risk of failure, and where an activity's failure results in the project's overall failure. In the project planning and scheduling literature, this technological uncertainty has typically been ignored and project plans are developed only for scenarios in which the project succeeds.**

Keywords - **project scheduling, net present value, stochastic activity durations, exponential distribution**

1 Introduction

Project management is a management discipline that is receiving a continuously growing amount of attention; [17, 20] are comprehensive references. Both in production and in service sectors, ever more organizations adhere to project-based organization and work, within a very wide variety of applications: research and development (R&D), software development, construction, public infrastructure, process re-engineering, maintenance operations, . . . A project itself can be informally defined as a unique undertaking, consisting of a complex set of precedence-related activities that have to be executed using diverse and mostly limited company resources. Project management deals with the selection and initiation of projects, as well as with their operation and control. *Project scheduling*, as a part of project management, is aimed at (a) deciding when in time to start (and finish) which activities and (b) the allocation of scarce resources to the project activities. In this paper, our focus will lie on function (a) of project scheduling.

During project execution, project activities are often subject to considerable uncertainty, which results from many different possible sources: activities may take more or less time than originally estimated, resources may become unavailable, material may arrive behind schedule, workers may be absent, etc. In this text we examine the case where this uncertainty is important enough to be incorporated into the planning phase. The sources of variability in processing times are manifold; nevertheless, the main scheduling objectives are mostly functions of the activity starting- (or ending-)times, which justifies a restriction to the study of uncertainty in processing times only, although many different sources may be at the basis of this variability.

In the absence of resource constraints, the minimum-makespan objective requires no real scheduling effort: all activities are started as soon as their predecessors are completed. The literature on this so-called PERT problem is usually concerned with the computation of certain characteristics of the minimum project makespan (earliest project completion) when the activity durations are random variables, mainly with exact computation, approximation and bounding of the distribution function and the expected value [1, 11, 18, 19].

Most firms undertake projects with the goal of making a profit, and project profitability is often measured by its net present value (NPV), the discounted value of the project's cash flows [5]. However, NPV is affected by the project schedule and in capital-intensive industries, the timing of expenditures has a major impact on project feasibility and profitability. Scheduling projects to maximize NPV in a deterministic setting has been studied under a broad range of contractual arrangements and planning constraints (see [14] for a review), but often in practice there is significant uncertainty, which was the motivation for Tilson et al. [22] to investigate project scheduling with stochastic activity durations to maximize expected NPV; a similar problem is studied by Benati [4]. The paper by Tilson et al. is most relevant to our setting since it describes how to find the best from a finite set of scheduling policies (for definitions see infra), while Benati only proposes a heuristic scheduling rule. Buss and Rosenblatt [7] also maximize a project's expected NPV and additionally consider activity delays.

Time/resource trade-offs with stochastic activity durations, in which the resource allocation influences the mean and/or the variance of the durations, are investigated in [3, 6, 12, 13, 23], amongst others. Kulkarni and Adlakha [18] also suggest an extension of their model towards such time/resource trade-offs, which was later examined in more detail by Azaron et al. [3]. In the current paper we will not be concerned with resource allocation and assume that such decisions have already been made at a higher hierarchical decision level. On the other hand, we will incorporate the concept of *activity success* or *failure*. For this text, the most relevant source of literature for this concept stems from the discipline of chemical engineering, most notably the work by Grossmann and his colleagues [21, 16], who initiated the work on scheduling failure-prone NPD (new product development) testing tasks when also non-sequential testing is admitted. They point out that in many industries, including the chemical and pharmaceutical sectors, a number of the tasks involved in producing a new product are regulatory requirements such as environmental and safety tests. The failure of a single required test may prevent a potential product from reaching the marketplace. This positions our work especially within the context of R&D: an important feature of R&D projects is that, apart from the commercial and market risks common to all projects, their constituent activities also carry the risk of technical failure. Therefore, besides projects overrunning their budgets or deadlines and the commercial returns not meeting their targets, such projects also carry the risk of failing altogether, resulting in time and resources spent without any tangible return; failure in one more activities leads to overall project termination. Our models are of particular interest to drug-development projects in the pharmaceutical industry, in which stringent scientific procedures have to be followed to ensure patient safety in distinct stages before a medicine can be approved for production. The project may need to be terminated in any of these stages, either because the product is revealed not to have the desired properties or because of harmful side effects.

In this text we examine the case where all activity cash flows during the development phase are negative, which is typical for R&D projects. The following insight is key when activity failures occur: if activity A ends no later than the start of activity B, then knowledge of the outcome (success or failure) of A can be used to reduce the expected cash outlay for activity B, since a failure in A would allow to abandon the project without investing in B. For a given selection of such 'information flows' (under the form of additional precedence constraints), a late-start schedule is then optimal when the activity durations are known. Unfortunately, such late-start scheduling is difficult to implement in case of stochastic durations, and Tilson et al. [22] implicitly restrict their attention to scheduling policies that start activities only at the end of other activities. Buss and Rosenblatt [7] partially remedy this problem by starting an activity only after a fixed time interval (delay), but do not decide which sets of activities to start at what time (all eligible activities are started as soon as possible after their delay time).

The goal of this paper is to investigate the development of scheduling policies for the described setting of stochastic durations and activity failures. The work in this paper is an extension of De Reyck and Leus [9], who examine uncertain activity outcomes but only allow for deterministic durations, and also of Creemers et al. [8], who concentrate only on duration variability.

2 Definitions

A project consists of a set of activities (or tasks) $N = \{0, 1, \ldots, n\}$, which are to be processed without interruption. The duration D_i of each activity i is a random variable (r.v.); the vector $(D_0, D_1, \ldots, D_n)$ is denoted by $\mathbf{D}$. The set A is a strict order on N, i.e. an irreflexive and transitive relation, which represents technological precedence constraints. The activities 0 and n represent start and end of the project, respectively, and are the (unique) least and greatest element of the partially ordered set (N, A). We use lowercase vector $\mathbf{d} = (d_0, d_1, \ldots, d_n)$ to represent one particular realization (or sample, or scenario) of $\mathbf{D}$. For a given realization $\mathbf{d}$, we can produce a schedule $\mathbf{s}$, i.e., a vector of starting times $(s_0, s_1, \ldots, s_n)$ with $s_i \geq 0$ for all $i \in N$. The schedule $\mathbf{s}$ is *feasible* if $s_i + d_i \leq s_j$ for all $(i, j) \in A$. In the remainder of this paper, we assume the durations of the activities $i \in N \setminus \{0, n\}$ to be mutually independent exponentially distributed r.v.s with mean $\frac{1}{\mu_i}, \mu_i > 0$; $D_0 = D_n = 0$. In the literature, a PERT network with independent and exponentially distributed activity durations is often referred to as a *Markovian PERT network*.

We focus on project scheduling with NPV (net present value) objective. A cash flow c_i is associated with each activity $i \in N$, which is a rational number that is assumed to be negative in the context of this paper. Each cash outflow is incurred at the start of the activity. Each activity $i \in N \setminus \{n\}$ also has a probability of technical success (PTS) p_i; we assume that $p_0 = 1$ and consider the outcomes of the different tasks to be independent. Information on activity success and duration becomes available only at the end of the activity. Overall project success generates an end-of-project payoff $C > 0$, which is received at the start of activity n. This final project payoff is only achieved when all activities are successful, consequently the probability π of project success equals $\prod_{i \in N \setminus \{n\}} p_i$. In order to account for the time value of money, we define r to be the applicable continuous discount rate: the present value of a cash flow c incurred at time t equals ce^{-rt}. Without loss of generality, we assume that C is large enough for the project to be undertaken (otherwise all starting times may be set to infinity).

For Markovian PERT networks, Kulkarni and Adlakha [18] describe an exact method for deriving the distribution and moments of the earliest project completion time using continuous-time Markov chains (CTMCs). Both Tilson et al. [22] and Buss and Rosenblatt [7] focus on the NPV objective and use the CTMC described in [18] as a starting point for their algorithm. We achieve a significant performance improvement compared to these existing approaches, based on a judicious partitioning of the state space, and also incorporate the concept of activity failures.

3 Problem statement

The execution of a project with stochastic durations can best be seen as a dynamic decision process. A solution is a *policy* Π, which defines *actions* at *decision times*. Decision times are typically $t = 0$ (the start of the project) and the completion times of activities; a tentative next decision time can also be specified by the decision maker. An action can entail the start of a set of activities that is 'feasible', so that a feasible schedule is constructed gradually through time. Next to the input data of the problem instance, a decision at time t may only use information (on activity-duration realizations) that has become available before or at time t; this requirement is often referred to as the *non-anticipativity constraint*.

As soon as all activities are completed, the activity durations are known, yielding a realization $\mathbf{d}$ of $\mathbf{D}$. Consequently, every policy Π may alternatively be interpreted [15] as a function $\mathbb{R}^{n+1}_{\geq} \mapsto \mathbb{R}^{n+1}_{\geq}$ that maps given samples $\mathbf{d}$ of activity durations to vectors $\mathbf{s}(\mathbf{d}; \Pi)$ of feasible activity starting times (schedules). For a given scenario $\mathbf{d}$ and policy Π, $s_n(\mathbf{d}; \Pi)$ denotes the makespan of the schedule. The earlier-mentioned PERT problem aims at characterizing the r.v. $s_n(\mathbf{D}; \Pi^{ES})$, where policy Π^{ES} starts all activities as early as

task i	cash flow c_i	mean duration $E[D_i]$	PTS p_i
0	0	0	100%
1	-3	1	75%
2	-1	2	100%
3	-2	3	80%
4	-12	4	100%
5	-17	3	75%

Table 1: Project data for the example project.

possible. Contrary to e.g. the expected makespan, however, NPV is a non-regular measure of performance: starting activities as early as possible is not necessarily optimal.

In this text we investigate the determination of an optimal scheduling policy for the expected-NPV objective. In the special case where the durations have constant values $\mathbf{d}$, the objective function corresponding with a schedule $\mathbf{s}$ is the following:

$$\max \qquad g(\mathbf{s}, \mathbf{d}) = \pi C e^{-rs_n} + \sum_{i=1}^{n-1} q_i(\mathbf{s}) c_i e^{-rs_i} \tag{1}$$

In this expression, $q_i(\mathbf{s})$ represents the probability that cash flow c_i will be incurred (so that activity i is paid for), for $i \in N \setminus \{n\}$. It can be seen that $q_i(\mathbf{s}) = \prod_{\substack{k\in N:\\ s_k+d_k\leq s_i}} p_k$, the probability that all activities ending no later than s_i succeed.

Our goal in this article is to select a policy Π^* within a specific class that minimizes $E[g(\mathbf{s}(\mathbf{D};\Pi), \mathbf{D})]$, with $E[\cdot]$ the expectation operator with respect to $\mathbf{D}$. The generality of this problem statement suggests that optimization over the class of all policies will probably turn out to be computationally intractable. We therefore restrict our optimization to a subclass that has a simple combinatorial representation and where decision points are limited in number: our solution space consists of all policies that start activities only at the end of other activities (activity 0 is started at time 0).

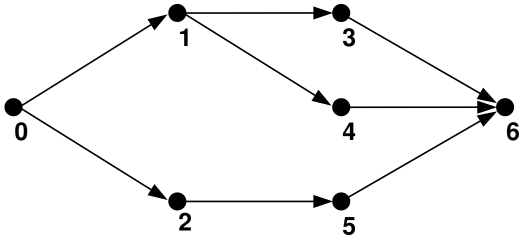

Figure 1: The precedence graph $G(N, t(A))$ for the example project. The name of each node is represented below the node.

4 Illustration with known durations

We first illustrate the setting of activity success or failure by means of a project with deterministic activity durations. We consider an example project with $n = 6$, for which the input order A is described by Figure 1, which shows the directed acyclic graph $G(N, t(A))$, where $t(A)$ is the transitive reduction[1] of A (which is unique [2]). Further input data are provided in Table 1; the project's payoff value C is 80 and the discount rate $r = 1\%$ per time unit. When all activity durations equal their expectation, the schedule $\mathbf{s}_1$ shown in Figure 2 (with $s_0 = 0$ and $s_6 = 9$) is feasible and has an objective-function value equal to

$$\begin{aligned} g(\mathbf{s}_1, E[\mathbf{D}]) = \quad & c_1 e^{-r} + c_2 + p_1 p_2 c_3 e^{-2r} + p_1 p_2 p_3 e^{-5r}(c_4 + c_5) \\ & + \pi C e^{-9r} \\ = \quad & -3e^{0.01} - 1 - 1.5e^{-0.02} - 0.6e^{-0.05} \times 29 \\ & + 36e^{-0.09} \\ = \quad & 10.90968. \end{aligned}$$

An optimal schedule for the same example is $\mathbf{s}_2$ as shown in Figure 3, with objective function $g(\mathbf{s}_2, E[\mathbf{D}]) = 12.194826$.

[1] The transitive reduction of a binary relation E on a set N is the minimal relation E' on N such that $T(E') = T(E)$; the transitive closure $T(E)$ of a binary relation E on a set N is the minimal transitive relation E' on N that contains E.

5 Stochastic durations

At any time instant t, each activity's *status* is either *idle* (= unstarted), *active* (= in the process of being executed) or *finished*; we write $\Omega_i(t) = 0$, 1 or 2, respectively, for $i \in N$. The state of the system is defined by the status of the individual activities and is represented by vector $\Omega(t) = (\Omega_0(t), \Omega_1(t), \ldots, \Omega_n(t))$. State transitions take place each time an activity finishes and are determined by the policy at hand. The project's starting and finishing conditions are $\forall i \in N : \Omega_i(0) = 0$ and $\forall i \in N : \Omega_i(t) = 2$, $\forall t \geq \omega$, respectively, where ω indicates the project completion time. The problem of finding an optimal scheduling policy corresponds to optimizing a discounted criterion in a continuous-time Markov decision chain (CTMDC) on the state space Q, with Q containing all the states of the system that can be visited by the transitions (which are called *feasible* states); the decision set is described below. We apply a backward SDP recursion to determine optimal decisions, in line with [22] and [8] and based on the CTMC as described in [18].

For ease of description, we adopt a different characterization of each state $\mathbf{v} = (v_0, v_1, \ldots, v_n) \in Q$: we let $I(\mathbf{v})$, $X(\mathbf{v})$ and $L(\mathbf{v})$ represent the activities in N that are idle, active and finished, respectively. There is a one-to-one correspondence between (not necessarily feasible) state vectors $\mathbf{v}$ and mutually exclusive sets I, X and L with $I \cup X \cup L = N$. The key instrument of the SDP recursion is the *value function* $G(\cdot)$, which determines the expected NPV of each feasible state at the time of entry of the state, conditional on the hypothesis that optimal decisions are made in all subsequent states. In the definition of the value function $G(I, X)$, we supply sets I and X of idle and active activities (which uniquely determines the finished activities).

We call an activity i *eligible* at time t if $\Omega_i(t) = 0$ and $\forall (j, i) \in A : \Omega_j(t) = 2$; let $E(\mathbf{v}) = \{i \in N | v_i = 0 \land \forall (j, i) \in A : v_j = 2\}$. At the

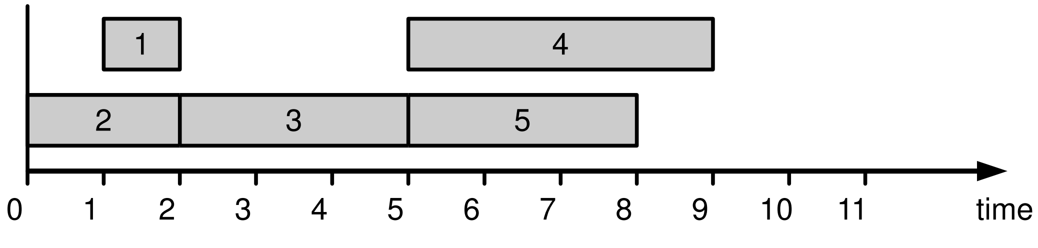

Figure 2: A feasible schedule for the example project with mean durations.

entry of a state $\mathbf{v} \in Q$, a decision needs to be made whether to start a set of eligible activities (and if so, which), or not to start any activities; the latter decision is possible only if $X(\mathbf{v}) \neq \varnothing$. If no activities are started, a transition towards another state takes place after the first completion of an element of $X(\mathbf{v})$. The probability that activity $i \in X(\mathbf{v})$ finishes first among the active activities equals $\mu_i / \sum_{k\in X(\mathbf{v})} \mu_k$. The expected time to the first completion is $\left(\sum_{i\in X(\mathbf{v})} \mu_i\right)^{-1}$ time units (the length of this timespan is also exponentially distributed). The appropriate discount factor to be applied for this timespan is $\sum_{k\in X(\mathbf{v})} \mu_k / \left(r + \sum_{k\in X(\mathbf{v})} \mu_k\right)$. The expected discounted NPV to be obtained from the next state on condition that no new activities are started, therefore equals

$$\frac{\sum_{k\in X(\mathbf{v})} \mu_k}{r + \sum_{k\in X(\mathbf{v})} \mu_k} \sum_{i\in X(\mathbf{v})} \frac{p_i \mu_i}{\sum_{k\in X(\mathbf{v})} \mu_k} G(I(\mathbf{v}), X(\mathbf{v}) \setminus \{i\}). \tag{2}$$

The second alternative is to start a non-empty set of eligible activities $S \subseteq E(\mathbf{v})$ when state $\mathbf{v} \in Q$ is entered. This leads to incurring a cost $\sum_{i\in S} c_i$ and an immediate transition to another state, with no discounting required. The corresponding immediate NPV (in expectation), conditional on set $S \neq \varnothing$ being started, is

$$G(I(\mathbf{v}) \setminus S, X(\mathbf{v}) \cup S) + \sum_{i\in S} c_i. \tag{3}$$

The total number of decisions S that can be made is $2^{|E(\mathbf{v})|}$ if we include $S = \varnothing$ according to Equation (2), otherwise it is one less. The decision corresponding with the highest value in (2) and (3) determines $G(I(\mathbf{v}), X(\mathbf{v}))$, which completes our description of the SDP recursion. The optimal objective-function value is $\max_{\Pi} E[g(\mathbf{s}(\mathbf{D}; \Pi), \mathbf{D})] = G(N, \varnothing)$.

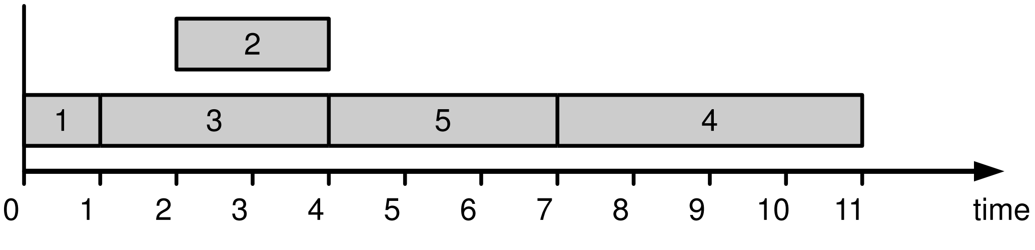

Figure 3: An optimal schedule for the example project with mean durations.

An upper bound on $|Q|$ is 3^n. Enumerating all these 3^n states is not recommendable, because typically the majority of the states do not satisfy the precedence constraints. Tilson et al. [22] develop a simple yet efficient algorithm to produce a set of possible states; this set contains Q but may be strictly larger. Additionally, to the best of our knowledge, all related studies in the published literature reserve memory space to store the entire state space of the CTMDC; Buss and Rosenblatt [7] point out that some method of decomposition to reduce these memory requirements would allow for considerable efficiency enhancements. We have implemented the techniques proposed in the recent working paper [8] to considerably improve upon the storage and computational requirements of earlier algorithms by means of efficient creation of Q and decomposition of the network of state transitions.

The SDP recursion starts in state $(2, 2, \ldots, 2, 0)$, so we omit states $(2, 2, \ldots, 1)$ and $(2, 2, \ldots, 2)$. Subsequently, the value function is computed stepwise for all other states. As the algorithm progresses, the information on the earlier generated states will no longer be required for further computation and therefore the memory occupied can be freed; this procedure is based on a partition of Q, where the necessary subsets are generated and deleted when appropriate.

6 Illustration (continued)

Reverting to the example presented in Section 4, we provide an illustration of a project network with stochastic durations. All main characteristics of the example project are preserved with the exception of the activity durations, which are assumed to be exponentially distributed with means $E[D_i]$.

At the onset of the project (i.e. at state $(2, 0, 0, 0, 0, 0, 0)$) we can decide to start the first activity, the second activity or to start both. The SDP recursion evaluates the expected outcomes of each of these decisions and selects one that yields the highest expected NPV (assuming optimal decisions are made at all future decision moments). In our example, it is optimal to start only the first activity (resulting in an objective function $\max_\Pi E[g(\mathbf{s}(\mathbf{D}; \Pi), \mathbf{D})] = 12.093676$) and we subsequently end up in state $(2, 1, 0, 0, 0, 0, 0)$. The only possible transition is then towards the state in which the first activity is finished ($(2, 2, 0, 0, 0, 0, 0)$). At this point, three activities become eligible and so 2^3 minus 1 (i.e., the decision not to start any activities) decisions

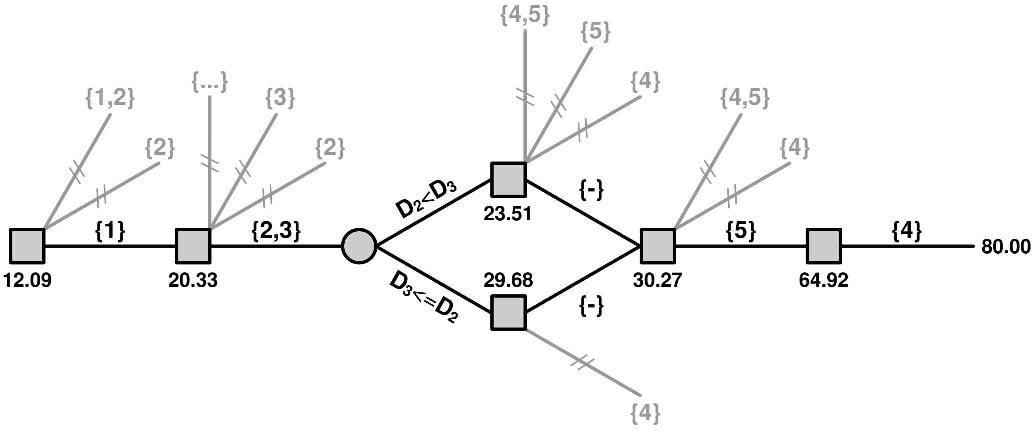

Figure 4: Optimal paths in the decision tree for the example project with stochastic durations.

are to be evaluated. The optimal decision is to start activities 2 and 3 (the resulting state is $(2,2,1,1,0,0,0)$). With two activities in progress, two possibilities arise: either activity 2 or activity 3 has to finish first. In both cases, however, it is optimal to wait for the other activity to finish as well prior to starting any additional activities. As such, we end up in state $(2,2,2,2,0,0,0)$ with eligible activities 4 and 5, from where the optimal decision is to start only activity 5. After the completion of activity 5, the only decision left is to start activity 4, after whose completion we end up in the final state $(2,2,2,2,2,2,0)$ and collect the project payoff. The relevant part of the corresponding decision tree is illustrated in Figure 4, in which the project evolves from left to right. A 'decision node', represented by a square □, indicates that a decision needs to be made at that point in the process; a 'chance node' (also called 'event node'), denoted by a circle ◯, indicates that a random event occurs at that point. At each of the decision nodes, we indicate the expected NPV at the decision moment. For each decision node, a double dash // is added to each branch that does not correspond with a best choice in the associated SDP recursion.

We observe a clear similarity between the case with deterministic and the one with stochastic durations. Not only are the objective function values very close to each other, but we also see that the optimal scheduling policy for the stochastic case will, the late-start character of activity 2 in Figure 3 left aside, re-produce the optimal schedule for the deterministic case when all stochastic variables take on their mean values. This comes as no surprise when we take into count the limited size of the example project and the fact that only few activities can be executed in parallel (limiting the number of decisions to be made). Results reported in Buss and Rosenblatt [7], on the other hand, indicate that, for some scheduling instances, significant differences in objective function values may be observed between the case with stochastic and with deterministic durations. This will certainly hold for larger and denser networks.

7 Performance

The SDP recursion developed in this article is an extension of the recursion presented in Creemers et al. [8] for scheduling Markovian PERT networks with maximum-NPV objective, which seems to be the current state of the art. We have used RanGen [10] to create project network instances of varying size (n) and order strength (OS), the latter being a measure for the density of the network: OS is the proportion of precedence-related activity pairs compared to the theoretically maximum number of such pairs. Preliminary computational results indicate that for $OS = 0.4$, we are able to analyze networks of up to 50 activities. With respect to $OS = 0.6$ and $OS = 0.8$, instances with up to respectively 80 and 120 activities are successfully solved.

References

[1] V.G. Adlakha and V.G. Kulkarni. A classified bibliography of research on stochastic PERT networks: 1966-1987. *INFOR*, 27:272–296, 1989.

[2] A.V. Aho, M.R. Garey, and J.D. Ullman. The transitive reduction of a directed graph. *SIAM Journal on Computing*, 1:131–137, 1972.

[3] A. Azaron, H. Katagiri, M. Sakawa, K. Kato, and A. Memariani. A multi-objective resource allocation problem in PERT networks. *European Journal of Operational Research*, 172:838–854, 2006.

[4] S. Benati. An optimization model for stochastic project networks with cash flows. *Computational Management Science*, 3:271–284, 2006.

[5] R.A. Brealey and S.C. Myers. *Principles of Corporate Finance.* McGraw-Hill, 2000.

[6] J.M. Burt. Planning and dynamic control of projects under uncertainty. *Management Science*, 24:249–258, 1977.

[7] A.H. Buss and M.J. Rosenblatt. Activity delay in stochastic project networks. *Operations Research*, 45:126–139, 1997.

[8] S. Creemers, R. Leus, and M. Lambrechts. Scheduling Markovian PERT networks with maximum-NPV objective. Technical Report 0811, Department of Decision Sciences and Information Management, FBE, KULeuven, Leuven, Belgium, 2008.

[9] B. De Reyck and R. Leus. R&D-project scheduling when activities may fail. *IIE Transactions*, 40:367–384, 2008.

[10] E. Demeulemeester, M. Vanhoucke, and W. Herroelen. A random network generator for activity-on-the-node networks. *Journal of Scheduling*, 6:13–34, 2003.

[11] S.E. Elmaghraby. *Activity Networks: Project Planning and Control by Network Models.* Wiley, 1977.

[12] Y. Gerchak. On the allocation of uncertainty-reduction effort to minimize total variability. *IIE Transactions*, 32:403–407, 2000.

[13] W.J. Gutjahr, C. Strauss, and E. Wagner. A stochastic branch-and-bound approach to activity crashing in project management. *INFORMS Journal on Computing*, 12:125–135, 2000.

[14] W. Herroelen, P. Van Dommelen, and E. Demeulemeester. Project network models with discounted cash flows: a guided tour through recent developments. *European Journal of Operational Research*, 100:97–121, 1997.

[15] G. Igelmund and F.J. Radermacher. Preselective strategies for the optimization of stochastic project networks under resource constraints. *Networks*, 13:1–28, 1983.

[16] V. Jain and I.E. Grossmann. Resource-constrained scheduling of tests in new product development. *Industrial and Engineering Chemistry Research*, 38:3013–3026, 19999.

[17] H. Kerzner. *Project Management. A Systems Approach to Planning, Scheduling and Controlling.* Wiley, 1998.

[18] V.G. Kulkarni and V.G. Adlakha. Markov and Markov-regenerative PERT networks. *Operations Research*, 34:769–781, 1986.

[19] A. Ludwig, R.H. Möhring, and F. Stork. A computational study on bounding the makespan distribution in stochastic project networks. *Annals of Operations Research*, 102:49–64, 2001.

[20] J.R. Meredith and S.J. Jr. Mantel. *Project Management. A Managerial Approach.* Wiley & Sons, 2000.

[21] C.W. Schmidt and I.E. Grossmann. Optimization models for the scheduling of testing tasks in new product development. *Industrial and Engineering Chemistry Research*, 35:3498–3510, 1996.

[22] V. Tilson, M.J. Sobel, and J.G. Szmerekovsky. Scheduling projects with stochastic activity duration to maximize EPV. *SSRN eLibrary*, 2006. Working Paper Series; available at http://ssrn.com/paper=1015785.

[23] R.D. Wollmer. Critical path planning under uncertainty. *Mathematical Programming Study*, 25:164–171, 1985.