Multi-Range Robust Optimization vs Stochastic Programming in Prioritizing Project Selection

Ruken Düzgün* Aurélie Thiele†

July 2012

Abstract

This paper describes a multi-range robust optimization approach applied to the problem of capacity investment under uncertainty. In multi-range robust optimization, an uncertain parameter is allowed to take values from more than one uncertainty range. We consider a number of possible projects with anticipated costs and cash flows, and an investment decision to be made under budget limitations. Uncertainty in parameter values – in our case, the cost and net present values of each project – could significantly impede the real-life viability of the suggested investment plan. We set up the multi-range robust optimization so that the possible values taken by the uncertain parameters match the three possible values of the cost or net present value distributions in the stochastic programming approach that we use as benchmark. While the stochastic programming approach suffers from tractability issues, the robust optimization approach solves the same capacity investment problem in seconds. We also show how to compute the project prioritization list to substantially decrease computation time.

1 Introduction

Data in real-life applications is often not known precisely at the time when the manager must make a decision. In this paper we are interested in the problem of capacity investment under uncertainty. We have a number of possible projects with anticipated costs and cash flows to choose

*Senior Programmer Analyst, Marriott International, Bethesda, MD rud207@lehigh.edu Work funded in part by NSF Grant NSF-0540143

†Associate Professor, Department of Industrial and Systems Engineering, Lehigh University, Bethlehem. aurelie.thiele@lehigh.edu Work funded in part by NSF Grant NSF-0540143. Corresponding author.

from, and must make an investment decision under budget limitations. Cost and net present values of the projects are uncertain. Specifically, the problem we consider will be to maximize the net present value (NPV) of the selected projects while ensuring that the total cost does not violate the budget on hand, as is typical in the literature. Two broad types of methodologies address data uncertainty: (i) stochastic (and dynamic) programming, pioneered by Dantzig (1955), which models uncertainty as random variables with known distributions and optimizes the expected value of the objective, and (ii) robust optimization, first suggested by Soyster (1973), which models uncertainty as uncertain parameters belonging to a known uncertainty set and optimizes the worst case over that set. The model in Soyster (1973) required that each uncertain parameter be equal to its worst-case value, and thus was deemed too conservative for practical implementation. In the mid-1990s, Ben-Tal and Nemirovski (1998, 1999, 2000), El-Ghaoui and Lebret (1997) and El-Ghaoui et al. (1998) proposed a tractable mathematical reformulation under ellipsoidal uncertainty sets that turn linear programming problems into second-order cone problems, while reducing the conservatism of the approach in Soyster (1973).

Bertsimas and Sim (2003, 2004) and Bertsimas et al. (2004) investigate the case where the uncertainty set is a polyhedron. Specifically, the main uncertainty set they studied consists of range forecasts (confidence intervals) for each parameter and a constraint called a budget-of-uncertainty constraint, which limits the number of coefficients that can take their worst-case value. The approach preserves the degree of complexity of the problem (the robust counterpart of a linear problem is linear) and allows the decision-maker to control the degree of conservatism of the solution. The reader is referred to Bertsimas et al. (2007) for a comprehensive review paper of robust optimization and Ben-Tal et al. (2009) for a book treatment of the topic.

The traditional robust optimization framework defines the *scaled deviations* of the parameters from their nominal values and reformulates the problem of optimizing the worst-case objective over that set in a tractable manner by either computing the worst-case value using convex optimization ideas (for ellipsoidal sets) or by invoking strong duality (for polyhedral sets). Because our application of project selection requires us to use binary variables and because the robust counterparts of linear problems under ellipsoidal sets are nonlinear, we will only consider the robust optimization approach using polyhedral sets here, so that we can preserve the linearity of the problem. We are especially interested in polyhedral sets built upon range forecasts of the uncertain coefficients and budgets of uncertainty due to their intuitive nature that appeals to practitioners. Robust optimization with polyhedral sets of this structure, however, reduces the uncertainty range of each coefficient to two values: the nominal value (if the budget of uncertainty is not used for that coefficient) and the worst-case value (if the budget of uncertainty is used), assuming the budget of uncertainty is integer. It is therefore not possible to capture the shape of the distribution, which can be considered a drawback by practitioners.

Our goal in this paper is to investigate the merits of an approach based on a concept called *multi-range robust optimization*, which was developed in Author1 and Author2 (2010), for the specific setting of project selection and prioritization presented in Koc et al. (2009). The paper by Koc et al. (2009) was selected as the benchmark because the authors implement a stochastic programming framework to a real-life problem using real data, and the main "selling point" of robust optimization has long been that it is more tractable than stochastic programming in real-life applications. (Another selling point is related to the difficulty in estimating underlying probabilities correctly in the stochastic framework, but we will not consider this point here.) In contrast with Author1 and Author2 (2010), the goal of which was to extend robust optimization to the case where the value taken by an underlying discrete random variable – such as the strength of a new compound, driving the performance of a drug under development – determined the range of values taken by the net present value, the main contribution of the present paper is to present multi-range robust optimization as a tractable alternative to stochastic programming when the budgets of uncertainty are set appropriately based on the probabilities of the stochastic programming model. (We note that Bienstock (2007) has studied histogram models in the context of robust portfolio optimization, but his approach uses a cutting-plane algorithm that is radically different from the multi-range robust optimization approach proposed here, which relies on total unimodularity.) A secondary contribution of the paper is that we show how to compute the project priority list in a far more efficient manner than what was proposed in Koc et al. (2009), thus substantially reducing computation times. When testing our approach using the problem setup provided in Koc et al. (2009), the stochastic programming approach does not solve within the allotted time while our robust optimization approach solves its model to optimality within seconds.

The rest of the paper is structured as follows. In Section 2, we review the robust optimization approach we proposed in Author1 and Author2 (2010). In Section 3, we describe how a simple change to the model implemented by Koc et al. (2009) will drastically improve the solution time of the stochastic programming problem by providing a more computationally efficient way of computing the project priority list. Section 4 presents the robust optimization formulation in the proposed setting, while Section 5 provides the details of the numerical implementation. Section 6 contains concluding remarks.

2 Problem Overview

2.1 Review of Multi-Range Robust Optimization

Consider the general model:

$$\begin{aligned} \max \quad & \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{1}$$

where $\mathcal{X}$ is the constraint set of the decision vector $\mathbf{x}$ and $\mathbf{c}$ is the vector of objective coefficients, which are subject to uncertainty in our setting. Because of the project selection application, it is appropriate to assume that all decision variables are non-negative; in fact, it is a natural assumption to make in the context of operations management in general. Author1 and Author2 (2010) present a robust optimization approach with multiple ranges for non-negative decision variables. In that approach, each parameter can belong to one of m ranges; the decision-maker limits the number of coefficients that fall within each range as well as the number of coefficients that deviate from their nominal value in a given range. This is formulated in mathematical terms as follows.

Let c_i^{k-}, resp. c_i^{k+} be the lower, resp. higher, bound of range k for parameter i, $i = 1, \ldots, n$, $k = 1, \ldots, m$. The budget Γ_k constrains the maximum number of coefficients that can fall within range k, $k = 1, \ldots, m$. Therefore, the uncertain coefficients satisfy:

$$\begin{aligned} & c_i = \sum_{k=1}^{m} c_i^k, \ \forall i, \\ & c_i^{k-}\, y_i^k \le c_i^k \le c_i^{k+}\, y_i^k, \ \forall i, k, \\ & \sum_{k=1}^{m} y_i^k = 1, \ \forall i, \\ & \sum_{i=1}^{n} y_i^k \le \Gamma_k, \ \forall k, \\ & y_i^k \in \{0, 1\}, \ \forall i, k. \end{aligned}$$

with $y_i^k = 1$ when parameter i is in range k and 0 otherwise, for all i and k.

Note that the budgets Γ_k will be chosen so that $\sum_{k=1}^{m} \Gamma_k \ge n$ to ensure that each coefficient can be assigned a range. Alternatively, the decision maker can specify the Γ_k for the $m-1$ lowest (most pessimistic) ranges and put the remaining coefficients in the last range.

We will denote the *nominal value* of parameter i in range k, by $\bar{c}_i^k$ for all $i = 1, \ldots, n$ and $k = 1, \ldots, m$. The *measure of uncertainty* for parameter i of range k is then defined as $\hat{c}_i^k = \bar{c}_i^k - c_i^{k-}$ for all $i = 1, \ldots, n$ and $k = 1, \ldots, m$. Because the decision variables are non-negative, the part of the range forecast above the nominal value will not be used in the robust optimization approach and the uncertain coefficients can be represented as:

$$c_i = \sum_{k=1}^{m} (\bar{c}_i^k - \hat{c}_i^k \, z_i^k) \, y_i^k,$$

where z_i^k is the scaled deviation of coefficient i, $i = 1, \ldots, n$, from its nominal value in range k, $k = 1, \ldots, m$ with:

$$\begin{aligned} &\sum_{i=1}^{n} \sum_{k=1}^{m} z_i^k \leq \Gamma, \\ &0 \leq z_i^k \leq 1, \ \forall i, k. \end{aligned}$$

$\Gamma \in [0, n]$ is the *budget of uncertainty* which specifies the maximum number of coefficients that can deviate from their nominal values. For $\Gamma = 0$, resp. $\Gamma = n$, all parameters will be equal to the nominal, resp. worst-case, value of the range they fall in.

For any feasible $\mathbf{x} \in \mathcal{X}$, the worst-case objective can then be computed as a mixed-integer programming problem:

$$\begin{array}{lll} \min\limits_{\mathbf{c},\mathbf{y}} & \displaystyle\sum_{i=1}^{n} \sum_{k=1}^{m} x_i \left(\bar{c}_i^k \, y_i^k - \hat{c}_i^k \, u_i^k \right) & \\ \text{s.t.} & u_i^k \leq y_i^k, & \forall i, k, \\ & \displaystyle\sum_{i=1}^{n} \sum_{k=1}^{m} u_i^k \leq \Gamma, & \\ & \displaystyle\sum_{k=1}^{m} y_i^k = 1, & \forall i, \\ & \displaystyle\sum_{i=1}^{n} y_i^k \leq \Gamma_k, & \forall k, \\ & y_i^k \in \{0, 1\}, & \forall i, k, \\ & u_i^k \geq 0, & \forall i, k. \end{array} \tag{2}$$

Author1 and Author2 (2010) proves that the constraint matrix of Problem (2) is totally unimodular. Thus, the linear relaxation of the problem has integer optimal solutions, given that the right-hand-side values of the constraints are integer. This allows us to invoke strong duality and convert the max-min problem into a large maximization problem with a linear structure:

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} p_i - \sum_{i=1}^{n}\sum_{k=1}^{m} z_i^k - \sum_{k=1}^{m} \gamma^k \,\Gamma_k - \Gamma\, \gamma_0 & \\
\text{s.t.} \quad & \pi_i^k + \gamma_0 \geq \hat{c}_i^k \, x_i, & \forall i, k, \\
& \pi_i^k + p_i - \gamma^k - z_i^k \leq \bar{c}_i^k \, x_i, & \forall i, k, \\
& \mathbf{x} \in \mathcal{X} & \\
& \gamma^k, \gamma_0, \pi_i^k, z_i^k \geq 0 & \forall i, k.
\end{aligned} \tag{3}
$$

2.2 Preliminaries

Our goal in this paper is to apply the robust optimization approach with multiple ranges to a capacity investment problem where cost and net present values (NPV) of the projects are uncertain. Our study is motivated by Koc et al. (2009), who consider an investment problem with cost and NPV uncertainties. They use a company-made analysis of the projects, provided by South Texas Project Nuclear Operating Company, which lists the anticipated NPV and costs in three possible scenarios: pessimistic, optimistic and most likely cases. The analysis also categorizes the projects in two groups: low-risk and medium-risk. Koc et al. (2009) compute a priority list so that the decision-maker can adjust immediately to changes in the budget (capacity) by implementing a greedy approach, i.e., she will go down the priority list selecting projects until capacity has been filled. In this paper we are interested in the approach called *Optimal Project Prioritization*, where Koc et al. (2009) formulate an optimization model that incorporates budget, cost and profit scenarios and outputs an *optimal priority list*.

Our robust optimization approach differs from Koc et al. (2009) at three levels.

- First, we develop a robust optimization model where we optimize the project portfolio performance and provide a robust priority list, which would still be viable under the worst cases of the cost and NPV outcomes as defined by our uncertainty set. Koc et al. (2009), on the other hand, models the problem as a two-stage stochastic programming problem and maximizes the expected NPV of the selected costs calculated over predefined scenarios.

- We do not consider uncertainty on the right-hand side (RHS) of the constraints here (representing the budget) and concentrate on the cost and NPV uncertainties assuming that the budget is given. If there was uncertainty on the RHS, robust optimization would require the manager to assign the RHS its worst-case value. Therefore, budget uncertainty – if it is present in the formulation – would be addressed in the same manner as Koc et al. (2009), using scenarios for different levels of the budget. In what follows, we assume that there is no budget uncertainty, both for our approach and our implementation of the Koc et al. (2009) approach.

- Finally, we do not make the assumptions on the behavior of the uncertain parameters that are made in Koc et al. (2009): we do not assume that the cost and NPV are perfectly correlated; furthermore, we do not assume that the projects in the same risk groups are perfectly correlated. We feel that our setting is more representative of real-life industry situations.

3 Improved Stochastic Formulation

3.1 Model

The notation and formulation of the prioritization model in Koc et al. (2009) are:

Indices and sets:

$i, i' \in I$	candidate projects
$p \in P$	priorities; $P = \{1, 2, \ldots |I|\}$
$t \in T$	time periods (years)
$\omega \in \Omega$	scenarios

Data:

a_i^ω	net present value of project i under scenario ω
b_t^ω	available budget in period t under scenario ω
c_{it}^ω	cost of project i in period t under scenario ω
q^ω	probability of scenario ω

Decision variables (binary):

x_i^ω	1 if project i is selected under scenario ω, 0 otherwise
$y_{i,i'}$	1 if project i has higher priority than i', 0 otherwise
z_{ip}	1 if project i is assigned priority level p, 0 otherwise

Formulation:

$$
\begin{aligned}
\max_{x,y,z} \quad & \sum_{\omega\in\Omega} q^{\omega} \sum_{i\in I} a_i^{\omega}\, x_i^{\omega} & & (a) \\
\text{s.t.} \quad & \sum_{i\in I} c_{i,t}^{\omega}\, x_i^{\omega} \le b_t^{\omega}, & t\in T, \omega\in\Omega \quad & (b) \\
& \sum_{i\in I} z_{i,p} = 1, & p\in P \quad & (c) \\
& \sum_{p\in P} z_{i,p} = 1, & i\in I \quad & (d) \\
& |P| y_{i,i'} \ge \sum_{p\in P} (|P|-p)(z_{i,p} - z_{i',p}), & i\neq i', i,i'\in I \quad & (e) \\
& y_{i,i'} + y_{i',i} = 1, & i< i', i,i'\in I \quad & (f) \\
& x_i^{\omega} \ge x_{i'}^{\omega} + y_{i,i'} - 1, & \omega\in\Omega, i\neq i', i,i'\in I \quad & (g) \\
& x_i^{\omega} \in \{0,1\} & i\in I, \omega\in\Omega \quad & (h) \\
& y_{i,i'} \in \{0,1\} & i\neq i', i,i'\in I \quad & (i) \\
& z_{i,p} \in \{0,1\} & i\in I, p\in P \quad & (j)
\end{aligned}
\tag{4}
$$

This formulation can be explained as follows:

Objective (a) The decision maker maximizes the expected NPV.

Constraints (b) The total cost cannot exceed the budget, in any given scenario.

Constraints (c) Each priority rank can only be assigned to one project.

Constraints (d) Each project can only be assigned to one priority rank.

Constraints (e) For any pair of projects (i, i'), if i' is assigned a lower priority than i then i is preferred to i'.

Constraints (f) For any pair of projects (i, i'), either i is preferred to i' or i' is preferred to i.

Constraints (g) For any pair of projects (i, i') and any scenario ω, if i' is selected in scenario ω and i is preferred to i', then i is selected in scenario ω as well.

Constraints (h),(i),(j) Decision variables are binary.

An important observation we made when we first attempted to implement the approach in Koc et al. (2009) is that the decision variables $z_{i,p}$, which provide the priority level of the projects, are not necessary in the formulation. The essential knowledge – the pairwise comparisons of the projects' priorities – lies in the variable $y_{i,i'}$. This makes constraints (c),(d),(e) unnecessary.

Note that the $|P|$ used in constraints (e) is nothing but a big-M constraint, which impairs the tightness of LP relaxations and increases the run times.

Further, we suggest the following changes for constraints (f) and (g):

$$\tilde{y}_{i,i'} \geq x_i^\omega - x_{i'}^\omega, \quad \forall \omega \in \Omega,\, i, i' \in I : i < i' \tag{5}$$

$$1 - \tilde{y}_{i,i'} \geq x_{i'}^\omega - x_i^\omega, \quad \forall \omega \in \Omega,\, i, i' \in I : i < i' \tag{6}$$

Note that we replace the variable $y_{i,i'}$ with $\tilde{y}_{i,i'}$, which is only defined for $i, i' \in I : i < i'$. We can set $\tilde{y}_{i,i'} = 0$ if $i \geq i'$ and drop them. If $x_i^\omega = 1$ and $x_{i'}^\omega = 0$, Eq. (5) forces $\tilde{y}_{i,i'} = 1$. Then Eq. (6) forces i to be preferred to i' for all ω. When the model is solved, the optimal $\tilde{y}_{i,i'}$ give us a two-by-two comparison of all variables. We can then build the priority list based on this information, because ($i \succ j$ and $j \succ k$) implies $i \succ k$. This statement is proved by noting that, by definition, $i \succ j$ means that if $x_i^\omega = 0$ in some scenario ω then $x_j^\omega = 0$ for the same scenario ω and there is at least one scenario ω for which $x_i^\omega = 1$ and $x_j^\omega = 0$. Similarly, $j \succ k$ means that if $x_j^\omega = 0$ in some scenario ω then $x_k^\omega = 0$ for the same scenario ω and there is at least one scenario ω for which $x_j^\omega = 1$ and $x_k^\omega = 0$. Combining the two statements yields the result immediately.

The new set of constraints described above is tight as the constraints do not require a big-M constraint. (In fact, $\tilde{y}_{i,i'}$ can even be relaxed to be in [0,1] for all pairs of projects.) Instead of Model (4), we can then solve the following problem as a more computationally efficient stochastic programming problem:

$$\begin{aligned}
\max_{x,y} \quad & \sum_{\omega \in \Omega} q^\omega \sum_{i \in I} a_i^\omega \, x_i^\omega \\
\text{s.t.} \quad & \sum_{i \in I} c_{i,t}^\omega \, x_i^\omega \leq b_t^\omega, && t \in T, \omega \in \Omega \\
& y_{i,i'} \geq x_i^\omega - x_{i'}^\omega, && \omega \in \Omega, i < i', i, i' \in I \\
& 1 - y_{i,i'} \geq x_{i'}^\omega - x_i^\omega, && \omega \in \Omega, i < i', i, i' \in I \\
& x_i^\omega \in \{0,1\} && i \in I, \omega \in \Omega \\
& y_{i,i'} \in \{0,1\} && i \neq i', i, i' \in I
\end{aligned} \tag{7}$$

3.2 Implementation

We solve Koc et al. (2009)'s model (4) and our suggested prioritizing model (7) using ILOG CPLEX version 12.1 for the full-size problem data given in Koc et al. (2009). Both problems hit the time limit, which was set to 100,000 seconds. However, our suggested problem solution was at 0.13% of optimality, while Koc et al. (2009)'s solution was at 37.20% of optimality. Koc et al. (2009) does not provide run time statistics but only state the model was ultimately solved within 1% of optimality. Figure 1 shows the optimality gap and objective function values for both problems with respect to simplex iterations. As it is seen from the left panel of Figure 1, the revised stochastic problem quickly reduces the optimality gap to within 1% of optimality. Although Benchmark Problem (4)'s initial lower bound (around 47) is larger (which is better in a maximization problem), the computer finds it in more iterations than what is needed for the revised model to improve the lower bound to above 60, as seen on the right panel of Figure 1.

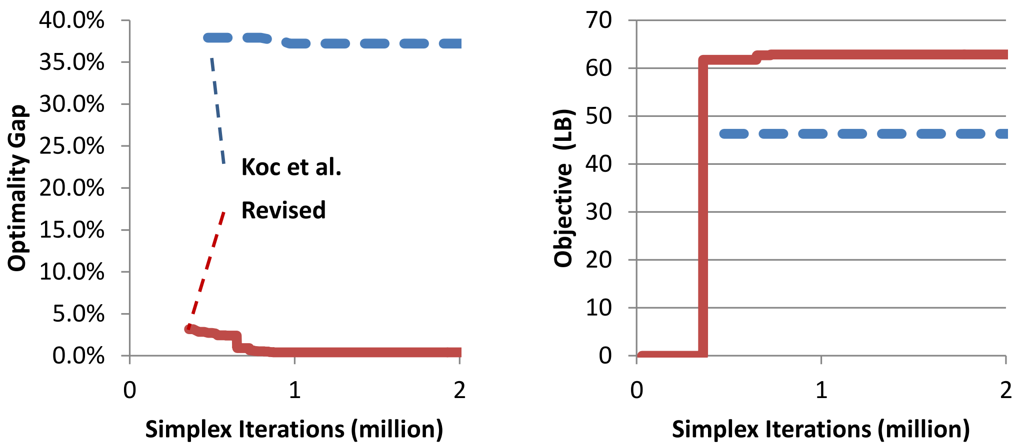

Figure 1: Comparison of Koc et al. (2009)'s model with its revised model (7)

Solving Model (7) gives us the y variables, from which we derive a priority list for the projects in seconds. We compare the two models for two subproblems provided in Koc et al. (2009), when the number of projects is 10 and 15, respectively. For both subproblems we get the same objective function value and the same x^ω in all scenarios. There were some differences in the priority lists of the two models. The fact that there were some pairwise reversed priorities can be explained by the existence of ties between projects, which the computer breaks arbitrarily.

4 The Multi-Range Robust Optimization Model

4.1 High-level modeling

The stochastic programming problem given in Koc et al. (2009) does not provide any optimal solution within a reasonable time frame. While our stochastic model (7) performs significantly better than the benchmark, we feel that the run times still raise issues in terms of large-scale tractability of the stochastic approach. Therefore, in this section, we derive the multi-range robust counterpart of Problem (7). The formulation will be solved in the next section to demonstrate the potential of robust optimization in terms of solution time and quality.

The approach proposed in Author1 and Author2 (2010) enables us to incorporate all the possible values that uncertain parameters can take in the optimization problem, and thus addresses the limitations of the traditional robust optimization framework. Specifically here, an uncertain parameter will be allowed to take any of the pessimistic, most likely or optimistic values. We have two uncertainty ranges for each uncertain NPV and cost parameter: *low* and *high*. Figure 2 summarizes how we construct our low and high uncertainty ranges for the NPV parameters.

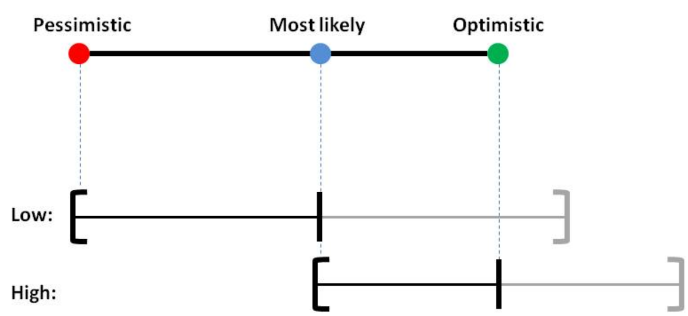

Figure 2: Construction of low and high ranges for the uncertain NPV parameters

For cost parameters, the place of optimistic and pessimistic values will be switched, so that the optimistic value for a cost will be the worst-case value of the low range, as shown in Figure 3.

The intervals are defined by using the fact that at optimality, the uncertain parameters in the robust optimization approach with two ranges will take one of four possible values:

1. The nominal value of the low range,

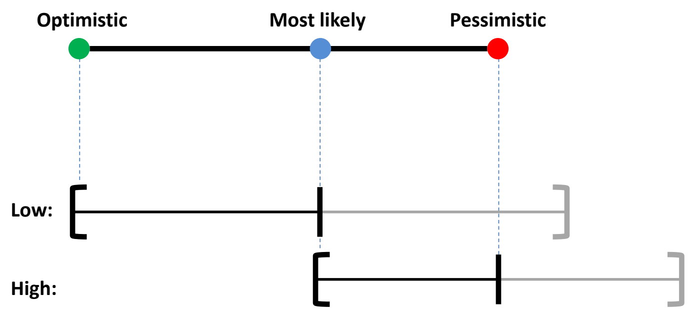

Figure 3: Construction of low and high ranges for the uncertain cost parameters

2. The nominal value of the high range,
3. The worst-case value of the low range,
4. The worst-case value of the high range.

Because we only want to consider three values, we will define the uncertainty intervals so that the nominal value of the low range coincides with the worst-case value of the high range. Again, it is not possible in traditional one-range robust optimization to consider three possible values for the data. With the help of multi-range robust optimization approach, we are able to construct the uncertainty sets such that we can incorporate these multiple values and yet obtain a robust solution without having to consider many scenarios.

Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be the uncertainty sets for NPV factors and cost factors, respectively. The robust problem that we are going to solve has the structure of Problem (1) below:

$$\begin{aligned} \max_{x} \quad & \min_{\mathbf{npv} \in \mathcal{P}_1} \mathbf{npv}\, \mathbf{x} \\ \text{s.t.} \quad & \max_{\mathbf{c} \in \mathcal{P}_2} \mathbf{c}\, \mathbf{x} \leq \mathbf{B} \\ & \mathbf{x} \in \{0,1\}^n \end{aligned} \tag{8}$$

4.2 Inner optimization problems

Inner minimization problem for NPVs.

We assign separate budgets of uncertainty for low-risk projects (with superscript $\mathcal{L}$) and medium-risk projects (with superscript $\mathcal{M}$) because projects in those groups have different probabilities of attaining their pessimistic, most likely and optimistic values. Superscript or subscript l denotes low-range coefficients, while superscript or subscript h denotes high-range coefficients. The rest of the notation is identical to that in Section 2.

$$
\begin{aligned}
\min_{u^l,u^h,y} \quad & \sum_{i=1}^{n} x_i \left[\overline{NPV}_i^l\, y_i^l - \widehat{NPV}_i^l\, u_i^l + \overline{NPV}_i^h\, y_i^h - \widehat{NPV}_i^h\, u_i^h\right] & \\
\text{s.t.} \quad & u_i^l \le y_i^l, & \forall i \in I, \\
& u_i^h \le y_i^h, & \forall i \in I, \\
& y_i^l + y_i^h = 1, & \forall i \in I, \\
& \sum_{i \in \mathcal{L}} y_i^l \le \Gamma_l^{\mathcal{L}}, & \\
& \sum_{i \in \mathcal{M}} y_i^l \le \Gamma_l^{\mathcal{M}}, & \\
& \sum_{i \in \mathcal{L}} (u_i^l + u_i^h) \le \Gamma^{\mathcal{L}}, & \\
& \sum_{i \in \mathcal{M}} (u_i^l + u_i^h) \le \Gamma^{\mathcal{M}}, & \\
& y_i^j \in \{0,1\}, & \forall i \in I,\ \forall j \in \{l,h\}, \\
& u_i^l, u_i^h \ge 0, & \forall i \in I.
\end{aligned} \tag{9}
$$

Inner maximization problems for cost factors.

In year t, we have:

$$
\begin{aligned}
\max_{u^l,u^h,y} \quad & \sum_{i=1}^{n} x_i \left[\overline{c}_{i,t}^l\, y_i^l - \widehat{c}_{i,t}^l\, u_{i,t}^l + \overline{c}_{i,t}^h\, y_i^h - \widehat{c}_{i,t}^h\, u_{i,t}^h\right] & \\
\text{s.t.} \quad & u_{i,t}^l \le y_i^l, & \forall i \in I, \\
& u_{i,t}^h \le y_i^h, & \forall i \in I, \\
& y_i^l + y_i^h = 1, & \forall i \in I, \\
& \sum_{i \in \mathcal{L}} y_i^l \ge \Gamma_l^{\mathcal{L}}, & \\
& \sum_{i \in \mathcal{M}} y_i^l \ge \Gamma_l^{\mathcal{M}}, & \\
& \sum_{i \in \mathcal{L}} (u_{i,t}^l + u_{i,t}^h) \ge \Gamma^{\mathcal{L}}, & \\
& \sum_{i \in \mathcal{M}} (u_{i,t}^l + u_{i,t}^h) \ge \Gamma^{\mathcal{M}}, & \\
& y_i^j \in \{0,1\}, & \forall i \in I,\ \forall j \in \{l,h\}, \\
& u_{i,t}^l, u_{i,t}^h \ge 0, & \forall i \in I.
\end{aligned} \tag{10}
$$

Note that we have greater-than-or-equal-to constraints for the uncertainty budgets because the inner problem is now a maximization problem. Thus, we will have exactly $\Gamma_l^{\mathcal{L}}$ and $\Gamma_l^{\mathcal{M}}$ values in the low range among low-risk and medium-risk projects, respectively. Similarly, exactly $\Gamma^{\mathcal{L}}$ low-risk and $\Gamma^{\mathcal{M}}$ medium-risk projects will take the worst case values in the range they fall into.

From Author1 and Author2 (2010), we know that the constraint sets of Problems (9) and (10) are totally unimodular. Therefore, we can relax the integrality of the y variables and still obtain an integer optimal solution, given that the right-hand-sides of the constraints are integer. This allows us to use strong duality and convert Problem (8) into one large maximization problem.

4.3 The formulation

The **objective function** of our robust optimization problem comes from the objective function of the dual problem of Problem (9):

$$\max \quad \sum_{i \in I} p_i - \sum_{i \in I} \left(z_i^l + z_i^h \right) - \Gamma_l^{\mathcal{L}} \gamma_l^{\mathcal{L}} - \Gamma_l^{\mathcal{M}} \gamma_l^{\mathcal{M}} - \Gamma^{\mathcal{L}} \gamma^{\mathcal{L}} - \Gamma^{\mathcal{M}} \gamma^{\mathcal{M}}$$

The constraints of the dual problem are added to the constraint set of our robust optimization problem. Dual constraints associated with variables y_i^l and y_i^h for low-risk and medium-risk projects are:

$$\begin{aligned}
p_i^l + p_i - \gamma_l^{\mathcal{L}} - z_i^l &\leq \overline{NPV}_i^l x_i \quad && i \in \mathcal{L} \\
p_i^h + p_i - z_i^h &\leq \overline{NPV}_i^h x_i && i \in \mathcal{L} \\
p_i^l + p_i - \gamma_l^{\mathcal{M}} - z_i^l &\leq \overline{NPV}_i^l x_i && i \in \mathcal{M} \\
p_i^h + p_i - z_i^h &\leq \overline{NPV}_i^h x_i && i \in \mathcal{M}
\end{aligned}$$

Similarly, dual constraints associated with variables u_i^l and u_i^h for low-risk and medium-risk projects are:

$$\begin{aligned}
p_i^l + \gamma^{\mathcal{L}} &\leq \widehat{NPV}_i^l x_i \quad && i \in \mathcal{L} \\
p_i^h + \gamma^{\mathcal{L}} &\leq \widehat{NPV}_i^h x_i && i \in \mathcal{L} \\
p_i^l + \gamma^{\mathcal{M}} &\leq \widehat{NPV}_i^l x_i && i \in \mathcal{M} \\
p_i^h + \gamma^{\mathcal{M}} &\leq \widehat{NPV}_i^h x_i && i \in \mathcal{M}
\end{aligned}$$

For the **uncertain cost parameters**, we have a maximization problem in the constraint set of Problem (8) but invoke strong duality and thus insert the dual problem of Problem (10) into our robust counterpart problem. The objective function of the dual of Problem (10) will represent our new budget constraint:

$$\sum_{i \in I} cp_{i,t} + \sum_{i \in I} \left(cz^l_{i,t} + cz^h_{i,t} \right) + c\Gamma^{\mathcal{L}}_l \, c\gamma^{\mathcal{L}}_{l,t} + c\Gamma^{\mathcal{M}}_l \, c\gamma^{\mathcal{M}}_{l,t} + c\Gamma^{\mathcal{L}} \, c\gamma^{\mathcal{L}}_t + c\Gamma^{\mathcal{M}} \, c\gamma^{\mathcal{M}}_t \leq B(t), \quad t \in T$$

Then, we will add the dual constraints to the robust counterpart problem. The dual constraints corresponding to y^l_i and y^h_i for low-risk and medium-risk projects in Problem (10) are:

$$\begin{aligned}
-cp^l_{i,t} + cp_{i,t} + c\gamma^{\mathcal{L}}_{l,t} + cz^l_{i,t} &\geq \overline{c}^l_{i,t} \, x_i \quad && i \in \mathcal{L}, t \in T \\
-cp^h_{i,t} + cp_{i,t} + cz^h_{i,t} &\geq \overline{c}^h_{i,t} \, x_i && i \in \mathcal{L}, t \in T \\
-cp^l_{i,t} + cp_{i,t} + c\gamma^{\mathcal{M}}_{l,t} + cz^l_{i,t} &\geq \overline{c}^l_{i,t} \, x_i && i \in \mathcal{M}, t \in T \\
-cp^h_{i,t} + cp_{i,t} + cz^h_{i,t} &\geq \overline{c}^h_{i,t} \, x_i && i \in \mathcal{M}, t \in T
\end{aligned}$$

Similarly, the dual constraints associated with variables u^l_i and u^h_i in Problem (10) for low-risk and medium-risk projects are:

$$\begin{aligned}
cp^l_{i,t} + c\gamma^{\mathcal{L}}_t &\leq -\widehat{c}^l_{i,t} \, x_i \quad && i \in \mathcal{L}, t \in T \\
cp^h_{i,t} + c\gamma^{\mathcal{L}}_t &\leq -\widehat{c}^h_{i,t} \, x_i && i \in \mathcal{L}, t \in T \\
cp^l_{i,t} + c\gamma^{\mathcal{M}}_t &\geq -\widehat{c}^l_{i,t} \, x_i && i \in \mathcal{M}, t \in T \\
cp^h_{i,t} + c\gamma^{\mathcal{M}}_t &\geq -\widehat{c}^h_{i,t} \, x_i && i \in \mathcal{M}, t \in T
\end{aligned}$$

In addition to these constraints, we have the constraints that were originally in the problem before reformulation and the sign constraints:

$$\begin{aligned}
& x_i \in \{0,1\}^n, && i \in I, \\
& p^l_i, p^h_i, cp^l_{i,t}, cp^h_{i,t} \geq 0, && i \in I, t \in T, \\
& z^l_i, z^h_i, cz^l_{i,t}, cz^h_{i,t} \geq 0, && i \in I, t \in T, \\
& \gamma^{\mathcal{L}}_l, \gamma^{\mathcal{M}}_l, \gamma^{\mathcal{L}}_l, \gamma^{\mathcal{M}}_l \geq 0, && i \in I, t \in T,
\end{aligned}$$

The **complete formulation** is given by:

$$
\begin{aligned}
\max \quad & \sum_{i\in I} p_i - \sum_{i \in I} \left(z_i^l + z_i^h \right) - \Gamma_l^{\mathcal{L}} \gamma_l^{\mathcal{L}} - \Gamma_l^{\mathcal{M}} \gamma_l^{\mathcal{M}} - \Gamma^{\mathcal{L}} \gamma^{\mathcal{L}} - \Gamma^{\mathcal{M}} \gamma^{\mathcal{M}} \\
\text{s.t.} \quad & \max_{\mathbf{c} \in \mathcal{P}_2} \mathbf{c}'\mathbf{x} \leq \mathbf{B} \\
& p_i^l + p_i - \gamma_l^{\mathcal{L}} - z_i^l \leq \overline{NPV}_i^l x_i & i \in \mathcal{L} \\
& p_i^h + p_i - z_i^h \leq \overline{NPV}_i^l x_i & i \in \mathcal{L} \\
& p_i^l + \gamma^{\mathcal{L}} \leq \widehat{NPV}_i^l x_i & i \in \mathcal{L} \\
& p_i^h + \gamma^{\mathcal{L}} \leq \widehat{NPV}_i^h x_i & i \in \mathcal{L} \qquad (11)\\
& p_i^l + p_i - \gamma_l^{\mathcal{M}} - z_i^l \leq \overline{NPV}_i^l x_i & i \in \mathcal{M} \\
& p_i^h + p_i - z_i^h \leq \overline{NPV}_i^l x_i & i \in \mathcal{M} \\
& p_i^l + \gamma^{\mathcal{M}} \leq \widehat{NPV}_i^l x_i & i \in \mathcal{M} \\
& p_i^h + \gamma^{\mathcal{M}} \leq \widehat{NPV}_i^h x_i & i \in \mathcal{M} \\
& \mathbf{x} \in \{0,1\}^n.
\end{aligned}
$$

Note that we no longer have any y binary variable establishing pairwise priorities because determining an appropriate priority order is straightforward once we have obtained the optimal solution: any order that ranks the selected ones above the non-selected ones will work. Our robust model is a deterministic model and finds a single portfolio unlike the stochastic programming model, which finds separate portfolios for different scenarios but a single ordering for all. Imposing a single priority order in that problem is, therefore, meaningful in the stochastic programming problem but redundant in the robust optimization one, since a priority can be inferred from the optimal solution. Therefore, for the multi-range problem we solve only the knapsack problem without prioritizing projects (Model (8)) and obtain the priority list through post-processing.

5 Numerical Study

5.1 Setup

We follow the setup described in Koc et al. (2009) and have 26 low-risk projects and 15 medium-risk projects. In the stochastic programming approach, the cost and NPV of a low-risk project are assigned the pessimistic value with probability $\frac{1}{6}$, the optimistic value with probability $\frac{1}{6}$, and the most likely value with probability $\frac{4}{6}$. For medium-risk projects these three probabilities become

$\frac{2}{6}$, $\frac{1}{6}$ and $\frac{3}{6}$, respectively. We use these probabilities to determine the budgets of uncertainty. On the average 4 or 5 projects out of 26 low-risk projects ($\frac{26}{6} = 4.33$) would take the pessimistic values. Similarly, 4 or 5 of them would take the optimistic values. 5 out of 15 medium-risk projects ($\frac{15 \cdot 2}{6} = 5$) would take the pessimistic values, 2 or 3 of them ($\frac{15}{6} = 2.5$) would take the optimistic values. Because the ranges are constructed so that the nominal value of the low range and the worst-case value of the high range coincide, we have one degree of freedom in setting the parameters.

We have 10 possible budgets: from \$2.5M to \$7M, in increments of \$0.5M. For each of these possible budgets, we solve Model (8) for three model settings, which can be seen on Table 1. The nominal model is the model where all NPV and cost components will take the *most likely* values.

Parameters		Model Setting		
Project	Model	Nominal	Robust1	Robust2
NPV	$\Gamma_l^{\mathcal{L}}$	26	8	13
	$\Gamma_l^{\mathcal{M}}$	15	5	7
	$\Gamma^{\mathcal{L}}$	0	18	13
	$\Gamma^{\mathcal{M}}$	0	13	11
Cost	$\Gamma_l^{\mathcal{L}}$	26	14	13
	$\Gamma_l^{\mathcal{M}}$	15	8	8
	$\Gamma^{\mathcal{L}}$	0	8	13
	$\Gamma^{\mathcal{M}}$	0	4	4

Table 1: Uncertainty budget combinations.

In the *Robust 1* combination, we have 8 low-risk projects in the low range, and the remaining 18 low-risk projects will be in the high range. We want 4 low-risk project to take the pessimistic values, in line with the probabilities mentioned above. Then, 4 of 8 projects in low range should be able to deviate from the nominal value. Also, 4 projects might take the optimistic values. It suffices to assume that 18-4 = 14 projects can deviate from the optimistic value (nominal value of the high range) and be at the lowest value of the high range, which is also the *most likely* value. Thus, we set $\Gamma^{\mathcal{L}} = 4 + 14 = 18$. Other values and the value in *Robust 2* are also assigned in a similar manner, recalling that we have one degree of freedom to set the budget parameters.

5.2 Results

Our focus is to determine whether the multi-range robust optimization approach has potential as a computational alternative to stochastic programming for this real-life problem. Figure 4 compares the objective function values for the Nominal and Robust 1 settings. The rightmost part of the figure shows the expected value of the objective function computed over all budget scenarios. The circle in this rightmost column is the stochastic-model solution reported by Koc et al. (2009). The triangle indicates the stochastic-model solution (4) when the solver hit the time limit at 100,000 seconds. We see that our expected objective function value is very close to the one given by the stochastic model. We were able to incorporate the expert knowledge given by the company and obtain robust solutions which were not very conservative. Moreover, the robust optimization problem is solved to optimality in less than a second, while we could not get the optimal solution of the stochastic case in a reasonable time frame.

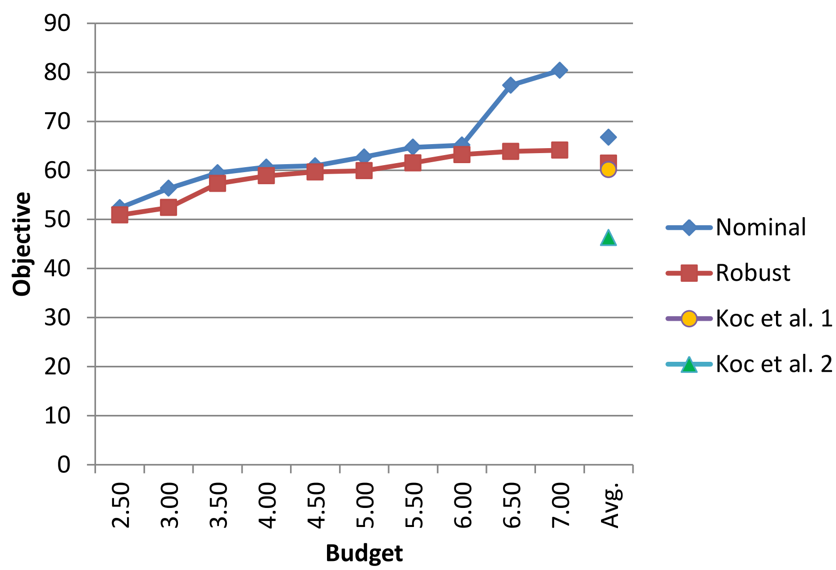

Figure 4: Objective function values of nominal and robust solutions for different budget scenarios.

Tables 2, 3 and 4 display the objective function and model statistics for each budget values for three model settings: Nominal, Robust 1 and Robust 2, respectively. The expected NPV is also given for each setting at the last row of the tables. Expected values are calculated using the probabilities of each budget and objective of that budget. We see that the nominal setting has the highest expected return, as anticipated. The Robust 1 and Robust 2 settings yield an expected NPV of 61.46% and 61.33%, respectively. Koc et al. (2009) reports an expected NPV of 60.18%. We observe that our model solves to optimality in less than a second. Koc et al.

Budget	Objective	Time (sec)	Iterations	Nodes
2.5	52.38	0.51	699	50
3.0	56.34	0.39	618	15
3.5	59.51	0.30	604	12
4.0	60.66	0.23	645	16
4.5	60.95	0.31	653	10
5.0	62.73	0.27	630	29
5.5	64.71	0.20	613	12
6.0	65.16	0.25	631	21
6.5	77.34	0.37	639	20
7.0	80.38	0.22	578	5
Average	**66.75**			

Table 2: Model results for *Nominal* uncertainty budget combinations

Budget	Objective	Time (sec)	Iterations	Nodes
2.5	50.91	0.17	538	42
3.0	52.43	0.38	708	67
3.5	57.31	0.21	464	11
4.0	58.89	0.22	435	12
4.5	59.69	0.11	431	8
5.0	59.92	0.19	436	9
5.5	61.53	0.17	460	31
6.0	63.22	0.22	399	6
6.5	63.88	0.21	382	6
7.0	64.13	0.20	412	16
Average	**61.46**			

Table 3: Model results for *Robust 1* uncertainty budget combinations.

(2009) does not report their model's statistics. Furthermore, if we do add constraints, such as constraints that create a priority list as part of the optimization problem, our solution time is only of the order of 1-2 seconds, as shown in Table 5.

6 Conclusions

We have shown how to implement multi-range robust optimization as a tractable alternative to stochastic programming, by selecting the budgets of uncertainty appropriately to match (the rounded values of) the expected number of times that the uncertain parameters will take their optimistic, most likely, pessimistic values. We have also shown how to improve solution time of the

Budget	Objective	Time (sec)	Iterations	Nodes
2.5	50.82	0.27	636	44
3.0	52.31	0.40	1050	108
3.5	57.19	0.41	681	13
4.0	58.76	0.18	547	21
4.5	59.56	0.25	452	8
5.0	59.78	0.26	477	9
5.5	61.40	0.26	538	37
6.0	63.09	0.44	634	10
6.5	63.73	0.26	377	5
7.0	63.97	0.21	485	19
Average	**61.33**			

Table 4: Model results for *Robust 2* uncertainty budget combinations.

Budget	Objective	Time (sec)	Iterations	Nodes
2.5	50.82	1.16	1725	30
3.0	52.31	1.48	3354	81
3.5	57.19	0.75	1379	15
4.0	58.76	0.75	1667	17
4.5	59.56	0.54	1582	11
5.0	59.78	0.53	1367	8
5.5	61.40	2.42	2449	43
6.0	63.09	1.04	1472	19
6.5	63.73	1.11	1245	17
7.0	63.97	0.62	1089	6
Average	**61.33**			

Table 5: Model with prioritization results for *Robust 2* uncertainty budget combinations.

stochastic programming approach by using post-processing. Numerical results are encouraging.

References

Author1 and Author2. Information withheld for double-blind review process. Technical report, 2010.

A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.

A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25:1–13, 1999.

A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88:411–424, 2000.

A. Ben-Tal, L. El-Ghaoui, and A. Nemirovski. *Robust optimization*. Princeton Series in Applied Mathematics. Princeton University Press, Princeton, NJ, 2009.

D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming B*, 98:49–71, 2003.

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

D. Bertsimas, D. Pachamanova, and M. Sim. Robust linear optimization under general norms. *Operations Research Letters*, 32:510–516, 2004.

D. Bertsimas, D. Brown, and C. Caramanis. Theory and applications of robust optimization. Technical report, University of Texas at Austin, Austin, TX, 2007.

D. Bienstock. Histogram models for robust portfolio optimization. *Journal of Computational Finance*, 11(1):1–64, 2007.

G. B. Dantzig. Linear programming under uncertainty. *Management Science*, 1(3-4):197–206, 1955.

L. El-Ghaoui and H. Lebret. Robust solutions to least-square problems to uncertain data. *SIAM Journal on Matrix Analysis and Applications*, 18(4):1035–1064, 1997.

L. El-Ghaoui, F. Oustry, and H. Lebret. Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, 9:33–52, 1998.

A. Koc, D. P. Morton, E. Popova, E. K. Stephen M. Hess, and D. Richards. Prioritizing project selection. *The Engineering Economist*, 54:267–297, 2009.

A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21:1154–1157, 1973.