Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

EVALUATING TWO-RANGE ROBUST OPTIMIZATION FOR PROJECT SELECTION

Ruken Düzgün

Marriott International
10400 Fernwood Rd
Bethesda, MD 20817, USA

Aurélie Thiele

Lehigh University
200 W Packer Ave.
Bethlehem, PA 18015, USA

ABSTRACT

This paper investigates empirically two-range robust optimization (2R-RO) as an alternative to stochastic programming in terms of computational time and solution quality. We consider a number of possible projects with anticipated costs and cash flows, and an investment decision to be made under budget limitations. In 2R-RO, each uncertain parameter is allowed to take values from more than one uncertainty range and the number of parameters that fall within each range is bounded by a budget of uncertainty. The stochastic description of uncertainty involves three values (high, medium and low) for each ambiguous parameter. We set up the 2R-RO model so that the possible values taken by the uncertain parameters match the three scenarios in the stochastic programming approach and test both in simulations. While the stochastic programming (SP) approach takes about a day to solve, the robust optimization (RO) approach solves the same project selection problem in seconds.

1 INTRODUCTION

Our goal in this paper is to determine whether a robust optimization approach we have developed, called the multi-range robust optimization approach, has potential as a computational alternative to stochastic programming for a real-life problem of project selection under uncertainty described below. Data in real-life applications is often not known precisely at the time when the manager must make a decision. Here, we have a number of possible projects with anticipated costs and cash flows to choose from, and must make a decision under budget limitations. Cost and net present values of the projects are uncertain. Specifically, our goal 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. 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), Ben-Tal and Nemirovski (1999), Ben-Tal and Nemirovski (2000), El-Ghaoui and Lebret (1997) and El-Ghaoui, Oustry, and Lebret (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), Bertsimas and Sim (2004) and Bertsimas, Pachamanova, and Sim (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, Brown, and Caramanis (2011) and Gabrel, Murat, and Thiele (2014) for comprehensive reviews of robust optimization and Ben-Tal, El-Ghaoui, and Nemirovski (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 in a real-life setting the potential, both in terms of computational time and solution quality, of an approach based on a concept called *multi-range robust optimization* (Duzgun and Thiele 2015, Duzgun 2012), compared to the traditional stochastic programming approach, using simulation. This approach was subsequently applied to uncertainty in innovation adoption parameters for new product launches, when the number of adopters over time is modeled using a Bass diffusion model (Cetinkaya and Thiele 2014). The specific example we consider here is project selection and prioritization. In contrast with Duzgun and Thiele (2015), 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 study is to present multi-range robust optimization as a tractable alternative to stochastic programming for industry practitioners, when the budgets of uncertainty are set appropriately based on the probabilities of the stochastic programming model. (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 in Duzgun and Thiele (2015), which relies on total unimodularity.) A secondary contribution of this study is that we provide an efficient manner to compute the project priority list, thus substantially reducing computation times and providing valuable information to practitioners. In our large-scale example, the stochastic programming approach does not solve within the allotted time while our robust optimization approach solves its model to optimality within seconds. We hope that this motivates industry leaders to consider multi-range robust optimization as a valuable tool to address decision-making under uncertainty.

Our goal in this study is to apply the robust optimization approach with multiple ranges to a project selection and prioritization problem where cost and net present values (NPV) of the projects are uncertain. 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. We compare our results to a benchmark two-stage stochastic programming problem and maximizes the expected NPV of the selected costs calculated over predefined scenarios.

The rest of the paper is structured as follows. In Section 2, we describe the stochastic programming formulation (benchmark) for this problem. Section 3 presents the robust optimization formulation in the proposed setting, while Section 4 provides the details of the numerical implementation. Section 5 contains concluding remarks.

2 STOCHASTIC PROGRAMMING FORMULATION

First, we model our problem in the stochastic programming framework.

2.1 Model

We will use the following notation for the stochastic model:
Indices and sets:

$i, i' \in I$	candidate projects
$p \in P$	priorities; $P = \{1, 2, \ldots \lvert I \rvert\}$
$t \in T$	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

The stochastic formulation models the following: (a) The decision maker maximizes the expected NPV. (b) The total cost cannot exceed the budget, in any given scenario. (c) For any pair of projects (i, i'), if i does not have priority over i', then i' is selected in all the scenarios where i is selected. (d) For any pair of projects (i, i'), if i has priority over i', then i is selected in all the scenarios where i' is selected. (e) Decision variables are binary.

In mathematical terms, this becomes:
Formulation:

$$
\begin{array}{lll}
\max\limits_{x,y} & \sum\limits_{\omega \in \Omega} q^\omega \sum\limits_{i \in I} a_i^\omega \, x_i^\omega & \\
\text{s.t.} & \sum\limits_{i \in I} c_{i,t}^\omega \, x_i^\omega \leq b_t^\omega, & t \in T, \omega \in \Omega \\
& y_{i,i'} \geq x_i^\omega - x_{i'}^\omega, & \omega \in \Omega, i < i', i, i' \in I \\
& 1 - y_{i,i'} \geq x_{i'}^\omega - x_i^\omega, & \omega \in \Omega, i < i', i, i' \in I \\
& x_i^\omega \in \{0, 1\}, & i \in I, \omega \in \Omega \\
& y_{i,i'} \in \{0, 1\}, & i \neq i', i, i' \in I.
\end{array}
\tag{1}
$$

2.2 Implementation

We solve our stochastic prioritizing model (1) using ILOG CPLEX version 12.1 and realistic data for a portfolio of 41 projects (26 low-risk and 15 medium-risk) over 5 years when each uncertain parameter can take 3 values (high, medium or low). The data set is described in more details in Section 4. When the solver hits the time limit, which was set to 100,000 seconds or slightly over 1 day, our solution is at 0.13% of optimality. Solving Model (1) gives us the y variables, from which we derive a priority list for the projects within seconds.

3 THE MULTI-RANGE ROBUST OPTIMIZATION PROBLEM

3.1 High-level modeling

While our stochastic model (1) performs rather well (was close to optimality when it hit the time limit), the run times raise issues in terms of general large-scale tractability of the stochastic approach. Therefore, in this section, we derive the robust counterpart of Problem (1) based on two-range robust optimization (Duzgun and Thiele 2015). Multi-range robust optimization 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, that the decision-maker cannot incorporate information on the distribution of the uncertain parameters in a description of uncertainty based on a single range for each parameter). In our example, 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 1 summarizes how we construct our low and high uncertainty ranges for the 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 (see Figure 2).

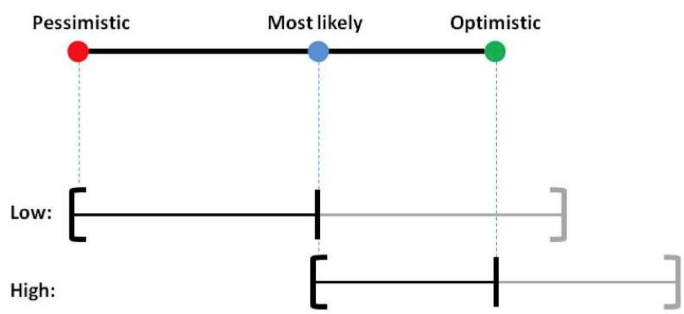

Figure 1: Construction of low and high ranges for the uncertain NPV parameters.

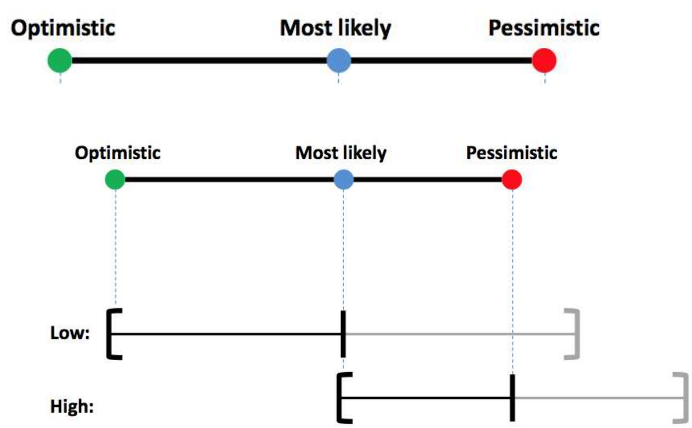

Figure 2: Construction of low and high ranges for the uncertain cost parameters.

The intervals are defined by using the fact that at optimality, the uncertain parameters in the two-range robust optimization approach will take one of four possible values: (i) the nominal value of the low range, (ii) the nominal value of the high range, (iii) the worst-case value of the low range, (iv) the worst-case value of the high range (Duzgun and Thiele 2015).

Because we only want to consider three values to match the stochastic distribution, we 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 following structure:

$$\begin{array}{rl} \max\limits_{x} & \min\limits_{\mathbf{npv}\in\mathcal{P}_1} \mathbf{npv}\,\mathbf{x} \\ \text{s.t.} & \max\limits_{\mathbf{c}\in\mathcal{P}_2} \mathbf{c}\,\mathbf{x} \leq \mathbf{B} \\ & \mathbf{x} \in \{0,1\}^n \end{array} \tag{2}$$

We provide an explicit, tractable reformulation of Problem (2) below.

3.2 Inner optimization problems

Inner minimization problem for NPVs.

The number of coefficients that can fall within a given range is bounded from above by a threshold, called budget of uncertainty. 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. y_i^l, reap. y_i^h, is a binary variable determining whether the NPV coefficient for project i falls within the low, resp. high, range. u_i^l, reps. u_i^h is a binary variable determining whether, if the NPV coefficient for project i falls within the low, reps. high, range, it is equal to its worst-case (lowest) value within that range or not, in which case it is equal to the nominal value. Each parameter is either in the low or high range. The total number of low-risk projects whose uncertain NPV coefficients fall into the low range is bounded from above. A similar constraint holds for the total number of medium-risk projects. (It is not necessary to introduce such constraints for the high range, since this is a minimization problem: the computer will naturally put as few coefficients into the high range as possible.)

The total number of NPV coefficients equal to the lowest value of their range (whichever – low or high – that range is) is also bounded for low-risk projects. A similar constraint holds for medium-risk projects. This leads to the formulation below.

$$\begin{array}{lll} \min\limits_{u^l,u^h,y} & \sum\limits_{i=1}^{n} x_i \left[\overline{NPV}_i^l\, y_i^l - \widehat{NPV}_i^l\, u_i^l + \overline{NPV}_i^h\, y_i^h - \widehat{NPV}_i^h\, u_i^h \right] & \\ \text{s.t.} & u_i^l \leq y_i^l, & \forall i \in I, \\ & u_i^h \leq y_i^h, & \forall i \in I, \\ & y_i^l + y_i^h = 1, & \forall i \in I, \\ & \sum\limits_{i\in\mathcal{L}} y_i^l \leq \Gamma_l^{\mathcal{L}}, & \\ & \sum\limits_{i\in\mathcal{M}} y_i^l \leq \Gamma_l^{\mathcal{M}}, & \\ & \sum\limits_{i\in\mathcal{L}} (u_i^l + u_i^h) \leq \Gamma^{\mathcal{L}}, & \\ & \sum\limits_{i\in\mathcal{M}} (u_i^l + u_i^h) \leq \Gamma^{\mathcal{M}}, & \\ & y_i^j \in \{0,1\}, & \forall i \in I,\, \forall j \in \{l,h\}, \\ & u_i^l, u_i^h \geq 0, & \forall i \in I. \end{array} \tag{3}$$

Inner maximization problems for cost factors.

This formulation is motivated in a similar manner as for the NPV minimization, except that we now have a maximization instead of minimization. In year t, the problem becomes:

$$\begin{array}{lll}
\max\limits_{u^l,u^h,y} & \sum\limits_{i=1}^{n} x_i \left[\overline{c}_{i,t}^{l}\, y_i^l - \widehat{c}_{i,t}^{l}\, u_{i,t}^l + \overline{c}_{i,t}^{h}\, y_i^h - \widehat{c}_{i,t}^{h}\, u_{i,t}^h\right] & \\
\text{s.t.} & u_{i,t}^l \leq y_i^l, & \forall i \in I, \\
& u_{i,t}^h \leq y_i^h, & \forall i \in I, \\
& y_i^l + y_i^h = 1, & \forall i \in I, \\
& \sum\limits_{i\in\mathcal{L}} y_i^l \geq \Gamma_l^{\mathcal{L}}, & \\
& \sum\limits_{i\in\mathcal{M}} y_i^l \geq \Gamma_l^{\mathcal{M}}, & \\
& \sum\limits_{i\in\mathcal{L}} (u_{i,t}^l + u_{i,t}^h) \geq \Gamma^{\mathcal{L}}, & \\
& \sum\limits_{i\in\mathcal{M}} (u_{i,t}^l + u_{i,t}^h) \geq \Gamma^{\mathcal{M}}, & \\
& y_i^j \in \{0,1\}, & \forall i \in I,\, \forall j \in \{l,h\}, \\
& u_{i,t}^l, u_{i,t}^h \geq 0, & \forall i \in I.
\end{array} \tag{4}$$

The constraint sets of Problems (3) and (4) are totally unimodular (Duzgun and Thiele 2015). 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 Problems (3) and (4) into maximization problems, which are then reinjected into Problem (2) to yield one large maximization problem. This tractable reformulation is presented in detail for a generic linear programming problem in Duzgun (2012).

3.3 The formulation

The **objective function** of our robust optimization problem is obtained by injecting the objective function of the dual problem of Problem (3):

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

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:

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

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

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

For the **uncertain cost parameters**, inserting the dual problem of Problem (4) into our robust counterpart problem yields our new budget constraint:

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

Then, we 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 (4) are:

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

Similarly, the dual constraints associated with variables u^l_i and u^h_i in Problem (4) for low-risk and medium-risk projects are:

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

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

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

The **complete formulation** is given by:

$$\begin{aligned}
\max\quad & \sum_{i\in I} p_i - \sum_{i\in I}\left(z^l_i + z^h_i\right) - \Gamma^{\mathcal{L}}_l\, \gamma^{\mathcal{L}}_l - \Gamma^{\mathcal{M}}_l\, \gamma^{\mathcal{M}}_l - \Gamma^{\mathcal{L}}\, \gamma^{\mathcal{L}} - \Gamma^{\mathcal{M}}\, \gamma^{\mathcal{M}} \\
\text{s.t.}\quad & \max_{\mathbf{c}\in\mathcal{P}_2} \mathbf{c}'\mathbf{x} \le \mathbf{B} \\
& p^l_i + p_i - \gamma^{\mathcal{L}}_l - z^l_i \le \overline{NPV}^l_i\, x_i && i\in\mathcal{L}\\
& p^h_i + p_i - z^h_i \le \overline{NPV}^l_i\, x_i && i\in\mathcal{L}\\
& p^l_i + \gamma^{\mathcal{L}} \le \widehat{NPV}^l_i\, x_i && i\in\mathcal{L}\\
& p^h_i + \gamma^{\mathcal{L}} \le \widehat{NPV}^h_i\, x_i && i\in\mathcal{L}\\
& p^l_i + p_i - \gamma^{\mathcal{M}}_l - z^l_i \le \overline{NPV}^l_i\, x_i && i\in\mathcal{M}\\
& p^h_i + p_i - z^h_i \le \overline{NPV}^l_i\, x_i && i\in\mathcal{M}\\
& p^l_i + \gamma^{\mathcal{M}} \le \widehat{NPV}^l_i\, x_i && i\in\mathcal{M}\\
& p^h_i + \gamma^{\mathcal{M}} \le \widehat{NPV}^h_i\, x_i && i\in\mathcal{M}\\
& \mathbf{x}\in\{0,1\}^n.
\end{aligned} \tag{5}$$

Note that we no longer have any y binary variable establishing pairwise priorities because determining an appropriate priority order is straightforward in the robust framework 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 (2)) and obtain the priority list through post-processing.

4 NUMERICAL STUDY

4.1 Setup

We explain the selection of the robust parameters in detail here so that the approach can be replicated easily by the interested reader. The key is to set the robust optimization parameters appropriately based on the stochastic programming parameters, to achieve benefits in terms of solution time without sacrificing solution quality. In the stochastic programming approach, which is due to Koc, Morton, and Popova (2009), the cost and NPV of a low-risk project are assigned the pessimistic value with probability $1/6$, the optimistic value with probability $1/6$, and the most likely value with probability $4/6$. (The probabilities themselves are not important; what is important is to set the robust budgets based on those values.) There are 26 low-risk projects. For medium-risk projects these three probabilities become $1/3$, $1/6$ and $1/2$, respectively. There are 15 medium-risk projects.

We use these probabilities to determine the budgets of uncertainty using the following procedure. On average, 4 or 5 projects out of 26 low-risk projects would take the pessimistic values because $26/6 = 4.33$. Similarly, 4 or 5 would take the optimistic values. 5 out of 15 medium-risk projects would take the pessimistic values ($15/3 = 5$) and 2 or 3 would take the optimistic values ($25/6 = 2.5$) . Recall that, for NPV coefficients, the pessimistic values are the low values of the low range and the optimistic values are the nominal values of the high range, while the most likely values are both the nominal values of the low range or the low values of the high range, which coincide by construction. For this reason (the fact that the most likely value can be achieved in two different ways), we have one degree of freedom in setting the budget parameters. Similar comments hold for cost coefficients.

We consider 10 possible budgets: from \$2.5M to \$7M, in increments of \$0.5M. For each of these possible budgets, we solve Model (2) for three model settings, described in Table 1. The nominal model is the model where all NPV and cost components take the *most likely* values.

Table 1: Uncertainty budget combinations.

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

In the *Robust 1* combination, we have 8 low-risk projects in the low range, and the remaining 18 low-risk projects are in the high range. We want 4 low-risk projects to take the pessimistic values and 4 low-risk projects to take the optimistic values, in line with the probabilities mentioned above. Thus, 18 low-risk projects should have their most likely value, 4 from the low range (equal to the nominal value of that range) and 14 from the high range (equal to the lowest value of that range). Thus, we set the number of parameters equal to the worst case value of the range they fall into to $\Gamma^{\mathcal{L}} = 4 + 14 = 18$. 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.

4.2 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 3 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. 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. Moreover, the robust optimization problem is solved to optimality in less than a second, in sharp contrast with the attained time limit of 100,000 seconds in the stochastic programming case (admittedly with an optimality gap of only 0.13%).

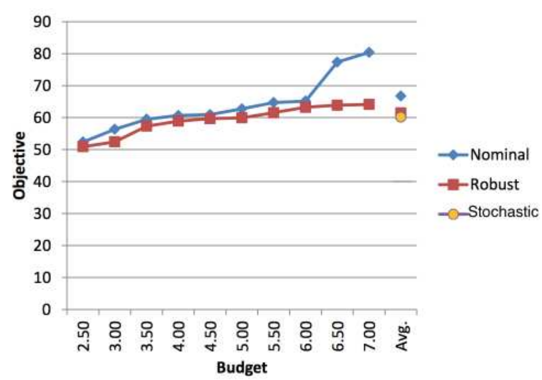

Figure 3: Objective function values of nominal and robust solutions for different budget scenarios.

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 \$61.46M and \$61.33M, respectively, against a nominal NPV of \$66.75M. We observe that our model solves to optimality in less than a second. 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.

5 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. In our example, the two-range robust optimization is solved within seconds while the three-scenario (for each parameter) stochastic counterpart takes about a day. Our methodology can be applied to any

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

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 3: Model results for *Robust 1* 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 4: Model results for *Robust 2* uncertainty budget combinations.

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**			

problem with three scenarios for each uncertain parameter. A limitation of the current approach is that it does not apply to problems where at least one parameter is described using more than three scenarios. Future work includes developing guidelines for setting the parameters of robust optimization with more than two ranges, when the stochastic programming problem has more than three scenarios for each parameter.

Table 5: Model with prioritization 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**			

While these guidelines need to be carefully analyzed and benchmarked, we expect this to be an extension of only moderate difficulty, following the broad framework described here. We also plan to study how the numerical results, especially regarding computational times, scale with respect to problem size. We feel that the approach has potential and should help the practitioner interested in obtaining quality solutions fast for decision-making under uncertainty in large-scale settings.

ACKNOWLEDGMENTS

The authors wish to thank three anonymous reviewers for their helpful comments on this paper.

REFERENCES

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

Ben-Tal, A., and A. Nemirovski. 1998. “Robust Convex Optimization”. *Mathematics of Operations Research* 23 (4): 769–805.

Ben-Tal, A., and A. Nemirovski. 1999. “Robust Solutions of Uncertain Linear Programs”. *Operations Research Letters* 25:1–13.

Ben-Tal, A., and A. Nemirovski. 2000. “Robust Solutions Of Linear Programming Problems Contaminated With Uncertain Data”. *Mathematical Programming* 88:411–424.

Bertsimas, D., D. Brown, and C. Caramanis. 2011. “Theory and Applications of Robust Optimization”. *SIAM Review* 53 (3): 464–501.

Bertsimas, D., D. Pachamanova, and M. Sim. 2004. “Robust Linear Optimization Under General Norms”. *Operations Research Letters* 32:510–516.

Bertsimas, D., and M. Sim. 2003. “Robust Discrete Optimization And Network Flows”. *Mathematical Programming B* 98:49–71.

Bertsimas, D., and M. Sim. 2004. “The Price Of Robustness”. *Operations Research* 52 (1): 35–53.

Bienstock, D. 2007. “Histogram Models For Robust Portfolio Optimization”. *Journal of Computational Finance* 11 (1): 1–64.

Cetinkaya, E., and A. Thiele. 2014. “New Product Launch Decisions with Robust Optimization”. Technical report, Lehigh University.

Dantzig, G. B. 1955. “Linear Programming Under Uncertainty”. *Management Science* 1 (3-4): 197–206.

Duzgun, R. 2012. *Multi-Range Robust Optimization*. Ph. D. thesis, Lehigh University.

Duzgun, R., and A. Thiele. 2015. "Robust Optimization With Multiple Ranges: Theory and Application to Pharmaceutical Project Selection". In *Proceedings of the INFORMS Computing Society Conference*, 103–118. INFORMS.

El-Ghaoui, L., and H. Lebret. 1997. "Robust Solutions To Least-Square Problems To Uncertain Data". *SIAM Journal on Matrix Analysis and Applications* 18 (4): 1035–1064.

El-Ghaoui, L., F. Oustry, and H. Lebret. 1998. "Robust Solutions To Uncertain Semidefinite Programs". *SIAM Journal on Optimization* 9:33–52.

Gabrel, M., C. Murat, and A. Thiele. 2014. "Recent Advances In Robust Optimization: An Overview". *European Journal of Operational Research* 235 (3): 471–483.

Koc, A., D. Morton, and E. Popova. 2009. "Prioritizing Project Selection". *The Engineering Economist* 54:267–297.

Soyster, A. L. 1973. "Convex Programming With Set-Inclusive Constraints And Applications To Inexact Linear Programming". *Operations Research* 21:1154–1157.

AUTHOR BIOGRAPHIES

RUKEN DUZGUN is an Operations Research Analyst at Marriott International in Bethesda, MD. She holds a Bachelor of Science in Industrial Engineering from the Middle East Technical University and a Master of Science in Management Science and PhD in Industrial Engineering from Lehigh University. While at Lehigh University she was a Rossin Doctoral Fellow. Her email is ruken.duzgun@lehigh.edu.

AURELIE THIELE is an Associate Professor at Lehigh University in Bethlehem, PA. She holds a diplome d'ingenieur from the Ecole des Mines de Paris in France and a Master of Science and PhD in Electrical Engineering and Computer Science from MIT, where she also was a Visiting Associate Professor for the 2014-15 academic year in the Sloan School of Management. Her research is on decision-making under high uncertainty and has been funded by the National Science Foundation, an IBM Faculty Award and a Lehigh Innovation Grant, among others. Her email is aurelie.thiele@lehigh.edu.