Contingent Portfolio Programming for the Management of Risky Projects

Janne Gustafsson • Ahti Salo
Systems Analysis Laboratory, Helsinki University of Technology,
Otakaari 1M, P.O. Box 1100, 02015 HUT, Finland
janne.gustafsson@hut.fi, ahti.salo@hut.fi

Abstract: Methods for selecting a research and development (R&D) project portfolio have attracted considerable interest among practitioners and academics. This notwithstanding, the industrial uptake of these methods has remained limited, partly due to the difficulties of capturing relevant concerns in R&D portfolio management. Motivated by these difficulties, we develop Contingent Portfolio Programming (CPP) which extends earlier approaches in that it (i) uses states of nature to capture exogenous uncertainties, (ii) models resources through dynamic state variables, and (iii) provides guidance for the selection of an optimal project portfolio that is compatible with the decision maker's risk attitude. Although CPP is presented here in the context of R&D project portfolios, it is applicable to a variety of investment problems where the dynamics and interactions of investment opportunities must be accounted for.

Keywords: *Research and development, project selection*; *Decision analysis, theory*; *Programming, linear, applications*

1 Introduction

The selection of research and development (R&D) projects has attracted considerable interest in the literatures on technology management and operations research (OR). These projects involve many characteristics – such as uncertainties and interdependent resource constraints – that are potentially amenable to analysis by OR techniques. Indeed, there exists a variety of related methods, ranging from *scoring methods* such as value trees (Keeney and Raiffa 1976, French 1986) to *optimization models* (see, e.g., Gear and Lockett 1973, Heidenberger 1996, Ghasemzadeh et al. 1999) and *dynamic programming methods* such as decision trees and real options (French 1986, Dixit and Pindyck 1994, Smith and Nau 1995, Trigeorgis 1996). Yet, despite the plethora of methodological approaches, these methods have not enjoyed widespread industrial use, possibly due to the difficulties of capturing the full range of phenomena that are relevant to the problem of selecting and managing R&D projects.

Building on the literatures on decision analysis, R&D management, and portfolio optimization, we develop *Contingent Portfolio Programming* (CPP) as a modeling framework which accommodates most of the characteristics that are relevant to the selection of risky projects. In CPP, projects are regarded as risky investment opportunities that consume and produce several resources over multiple time periods. The staged nature of R&D projects is captured through project-specific decision trees (cf. Gear and Lockett 1973) which support managerial flexibility by allowing the decision maker (DM) to take stepwise decisions on each project in view of most recent information (Trigeorgis 1996). Uncertainties, on the other hand, are modeled through a state tree in the spirit of stochastic programming (see, e.g., Birge and Louveaux 1997).

While CPP permits a wide range of risk attitudes, we focus on a class of objective functions that are a combination of a mean-risk model (Markowitz 1959, 1987) and a multi-attribute value function (Keeney and Raiffa 1976). In particular, we consider two objective functions which lead to linear programming models, permitting the solution of relatively large-scale project portfolios. The first one is a *mean-lower semi-absolute deviation model*, (mean-LSAD model; Ogryczak and Ruszsynski 1999) which is a special case of generalized disappointment models with a standard measure of risk (Jia and Dyer's 1996, Jia et al. 2001). The second is a *mean-expected downside risk model* (mean-EDR model; Eppen et al. 1989) which is consistent with expected utility theory (Fishburn 1977).

The rest of the paper is structured as follows. §2 provides a brief overview of earlier approaches and §3 presents an introductory example. A formal development of CPP is given in §4, followed by an analysis of computational complexity in §5. §6 concludes the paper with suggestions for future research directions.

2 Earlier Approaches

Several methods for the selection of R&D projects have been developed over the past few decades (for a review, see Martino 1995 and Henriksen and Traynor 1999). These methods can be categorized into three aggregate groups: (1) scoring models, (2) optimization models, and (3) dynamic programming models. Among these, the two latter groups are more relevant to CPP, although some ideas of scoring models (e.g., consideration of multiple attributes) are also included in CPP.

Optimization models for project selection can be viewed as extensions of standard capital budgeting models (see, e.g., Luenberger 1998). These models capture project interdependencies and resources constraints, but they do not usually address uncertainties associated with the projects' outcomes, which makes it impossible to attach risk measures to project portfolios. There are some approaches based on utility functions, fuzzy set theory, and chance-constraints, but the resulting models are problematic as they make restrictive assumptions about the nature of uncertainty or the DM's risk preferences.

Stochastic optimization models analogous to R&D portfolio selection models have appeared in investment planning as well as asset and liability management (e.g., Birge and Louveaux 1997, pp. 20–28, and Mulvey et al. 2000). These two problem contexts share similarities with the selection of R&D projects in that (i) the DM seeks to maximize the value of a portfolio of risky assets in a multi-periodic setting and (ii) there are several asset categories which parallel the multiple resource types consumed and produced by R&D

projects. A key difference, however, is that in financial optimization the (dis)investment decisions are unconstrained quantities that do not restrict the DM's future decision opportunities (e.g., security trading). In contrast, R&D project selection involves "go /no go"-style decisions where the "go"-decision leads to later project management decisions while the "no go"-decision terminates the project without offering further decision opportunities.

The staged nature of R&D projects has motivated the development of dynamic programming approaches, most notably (1) *decision trees* based on decision analysis (see French 1986) and (2) *real options* (see, e.g., Dixit and Pindyck 1994, Trigeorgis 1996); for a comparative analysis of these two approaches, we refer to Smith and Nau (1995). Dynamic programming methods capture the structure of consecutive decisions and uncertainties of an R&D project, but they do not explicitly address projects interactions or resource constraints. In consequence, researchers have sought to link decision trees with portfolio models (e.g., Heidenberger 1996 and Gear and Lockett 1973); this notwithstanding, the resulting models have failed to capture many relevant phenomena in R&D portfolio selection, such as risk aversion and resource dynamics.

The ability to yield theoretically defensible discount rates is often stated as a major advantage of the real options approach over decision trees (Trigeorgis 1996). However, this approach assumes that the cash flows from the project can be replicated with financial instruments for all states of nature, which may be unrealistic if the project results in innovative products that are not similar to market-traded assets. Furthermore, much of the real options literature employs continuous stochastic processes in the modeling of uncertainties, whereas the uncertainties of R&D projects often relate to discrete events. These features may make it difficult to use the real options approach in practical project selection problems.

3 Preliminary Example

In CPP, the DM makes decisions about which projects are started, when they are started, what resources are allocated to them, and in what situations they are terminated or expanded, among others. The decisions are subjected to relevant constraints (e.g., availability of resources), and they influence the resource flows that are acquired from the project portfolio.

For each project, the resource flows depend on the future states of nature. The resource flows associated with a given portfolio management strategy are consequently uncertain, which means that the final resource position at the end of the planning horizon is risky. It is assumed that the DM seeks to maximize the utility (or equivalently, certainty equivalent) of her final resource position.

For illustrative purposes, let us assume that the DM can invest in projects A and B in two phases (see Figure 1). In period zero, she can start either one or both of the projects. If a project is started, she can make an additional investment in period one, in which case the project generates a positive cash flow in period two; otherwise, the project is terminated in which case it yields no further cash flows. Any surplus that is not invested can be deposited at an 8% risk-free interest rate. The initial budget is $9 million.

Uncertainties are captured through a state tree with seven states, of which two are associated with period one and four with period two (see Figure 2). Taken together, Figures 1 and 2 correspond to the decision trees in Figures 3 and 4 where the projects' cash flows are shown as a function of the indexed action variables X ($X = 1$, if action is selected, X=0 if it is not). In Figures 5 and 6, these cash flows are shown as a function of action variables, while Figure 7 shows the cash flows from the entire portfolio. The two projects are negatively correlated so that that if project A performs poorly, project B yields a high return, and vice versa.

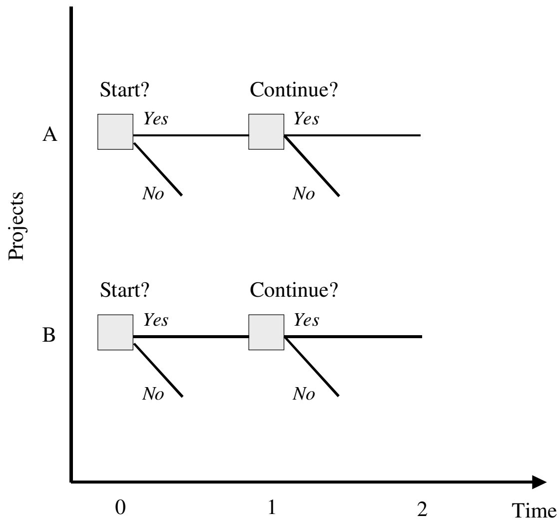

Figure 1 Decisions for projects *A* and *B*

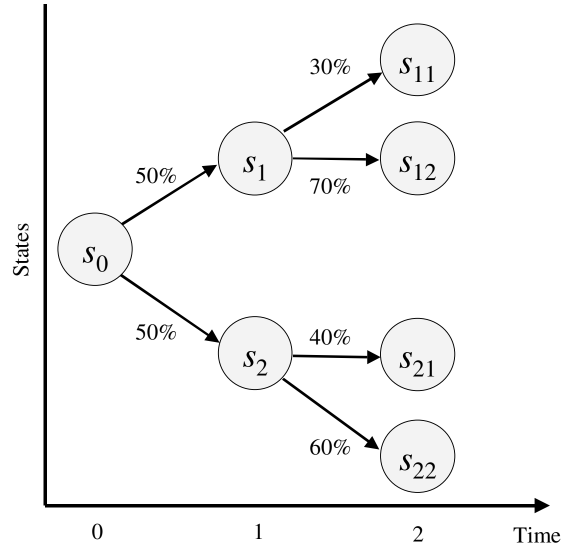

Figure 2 States of nature

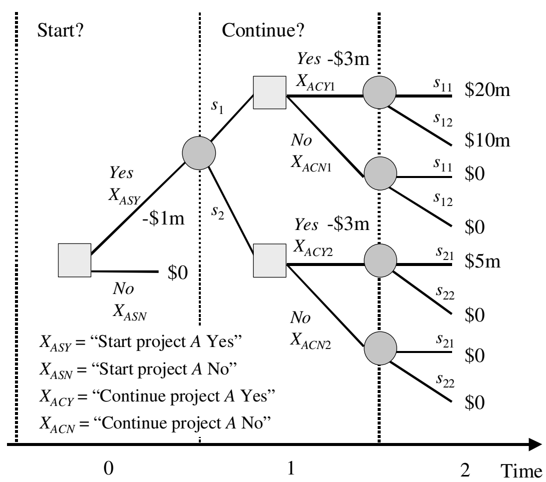

Figure 3 Decision tree of project *A*

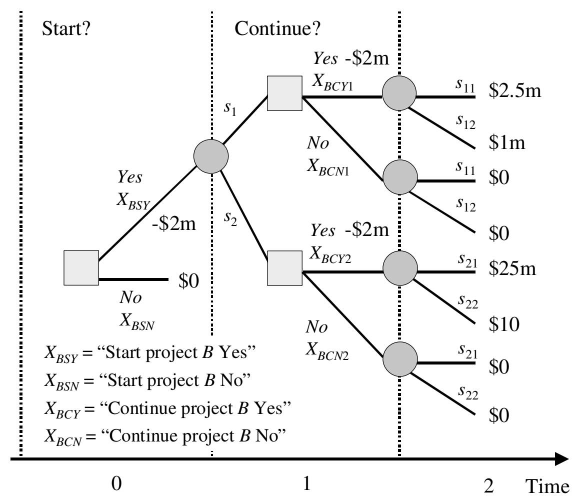

Figure 4 Decision tree of project *B*

Based on Figure 7, resource constraints can now be written as

$$-1 \cdot X_{ASY} - 2 \cdot X_{BSY} + 9 - RS_{s0} = 0$$

$$-3 \cdot X_{ACY1} - 2 \cdot X_{BCY1} + 1.08 \cdot RS_{s0} - RS_{s1} = 0$$

$$-3 \cdot X_{ACY2} - 2 \cdot X_{BCY2} + 1.08 \cdot RS_{s0} - RS_{s2} = 0$$

$$20 \cdot X_{ACY1} + 2.5 \cdot X_{BCY1} + 1.08 \cdot RS_{s1} - RS_{s11} = 0$$

$$10 \cdot X_{ACY1} + 1 \cdot X_{BCY1} + 1.08 \cdot RS_{s1} - RS_{s12} = 0$$

$$5 \cdot X_{ACY2} + 25 \cdot X_{BCY2} + 1.08 \cdot RS_{s2} - RS_{s21} = 0$$

$$10 \cdot X_{BCY2} + 1.08 \cdot RS_{s2} - RS_{s22} = 0,$$

where *RS*'s are nonnegative real-valued variables that denote the resource surplus in each state (i.e., resource position).

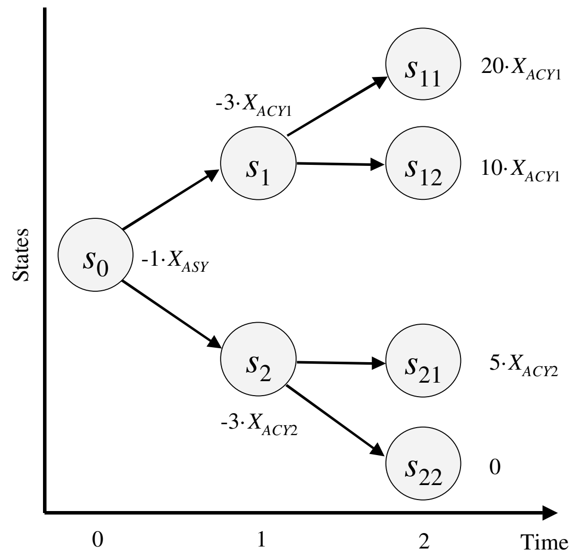

Figure 5 Cash flows of project *A*

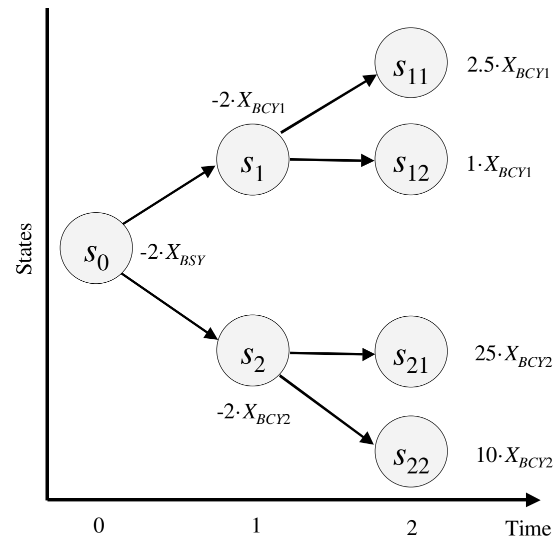

Figure 6 Cash flows of project *B*

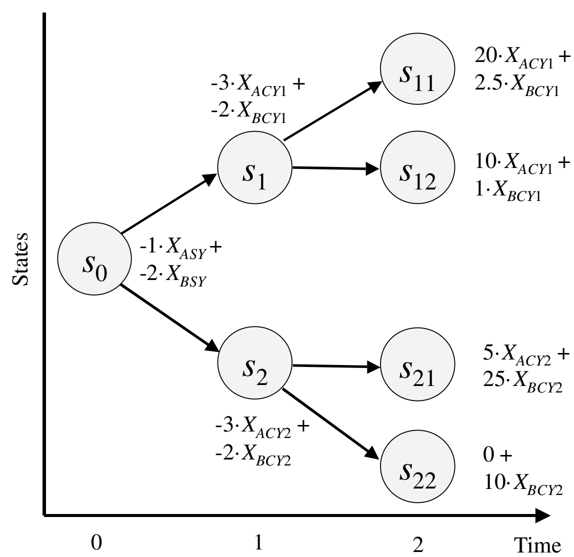

Figure 7 Cash flows of the project portfolio

For the measurement of risk, a second set of constraints is developed by introducing deviation variables ΔV_s^- (ΔV_s^+) which indicate by how much the value of the DM's resource position in a specific terminal state falls short of (or exceeds, respectively) the expected value of the resource position at the end of the planning horizon. These constraints are $V_s - EV - \Delta V_s^+ + \Delta V_s^- = 0$, where V_s denotes the value of the resource position in state *s*, EV is the expected value of the resource position over all terminal states, and the sum $-\Delta V_s^+ + \Delta V_s^-$ measures by how much V_s differs from EV. Since there are no other resource types except money, EV is given by

$$EV = 50\% \cdot 30\% \cdot RS_{s11} + 50\% \cdot 70\% \cdot RS_{s12} + 50\% \cdot 40\% \cdot RS_{s21} + 50\% \cdot 60\% \cdot RS_{s22},$$

while the deviation constraint for state s_{11} is $RS_{s11} - EV - \Delta V^{+}_{s11} + \Delta V^{-}_{s11} = 0$. The constraints for other terminal states can be expressed similarly. Because positive realizations of ΔV^{-}_{s} will be penalized in the objective function by a negative coefficient, only one of the terms ΔV^{+}_{s} and ΔV^{-}_{s} can be positive.

Because continued investments in period 1 are possible only if the project was initially started, the following consistency constraints apply (cf. Figures 3 and 4):

$$X_{ASY} + X_{ASN} = 1 \qquad X_{ACY1} + X_{ACN1} = X_{ASY} \qquad X_{ACY2} + X_{ACN2} = X_{ASY}$$
$$X_{BSY} + X_{BSN} = 1 \qquad X_{BCY1} + X_{BCN1} = X_{BSY} \qquad X_{BCY2} + X_{BCN2} = X_{BSY}$$

The action variables *X*'s are nonnegative integers; in fact, they are binary variables due to the two leftmost consistency constraints.

The DM seeks to maximize the certainty equivalent of her terminal resource position. It is assumed that this can be approximated in the mean-risk form by deducting a risk term based on lower semi-absolute deviation (LSAD) from the expected resource position in period 2 (see, e.g., Ogryczak and Ruszsynski 1999). For example, if the risk aversion coefficient for LSAD has been estimated at $\lambda = 0.5$, the objective function is

Maximize CE = *EV* – 0.5·LSAD =

$$50\% \cdot 30\% \cdot RS_{s11} + 50\% \cdot 70\% \cdot RS_{s12} + 50\% \cdot 40\% \cdot RS_{s21} + 50\% \cdot 60\% \cdot RS_{s22} - 0.5 \cdot$$
$$\left[50\% \cdot 30\% \cdot \Delta V^{-}_{s11} + 50\% \cdot 70\% \cdot \Delta V^{-}_{s12} + 50\% \cdot 40\% \cdot \Delta V^{-}_{s21} + 50\% \cdot 60\% \cdot \Delta V^{-}_{s22}\right].$$

With this objective function, the optimal strategy is to start both projects, but to terminate project *A* in period 1 if state s_2 occurs and project *B* if state s_1 occurs (i.e., decision variables X_{ASY}, X_{ACY1}, X_{ACN2}, X_{BSY}, X_{BCN1}, and X_{BCY2} are one while all other *X*'s are zero). In period 2, the corresponding expected resource position is *EV* = \$18.80 million and the LSAD term is \$2.95 million, which leads to the optimal CE value \$18.80 – 0.5 × \$ 2.95 = \$17.33 million. In period 2, the resource position attains its lowest level in state s_{12} at \$13.76 million, well above the $1.08^2 \times \$9 \approx \$10.50$ million obtained by depositing the initial budget at the risk-free interest rate. Thus, even though both projects entail the risk of losing most of the initial investment when evaluated in isolation, the optimal strategy yields a return that surely exceeds the risk-free interest rate.

Assuming that the same risk-free interest rate is applied to all states, the present value of the optimal project portfolio can be readily calculated. That is, by discounting the CE of the final resource position at the 8 % risk-free interest rate and by deducting the initial budget of $9 million from this value, the net present value of the portfolio is found to be $5.85 million. The risk-adjusted discount rate ρ that accounts for both time and risk preferences can be computed from $EV/(1+\rho)^2 = CE/(1+r_f)^2$, which gives $\rho = (1+r_f)\sqrt{EV/CE} - 1 = 1.08 \cdot \sqrt{18.80/17.33} - 1 \approx 12.5\%$.

4 Mathematical Development

The constraints and the objective function of a CPP model are defined by *resource types*, the *state tree*, and *project-specific decision trees*. We first define these three concepts and then discuss the constraints and the objective function in CPP.

4.1 Resources

Resources are inputs and outputs that are either consumed or produced by projects. They can be production factors (e.g., money, equipment), intangibles (e.g., intellectual property rights), or other relevant assets that the DM may be interested in. A resource type is denoted by r and the set of all resource types by R.

4.2 States of Nature

The time-state model of CPP is a state tree which represents the structure of future *states of nature*. Each state prevails during one period within the planning horizon $\{0,...,T\}$. The set of states in period t is denoted by S_t, and the set of all states is $S = \bigcup_{t=0}^{T} S_t$. The time period of state $s \in S$ is denoted by $t(s)$.

The state tree starts with a single base state s_0 in period 0. Each state $s' \in S_{t-1}, 0 < t \le T$ is followed by at least one state $s \in S_t$. This relationship is modeled by the function $B: S \to S$ which returns the unique (immediate) predecessor $s' \in S_{t-1}$ of state $s \in S_t, t > 0$, (by convention, $B(s_0) = s_0$). The n-th predecessor of $s \in S_t$ $(t \ge n)$ is defined recursively by $B^n(s) = B(B^{n-1}(s))$, where $B^0(s) = s$. This function can be used to obtain the states on a path from the base state s_0 to state s. These states, together with state s, are contained in $S^B(s) = \{s' \in S \mid \exists k \geq 0 \text{ such that } B^k(s) = s'\}$.

States result through uncertain *events* (e.g., "markets went up"). The probability that state $s \in S_t$ $(t > 0)$ obtains, subject to the assumption that its predecessor $B(s)$ has occurred, is given by the conditional probability $p_{B(s)}(s)$. The base state s_0 occurs with probability one, i.e., $p(s_0) = 1$. Unconditional probabilities for the other states $s \in S_t$ $(t > 0)$ are computed recursively from the equation $p(s) = p_{B(s)}(s) \cdot p(B(s))$.

4.3 Projects

4.3.1 Decision Points

The DM takes decisions with regard to projects $z \in \mathbf{Z}$. Following Gear and Lockett (1973), the decision opportunities for each project are structured as a *decision tree* which consists of *decision points*: that is, for each project $z \in \mathbf{Z}$, there is a set of decision points D_z such that at decision point $d \in D_z$, the DM chooses one of the actions in $a \in A_d$. The decision point at which action a can be taken is $d(a)$. The first decision point of project z is the *base decision point* d_z^0.

At each decision point d, the DM has information about (i) what state $s(d) \in S$ prevails at this point, and (ii) what actions she has taken earlier on with regard to project z, if any; this information is conveyed by the action that immediately preceeds *d*. For all decisions points $d \in D_z$ other than the base decision point, this action, called the *parent action* of d, is given by the function $ap(d)$. It is assumed that the decision points form a consistent tree so that each decision point has a unique parent action.

For each action a there is an action variable X_a that is equal to the number of times that this action is selected at decision point $d(a)$ (e.g., 1 if the action is selected once; 0 if the action is not selected). Apart from binary choices, action variables can be used to model decisions that correspond to nonnegative integers or continuous real numbers.

A project management strategy $\mathbf{X}_z$ is defined by the action variables that are associated with the decision points $d \in D_z$ of project z. *A portfolio management strategy* $\boldsymbol{X}$ is the DM's complete plan of action for all projects $z \in \boldsymbol{Z}$ and all states $s \in S$.

4.3.2 Resource Flows

The project management strategy X_z induces a resource flow $RF_z^r(\boldsymbol{X_z}, s)$ of resource type r in state s. Letting $c_a^r(s)$ denote the flow of resource type r in state s due to action a, this flow is given by

$$RF_z^r(\boldsymbol{X_z}, s) = \sum_{\substack{d \in D_z: \\ s(d) \in S^B(s)}} \sum_{a \in A_d} c_a^r(s) \cdot X_a \ ,$$

where the restriction in the summation of decision points ensures that actions influence resource flows only in the current state and relevant future states. The aggregate resource flow $RF^r(\mathbf{X}, s)$ in state s is obtained by adding the flows for all projects, i.e.,

$$RF^r(\boldsymbol{X}, s) = \sum_{z \in Z} RF_z^r(\boldsymbol{X_z}, s) = \sum_{z \in Z} \sum_{\substack{d \in D_z: \\ s(d) \in S^B(s)}} \sum_{a \in A_d} c_a^r(s) \cdot X_a \ .$$

For the time being, we assume that resource flows are linear in the action variables, which means that interactions among actions (e.g., synergies) are not accounted for. In principle, such interactions can be captured through cross-terms for pairs of action variables. In this case, the aggregate resource flow becomes

$$RF^r(\boldsymbol{X}, s) = \sum_{z \in Z} \sum_{\substack{d \in D_z: \\ s(d) \in S^B(s)}} \sum_{a \in A_d} \sum_{z' \in Z} \sum_{\substack{d' \in D_{z'}: \\ s(d') \in S^B(s)}} \sum_{a' \in A_d} c_{a,a'}^r(s) \cdot X_a \cdot X_{a'} \ .$$

4.4 Constraints

The four main constraint types in CPP are (i) *decision consistency constraints*, (ii) *resource constraints*, (iii) *optional constraints*, and (iv) *deviation constraints*.

4.4.1 Decision Consistency Constraints

The structure of decision points influences how many actions can be selected at a given decision point d. For instance, if the parent action of d was selected once, the DM arrives at d and chooses one of the actions in A_d; but if the parent action was not selected, the DM does not arrive at d so that none of the actions at d can be selected. Thus, at each decision point other than the base decision point, the number of selected actions is the same as the number of times that its parent action was selected. At the base decision point d_z^0, the DM usually chooses one of the alternative actions; however, multiple project instantiations (i.e., exact copies) can be modeled by allowing the number of selected actions to be greater than one, say L_z.

The above requirements imply the following decision consistency constraints

$$\sum_{a \in A_{d_z^0}} X_a = L_z \quad \forall z \in \mathbf{Z} \tag{1}$$

$$\sum_{a \in A_d} X_a = X_{ap(d)} \quad \forall d \in D_z \setminus \{d_z^0\} \quad \forall z \in \mathbf{Z}\,. \tag{2}$$

Unconstrained quantitative decisions, which do not form a decision tree, can be modeled by omitting the constraints (1) and (2). For example, decisions in securities trading can be modeled by associating with each state a continuous unconstrained action variable which is equal to the amount of transactions in that state.

4.4.2 Resource Constraints

Resource constraints can be employed to ensure that there is a nonnegative stock of resources in each state. They are modeled through *resource surpluses* that would remain in state $s \in S$, if the DM were to choose the portfolio strategy $\mathbf{X}$. The surplus of resource type *r* in state $s \in S$ is

$$RS_s^r = \begin{cases} b^r(s) + RF^r(\mathbf{X}, s) & \text{if } s = s_0 \\ b^r(s) + RF^r(\mathbf{X}, s) + \alpha_{B(s)\to s}^r \cdot RS_{B(s)}^r & \text{if } s \neq s_0 \end{cases},$$

where $b^r(s)$ is the initial endowment of resource *r* in state $s \in S$ and $\alpha_{B(s)\to s}^r$ is the rate at which the surplus in state $B(s)$ is transferred to s. Initial endowments cannot be influenced by the DM and they do not entail any costs. The transfer rate may depend on the resource type and the state: it may be equal to (1 + risk-free interest rate) for money, while the rate for perishable goods may be zero.

The *resource surplus variables* RS_s^r are continuous, and for a given portfolio strategy $\mathbf{X}$, they can be solved using the following resource constraints

$$RF^r(\mathbf{X}, s_0) - RS_{s_0}^r = -b^r(s_0) \quad \forall r \in R$$

$$RF^r(\mathbf{X}, s) + \alpha_{B(s)\to s}^r \cdot RS_{B(s)}^r - RS_s^r = -b^r(s) \quad \forall s \in S \ \setminus \ \{s_0\} \quad \forall r \in R\,.$$

Resource surplus variables are usually constrained to non-negative values. However, negative values may also be permitted in order to allow for the possibility to borrow funds, for instance.

4.4.3 Optional Constraints

Optional constraints include any other constraints that may apply. The two most commonly discussed optional constraints in the literature are *prerequisite constraints*, which define relations between follow-up and prerequisite projects, and *project version constraints* which model alternative versions of a project. Additional examples on optional constraints are given by Ghasemzadeh et al. (1999). Note that in CPP project versions can be modeled with a single project in which the choice among the project versions is made at the base decision point so that no dedicated constraints for this purpose are needed.

4.5 Objective Function

The DM seeks to maximize the utility of the terminal resource position, viz.

$$\max U[X],$$

where U is the DM's preference functional and X is the value of the resource position in period T. Under expected utility theory, the preference functional is given by $U\left[X\right]=E\left[u(X)\right]$, where u is the DM's von Neumann-Morgenstern utility function.

4.5.1 Linear Preference Models

Because nonlinear utility functions can entail computational challenges in the context of large-scale portfolios, we focus on two special cases of the objective function that (i) implement a reasonable model of risk aversion and (ii) lead to a linear programming model. In both cases, we assume that the DM's preference functional can be approximated as a mean-risk model; such models have been widely used in the field of portfolio selection (see Markowitz 1952, 1959).

We also assume that the value of the final resource position is (i) additive with regard to resource types and (ii) linear with respect to the amount of surplus of each resource (see, e.g., Keeney and Raiffa 1976). These two latter assumptions are not overly restrictive as linear pricing is widely employed in financial modeling. For a given state, the total value of resource surpluses can be obtained by associating state-dependent weights w_s^r with each resource type. These weights can be interpreted as *unit prices* so that the *monetary value* of resource surpluses in state s is given by

$$V_s(\boldsymbol{RS}_\mathbf{s}) = \sum_{r \in R} w_s^r \cdot RS_s^r,$$

where w_s^r is the unit price of resource type r in state $s \in S_T$. The expected (monetary) value of the terminal resource position is thus given by

$$EV_T(\boldsymbol{RS}_\mathbf{T}) = \sum_{s \in S_T} p(s) \cdot V_s\ (\boldsymbol{RS}_\mathbf{s}) = \sum_{s \in S_T} p(s) \cdot \sum_{r \in R} w_s^r \cdot RS_s^r\ , \tag{3}$$

where $\boldsymbol{RS}_T$ is a vector of all RS_s^r's for which $r \in R$ and $s \in S_T$.

4.5.2 Risk Measures

Several dispersion statistics have been proposed in the literature on portfolio selection. Markowitz (1952, 1959) suggests the use of variance and semivariance for the selection of securities. Expected downside risk (EDR) has been employed in capacity planning (Eppen et al. 1989), while absolute deviation has been applied in real-time stock market analysis (Konno and Yamazaki 1991). These last two measures are linear, which make them attractive for large-scale portfolio selection problems. Also, these measures lead to mean-risk models that are consistent with the first and second degrees of stochastic dominance (FSD and SSD; Levy 1992, Ogryczak and Ruszczynski 1999, Fishburn 1977), which suggests that the resulting models are theoretically reasonable.

In particular, we employ *expected downside risk* (EDR; Eppen et al. 1989, Fishburn 1977) and *lower semi-absolute deviation* (LSAD) as measures of risk. Specifically, EDR is given by

$$EDR[X] = \sum_{\substack{\text{all } x: \\ x<t}} p(x) \,|\, x - t \,| = \sum_{\substack{\text{all } x: \\ x<t}} p(x)(t - x)\,,$$

where $p(x)$ is the probability mass function of X and t is the target value from which deviations are computed. When the target value is equal to the expression $t = \mu_X = E[X]$, we have

$$LSAD[X] = \sum_{\substack{\text{all } x: \\ x<\mu_X}} p(x) \,|\, x - \mu_X \,| = \sum_{\substack{\text{all } x: \\ x<\mu_X}} p(x)(\mu_X - x)\,.$$

Both measures can be calculated from deviation constraints as follows. Let ΔV_s^+ and ΔV_s^- be nonnegative *deviation variables* which measure how much the total value of the resource surpluses in state $s \in S_T$ (i.e., V_s) differs from the risk measure's target value t. For EDR, these variables satisfy the equations

$$V_s\ (\mathbf{RS}_\mathbf{s}) - t - \Delta V_s^+ + \Delta V_s^- = 0 \qquad \forall s \in S_T\,, \tag{4}$$

where only one of the variables ΔV_s^+ and ΔV_s^- can be positive, because ΔV_s^- has a negative coefficient in the objective function. The EDR of the value of the final resource position is given by the sum

$$\sum_{s \in S_T} p(s) \cdot \Delta V_s^- . \tag{5}$$

The LSAD measure can be computed by using $t = EV_T(\boldsymbol{RS}_T)$ in (4) instead of a fixed target value t. This leads to

$$V_s(\mathbf{RS_s}) - EV_T(\boldsymbol{RS}_T) - \Delta V_s^+ + \Delta V_s^- = 0 \quad \forall s \in S_T , \tag{6}$$

whereafter the LSAD can be obtained from (5).

4.5.3 Mean-Risk Model

The objective function can now be stated in the mean-risk form

$$\max EV_T(\mathbf{RS_T}) - RP_T(\mathbf{\Delta V_T^-}) , \tag{7}$$

where EV_T is defined by Equation (3), and $RP_T(\Delta \mathbf{V}_T^-)$ is given by

$$RP_T(\mathbf{\Delta V_T^-}) = \lambda \cdot \sum_{s \in S_T} p(s) \cdot \Delta V_s^- , \tag{8}$$

where deviation variables ΔV_s^- terms are obtained either from (4) (EDR) or (6) (LSAD). When the mean-risk model gives a certainty equivalent for a random variable, like the mean-LSAD model does, RP_T can be interpreted as the DM's *risk premium*. By substituting Equations (3) and (8) into (7), the objective function becomes

$$\max \left(\sum_{s \in S_T} p(s) \left(\sum_{r \in R} w_s^r \cdot RS_s^r - \lambda \cdot \Delta V_s^- \right) \right).$$

Importantly, the mean-EDR model falls within the scope of expected utility theory[1] (Fishburn 1977), and can therefore be regarded as an acceptable model of risk aversion; in particular, it does not suffer from *dynamic inconsistencies* (Machina 1989), which may occur with the mean-LSAD model and other non-expected utility models. On the other hand, the mean-LSAD model can be motivated by its link to disappointment models and standard measures of risk (Jia and Dyer 1996, Jia et al. 2001) and the properties of *constant abso-*

---

[1] Since $E[X] - \lambda \cdot EDR[X] = \sum_{\substack{\text{all } x: \\ x<t}} p(x)\left(x - \lambda(t-x)\right) + \sum_{\substack{\text{all } x: \\ x \geq t}} p(x)x$, the mean-EDR model is equivalent to the utility function $u(x) = \begin{cases} (1+\lambda)x - \lambda t, & x < t \\ x, & x \geq t \end{cases}$.

lute risk aversion and *constant relative risk aversion* (see French 1986): if the value of the final resource position is subjected to a positive affine transformation, the certainty equivalent undergoes a similar transformation so that $CE[a \cdot X + b] = a \cdot CE[X] + b$ for constants $a > 0$ and b. In expected utility theory, no utility function implies both risk aversion and such a linear pricing property.

5 Complexity Analysis

5.1 Model Size

Although CPP is based on linear programming, the required computational effort may become prohibitive if the model is very large. It is therefore instructive to examine the number of decision variables and constraints in a CPP model (see Tables 1 and 2). Here, $D = \bigcup_{z \in Z} D_z$ is the set of all decision points and $A = \bigcup_{z \in Z} \bigcup_{d \in D_z} A_d$ is the set of all actions.

Because deviation variables and resource surplus variables are continuous, there are at most $|A|$ integer variables in a CPP model. The number of integer variables can be reduced by *not* constraining one of the action variables at each decision point to integer values. Still, this variable can assume integer values only, because it is related to the other integer-valued actions at the same decision point through Equations (1) and (2). The upper bound for the number of integer variables can thus be reduced to $|A| - |D|$.

Table 1 Number of Decision Variables

Decision variable	**Number**	**Type**
Action variables (X's)	$\|A\|$	Typically integer
Resource surplus variables (RS's)	$\|S\| \cdot \|R\|$	Continuous
Deviation variables (ΔV's)	$2 \cdot \|S_T\|$	Continuous
TOTAL	$\|A\| + \|S\| \cdot \|R\| + 2 \cdot \|S_T\|$	

Table 2 Number of Constraints

Constraint	**Number**
Decision consistency constraints	$\|D\|$
Resource constraints	$\|S\| \cdot \|R\|$
Deviation constraints	$\|S_T\|$
Optional constraints	O
TOTAL	$\|D\| + \|S\| \cdot \|R\| + \|S_T\| + O$

For example, a CPP model with three resource types, five time periods, a binary state tree (where each state is split into two further states in the next period) and thirty projects with four consecutive "go/no go" decisions leads to $15 \cdot 30 = 450$ integer action variables and equally many continuous action variables, as well as $31 \cdot 3 = 93$ continuous resource surplus variables and $2 \cdot 16 = 32$ continuous deviation variables (cf. Table 1). This leads to a total of 450 integer variables and 575 continuous variables. The number of constraints is $450 + 31 \cdot 3 + 16 + 0 = 559$. This model can be readily solved using standard techniques of mixed integer programming (MIP).

In comparison, a conventional decision tree for the same portfolio selection problem would contain an enormous number of decision and chance nodes. The first decision node of the tree would entail 2^{30} alternative decisions, one for each possible project portfolio, and each of these decisions would lead to a binary chance node that resolves at the end of the first period. At the start of the second period, there would be $2^{30} \cdot 2$ decision nodes, each preceded by the earlier portfolio decision and the chance outcome. Assuming that these and ensuing decision nodes entail at most 2^{30} decision alternatives each, we obtain an upper limit of $1 + 2^{30} \cdot 2 + (2^{30} \cdot 2)^2 + (2^{30} \cdot 2)^3 \approx 10^{28}$ for the number of decision nodes. Apart from being intractable, the resulting tree would still require that the sufficiency of resources is verified at each decision node. On the other hand, building a separate conventional decision tree for each project would not support the consideration of project interactions or the variability of portfolio returns. Nevertheless, in this case, each decision tree would contain 15 decision nodes and 15 chance nodes, resulting in a total of 450 decision nodes and equally many chance nodes.

A challenge with all state tree based approaches, including CPP, is that the size of the state tree can become excessively large if the outcome of the project portfolio depends on a large number of risk factors. This is often the case when the outcome of each project depends on a project-specific risk factor, because the number of states then increases exponentially with the number of projects. For example, in a setting where each project either fails or succeeds and the success of each project is independent of that of other projects, the number of corresponding terminal states becomes 2^n, where n is the number of projects. With a sufficiently large n, this necessarily leads to a CPP model that cannot be solved in a reasonable time.

The above situation does not arise if CPP is used to capture *external* uncertainties, such as market risks and regulatory changes that are not influenced by the projects. This focus on external uncertainties is natural in CPP, because the state tree is shared by all projects and project decisions do not influence state probabilities. Hence, CPP may be particularly suitable for *scenario analysis* where portfolio management strategies are optimized with regard to a relatively small number of states.

5.2 Computational Experiments

The computational performance of CPP models was tested with a dedicated C++-application that runs under Windows XP operation system using LP Solve 4.0.1.9 software package for LP and MIP models (available at ftp://ftp.es.ele.tue.nl/pub/lp_solve/). The optimizations were run on a laptop computer with 512 MB of memory and 1.06 GHz Pentium III processor.

5.2.1 Experimental Setup

In our numerical experiments, the number of projects, stages per project, time periods, and resources varied as described in Tables 3 and 4. For each model type, 30 models with randomized resource flows and state probabilities were generated. The timeout for the solution algorithm was set to 20 minutes to ensure that the total computation time remained reasonable even in cases where the median solution time was several minutes and the worst case solution time could have been several hours.

INSERT TABLES 3 AND 4 AROUND HERE

In most cases, we used two resource types, (i) money with the risk-free interest rate of 5% and (ii) a perishable capacity resource with a zero transfer rate. Any additional resource types, if present, were perishable capacity resources. The weight (i.e., unit price) of capacity resources in the objective function was 0 and that of money was 1. In the first period, the initial budget for money was $2 million multiplied by the number of projects, while initial monetary endowments in other periods were zero. In each period, initial endowments for other resources were equal to one unit of resources multiplied by the number of projects.

The state tree had a binary structure such that each non-terminal state was split into two states in the next period. The probabilities for the terminal states were computed by generating real numbers from the uniform distribution over the unit interval and by normalizing the resulting numbers. The probabilities of the other states were aggregated from those of the terminal states.

Project decision trees consisted of binary "go/no go"-decisions in two or more stages. The first decision was taken in period 0, followed by the next decision in every consecutive period up to the total number of stages. Each "go"-decision entailed an immediate cost. Each cost was obtained by (i) deciding on the most likely value of the cost and (ii) by multiplying this value with a random number from the (0,1)-lognormal distribution, implying a lognormal distribution for all costs. The most likely values for costs were assumed to rise linearly with time so that later stages were more expensive to carry out than earlier ones. We used the value of $1 million for the first stage, implying a cost of $2 million of the second stage, $3 million for the third stage, and so on. Costs for other resources were modeled similarly, using a most likely value of 1 resource unit for the first stage and a linear growth model.

The revenues began one period after the last "go"-decision. Similarly to costs, revenues were randomized by first selecting a most likely value for the revenue and then multiplying this number by a random number drawn from the (0,1)-lognormal distribution. For each project, revenues were distributed evenly over time, in the sense that the most likely revenue was the same for each period. The cumulative sum of the most likely values for revenues over the time horizon was assumed to be 1.15 times the sum of most likely values for costs over the time horizon. For example, for a two-staged project the sum of the most likely values for costs was $3 million and hence the sum of the most likely values for revenues was $3.45 million. In a model with four periods, this would imply that the most likely revenue in the last period and in the second last period was $1.725 million. Projects did not yield inflows for other resources.

In most cases, a mean-LSAD model with the risk aversion coefficient $\lambda = 0.5$ was employed. We also conducted experiments with the mean-EDR model with $\lambda = 0.5$ and the target value of $b \cdot 1.05^T$, where b is the initial budget and T denotes the last time period. These are indicated by the symbol ¤ in Tables 3 and 4.

5.2.2 Results

Some CPP models were solved as MIP models (Table 4) and some as LP models where integer variables were left continuous (Table 3). The results for LP models suggest that CPP models for realistic portfolio selection problems can be solved in a reasonable time, and that relatively few projects involve action variables with non-integer values. For example, CPP models with 250 3-staged projects, 2 resources, and 5 time periods (16 terminal states) were solved in a median time of 46.7 s. Models with 50 5-staged projects, 2 resources, and 9 time periods (256 terminal states) had a median solution time of 6 min 59 s. Typically, LP models with less than 6,000 variables and 3,000 constraints could be solved within the 20-minute time-out limit. In most cases, the number of non-integer action variables was small, about 3%–10% of all action variables.

In LP models, the possibility to borrow additional resources and the DM's risk neutrality led to (i) a lower number of fractional integer variables and (ii) a higher probability of attaining an integer solution (Table 3). In the presence of both assumptions, action variables always assumed integer values, suggesting that fractional project management decisions were either due to limited resources or the DM's risk aversion.

Typically, MIP models were much more time-consuming to solve than LP models (Table 4): for example, an MIP model with 30 3-staged projects, 2 resources, and 5 time periods could be solved in a median time of 49.4 s, while the LP model took only 0.32 s to solve. The mean and standard deviation of the solution time seemed to grow exponentially with the number of integer variables, wherefore models with more than 350 integer variables could not usually be solved within the 20-minute time-out limit. However, the possibility to borrow resulted in significantly shorter solution times, as models with 100 3-staged projects, 1 resource, and 5 time periods were solved in a median time of 3.335 s which is not much more than the median solution time of 2.835 s for LP models. When both the possibility to borrow and the assumption of risk neutrality were introduced, there was no significant difference in the solution times of MIP and LP models.

6 Summary and Conclusion

The CPP modeling framework presented in this paper is applicable to the portfolio management of correlated R&D projects and, more generally, to the analysis of investment problems where the dynamics and interdependencies of risky investment opportunities must be accounted for. This framework has several appealing characteristics, such as the explicit consideration of resource dynamics and managerial flexibility. It also accommodates a wide range of risk attitudes, including two risk averse preference models that lead to linear CPP models.

Our simulations indicate that LP and MIP formulations for CPP models of realistic size can be solved in a reasonable time. LP models of about a hundred five-staged projects and several hundreds of states can be solved in a reasonable time by using a standard personal computer and a public domain C++ LP package. MIP models, on the other hand, usually take much more time solve; MIP models with a couple of tens of three-staged projects and less than a hundred states have usually an acceptable solution time.

There are several avenues for further research. On the theoretical side, CPP needs to be extended to settings where more complex resource dynamics must be accounted for (e.g., cost of storage, proactive management of multi-purpose resources) or where the DM's actions influence the structure of the state tree. In terms of future applications, CPP seems particularly useful in settings where separate decision trees for each project can be developed, but where the optimal decisions are interlinked by resource constraints and the need for a portfolio management strategy which accounts for the DM's risk attitude.

7 References

Birge, J. R., F. Louveaux. 1997. *Introduction to Stochastic Programming*. Springer, New York, NY.

Dixit, A. K., R. S. Pindyck. 1994. *Investment Under Uncertainty*. Princeton University Press, Princeton, NJ.

Eppen, G. D., R. K. Martin, L. Schrage. 1989. A scenario approach to capacity planning. *Operations Research* **37**(4) 517–527.

Fishburn, P. C. 1977. Mean-Risk Analysis with Risk Associated with Below-Target Returns. *Amer. Econ. Rev.* **67**(2) 116–126.

French, S. 1986. *Decision Theory – An Introduction to the Mathematics of Rationality*. Ellis Horwood Limited, Chichester.

Gear, A. E., A. G. Lockett. 1973. A Dynamic Model of Some Multistage Aspects of Research and Development Portfolios, *IEEE Trans. Eng. Management* **EM20**(1) 22–29.

Ghasemzadeh, F., N. Archer, P. Iyogun. 1999. A zero-one model for project portfolio selection and scheduling. *J. Oper. Res. Soc.* **50**(7) 745–755.

Heidenberger, K. 1996. Dynamic project selection and funding under risk: A decision tree based MILP approach. *European J. Oper. Res.* **95**(2) 284–298.

Henriksen, A., A. Traynor. 1999. A Practical R&D Project-Selection Scoring Tool. *IEEE Trans. Eng. Management* **46**(2) 158–170.

Jia, J. J. S. Dyer. 1996. A standard measure of risk and risk-value models. *Management Science* **42**(12) 1691–1705.

Jia, J., J. S. Dyer, J. C. Butler. 2001. Generalized Disappointment Models. *Journal of Risk and Uncertainty* **22**(1) 59–78.

Keeney, R. L., H. Raiffa. 1976. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. John Wiley & Sons, New York, NY.

Konno, H., H. Yamazaki. 1991. Mean-Absolute Deviation Portfolio Optimization and Its Applications to the Tokyo Stock Market. *Management Sci.* **37**(5) 519–531.

Levy, H. 1992. Stochastic Dominance and Expected Utility: Survey and Analysis. *Management Science* **38**(4) 555–593.

Luenberger, D. G. 1998. *Investment Science*. Oxford University Press, New York, NY.

Machina, M. 1989. Dynamic Consistency and Non-Expected Utility Models of Choice Under Risk. *Journal of Economic Literature* **27**(4) 1622–1668.

Markowitz, H. M. 1952. Portfolio Selection. *J. Finance* **7**(1) 77–91.

———. 1959. *Portfolio Selection: Efficient Diversification of Investments*. Cowles Foundation, Yale.

Martino, J. P. 1995. *Research and Development Project Selection*. John Wiley & Sons, New York, NY.

Mulvey, J. M., G. Gould, C. Morgan. 2000. An Asset and Liability Management Model for Towers Perrin-Tillinghast. *Interfaces* **30**(1) 96–114.

Ogryczak, W., A. Ruszczynski. 1999. From stochastic dominance to mean-risk models: Semideviations as risk measures. *European J. Oper. Res.* **116**(1) 33–50.

Smith, J. E., R. F. Nau. 1995. Valuing Risky Projects: Option Pricing Theory and Decision Analysis. *Management Sci.* **41**(5) 795–816.

Trigeorgis, L. 1996. *Real Options: Managerial Flexibility and Strategy in Resource Allocation.* MIT Press, Massachussets.

8 Acknowledgments

This research has been supported by the Academy of Finland. We would like to thank area editor James S. Dyer, an anonymous associate editor, and three anonymous referees for their helpful comments on earlier versions of this paper. This work has also benefited from discussions with Victor De Miguel, Jaakko Dietrich, Tommi Gustafsson, Toni Jarimo, Juha Koljonen, Pertti Laininen, Harry M. Markowitz, and Juha Ojala. We would also like to thank Kjell Eikland for his invaluable help with LP Solve software package. The financial support of the Finnish Cultural Foundation, and the Foundation of Jenny and Antti Wihuri is gratefully acknowledged.

Table 3 Solution times for LP CPP models. 30 iterations were performed per setting.

Projects	Stages	Periods	Resources	No. of variables	No. of constraints	No. of integer variables	Solution time (s) Expectation	Standard deviation	25% quartile	Median	75% quartile	No. of fractional integer variables Expectation	Standard deviation	25% quartile	Median	75% quartile	% integer solutions
20	3	5	2	374	218	140	0.184	0.028	0.161	0.18	0.201	13.67	7.24	8	13	18	3 %
30	3	5	2	514	288	210	0.332	0.064	0.3	0.32	0.35	14.07	6.91	10	14	16	0 %
60	3	5	2	934	498	420	1.208	0.232	0.992	1.152	1.312	15.03	6.57	10	13	19	0 %
100	3	5	2	1494	778	700	4.11	0.671	3.645	4.016	4.516	19.03	7.53	12	19	23	0 %
250	3	5	2	3594	1828	1750	47.541	7.042	43.032	46.738	50.923	23.53	5,75	19	23	26	0 %
100	4	5	2	3094	1578	1500	19.353	5.178	15.723	18.015	19.348	47.23	11.92	36	46	58	0 %
100	3	6	2	1590	858	700	7.272	1.352	6.299	6.9	8.322	24.6	7.89	19	24	29	0 %
100	4	6	2	3190	1658	1500	35.672	5.917	30.884	35.451	39.036	58.97	15.51	46	57	65	0 %
100	5	6	2	6390	3258	3100	285.30	77.21	214.92	278.89	335.79	126.9	34.99	100	120	140	0 %
100	4	9	2	4534	2778	1500	481.58	111.30	374.06	470.59	550.49	65.1	16.8	53	64	69	0 %
25	5	9	2	3084	2053	775	145.95	38.66	117.64	141.68	166.22	123.27	31.25	94	117	145	0 %
50	5	9	2	4634	2828	1550	413.54	100.53	318.94	418.52	452.75	152.53	38.19	115	146	181	0 %
30	4	6	5	1279	797	450	4.821	0.791	4.226	4.667	5.198	60.33	17.34	45	57	72	0 %
100	4	6	5	3379	1847	1500	73.554	25.105	53.196	64.153	77.932	65.8	18.72	49	61	81	0%
20	3	5	1	343	187	140	0.146	0.031	0.13	0.14	0.15	12.93	5.35	10	11	15	0 %
100	3	5	1	1463	747	700	3.386	0.598	2.884	3.195	3.745	19.6	6.16	14	19	21	0 %
100*	3	5	1	1463	747	700	3.167	1.017	2.674	2.835	3.004	6.47	4.64	2	7	10	20 %
100#	3	5	1	1463	747	700	2.869	0.402	2.574	2.804	3.145	11.6	3.46	9	11	13	0 %
100*#	3	5	1	1463	747	700	2.45	0.493	2.263	2.323	2.423	0	0	0	0	0	100 %
1000*#	3	5	1	14063	7047	7000	618.18	268.27	323.91	605.19	827.79	0	0	0	0	0	100 %
100¤	3	5	1	1463	747	700	3.047	0.401	2.704	3.004	3.194	11.87	3.9	9	12	13	0 %
100¤	3	5	2	1494	778	700	3.581	0.437	3.255	3.545	3.936	12.7	3.45	10	12	14	0 %

*: borrowing is allowed, #: risk neutrality, ¤: mean-EDR model.

Table 4 Solution times for MIP CPP models. 30 iterations were performed per setting.

							Solution time (s)					
Projects	Stages	Periods	Resources	No. of variables	No. of constraints	No. of integer variables	Expectation	Standard deviation	25% quartile	Median	75% quartile	Timeouts (1200 s)
10	3	5	2	234	148	70	1.626	1.861	0.471	0.952	1.552	0
15	3	5	2	304	183	105	7.807	22.855	0.971	1.622	4.076	0
20	3	5	2	374	218	140	10.371	12.612	2.484	7.05	12.147	0
25	3	5	2	444	253	175	41.841	58.301	6.35	22.442	43.943	0
30	3	5	2	514	288	210	128.988	196.7	7.28	49.391	130.498	0
35	3	5	2	584	324	245	242.688	319.226	30.263	99.162	277.799	1
40	3	5	2	654	358	280	406.089	389.605	90.05	285.33	484.627	4
20	3	5	1	343	187	140	7.394	11.98	0.912	2.203	9.383	0
20*	3	5	1	343	187	140	0.247	0.073	0.21	0.23	0.24	0
60*	3	5	1	903	467	420	1.573	0.648	1.141	1.272	1.773	0
100*	3	5	1	1463	747	700	4.31	2.402	2.904	3.355	4.727	0
200*	3	5	1	2863	1447	1400	26.062	16.252	14.661	18.917	28.772	0
100#	3	5	1	1463	747	700	> 1200	-	> 1200	> 1200	> 1200	30
10¤	3	5	2	234	148	70	1.626	3.15	0.32	0.691	1.412	0

*: borrowing is allowed, #: risk neutrality, ¤: mean-EDR model.