Iterative Parametric Separation Scheme for Robust Optimization in Two-stage Stochastic Program

Wan-Lung NG
(mswlung@cityu.edu.hk)

Department of Management Sciences
City University of Hong Kong
Tat Chee Avenue, Kowloon Tong, Hong Kong.

Abstract – *Stochastic programming is a pro-active approach for decision making under uncertainty. However, a drawback of traditional stochastic linear programming is the lack of consideration of the variability of recourse costs. A novel formulation named robust optimization is recently gaining interest. Yet, when applying robust optimization to two-stage stochastic program, the model is in non-linear form and not separable by scenarios. This resulted that the second-stage problem, cannot be directly divided into smaller sub-problems, challenges the traditional solution scheme to standard linear program. In this paper, we propose an iterative parametric separation scheme. The second-stage problem can then be separated into a series of smaller problem. Efficient solution scheme can then worked out. A design flow diagram is also presented for developing a decision support system for such a problem.*

INTRODUCTION

Classical mathematical programming approaches are usually assumed the system parameters are given in deterministic. When solving such a problem under an uncertain environment, the general approach by using the classical mathematical programming model is to assume these parameters are well estimated by their corresponding expected value. However, it is found that a large error bound may arise if one attempts to adopt this mean-value approach. [1]

Early in 1950s, operation researchers recognized that the demand to take some pro-active approaches to consider the stochastic nature of the problem at the very beginning of modeling. One of widely adopted approaches is the stochastic programming [2,3]. One of the main streams in modeling the stochastic problem is to use the concept of recourse. The decision maker first choose a decision, some corrective actions are then taken after a realization of the random event. The objective is thus to minimize the total cost for first-stage and expected recourse cost for second-stage.

Most methodologies to solve the recourse problems are based on the decomposition principle and cutting planes algorithm. The most popular one is L-shape method by Van Slkye and Wets [4]. However, it has been argued that the linear stochastic program is not able to consider the variability of the uncertainty. The lack of consideration of variability will create some problems for decision maker. The standard stochastic linear programming is based on the expected recourse cost. When an extreme case of realization is given, the actual recourse cost can be very high. In this case, a company may run in to a financial problem if variability was not considered.

A recent approach by Mulvey et al [5] called robust optimization is to incorporate the variability of recourse cost into the planning. Similar to literatures in financial optimization and risk management, variance is usually used as the measurement of variability. However, when applying to two-stage stochastic program, the introduction of this higher-moment term creates challenges to traditional methodology. The L-shape method become handicapped as the variance term is not linear and is not separable by scenarios.

In this paper, we propose an iterative parametric separation scheme for the robust-optimization in two-stage stochastic program. The organization of this paper is as the following. Section 2 will briefly discuss the L-shape, which is the core concept for solving two-stage stochastic linear program. Robust optimization will be briefly discussed in section 3. We will also attempt to apply the robust optimization to the two-stage stochastic program. In particular, the challenges for solving the robust optimization second-stage recourse function will be highlighted. In section 4, we will establish our iterative parametric separation scheme for such robust optimization recourse function. In section 5, a design block diagram for the two-stage stochastic program with robust optimization will be presented. Some conclusion and recommendation will be provided in last section.

TWO-STAGE STOCHASTIC LINEAR PROGRAM

For simplification in our discussion, we assume the stochastic nature is discrete and modeled by scenario representation. Define Ω is a collection set of all possible scenarios $s = 1,2,\ldots,S$ such that $\Omega = \left\{ s \mid p_s \geq 0, \sum_{s=1}^{S} p_s = 1 \right\}$. A general stochastic linear program in extensive form is as the following,

[SLP_EF]

$$\begin{aligned} Min \quad & c^T x + \sum_{s=1}^{S} p_s \left(q_s^{\ T} y_s \right) \\ s.t. \quad & Ax = b \\ & W_s y_s = h_s - T_s x \qquad s = 1,2,\ldots,S \\ & x \geq 0 \ y_s \geq 0 \qquad s = 1,2,\ldots,S \end{aligned} \tag{1}$$

Where $x \in R^{n_1}$ is the first-stage decision vector. $c \in R^{n_1 \times 1}$ is the cost coefficients, and $Ax = b$, $x \geq 0$ are the constraints on x with $A \in R^{m_1 \times n_1}$ and $b \in R^{m_1 \times 1}$. We denote $X = \{ x \mid Ax = b, x \geq 0 \}$ as the feasible set for first-stage decision variables.

While in the second-stage, for scenario s ($s = 1,2,\ldots,S$), $y_s \in R^{n_2^s}$ is the decision variable under scenario s, $q_s \in R^{n_2^s}$ is the cost coefficient corresponding to scenario s. We denote the feasible set as $Y_s = \{ y_s \mid W_s y = h_s - T_s x, y_s \geq 0 \}$ with $W_s \in R^{m_2^s \times n_2^s}$, $h \in R^{m_2^s}$ and $T_s \in R^{m_2^s \times n_1}$.

When problem is small, the extensive form can be solved by standard linear optimization package. But, when the number of scenarios is large, the problem in extensive form becomes a large-scale problem. There are $n_1 + \left(n_2^1 + n_2^2 + \Lambda + n_2^S \right)$ decision variables, and totally $m_1 + \left(m_2^1 + m_2^2 + \Lambda + m_2^S \right)$ constraints.

The formulation indeed has a special block structure. We can apply decomposition principle and embed the second-stage cost as a function of x. The linear program is then decomposed into a series of sub-problems for second stage (with given x), and a master problem.

Decomposition for Stochastic Programming

A master problem of two-stage stochastic programming is as the following

[Master Problem]

$$\begin{aligned} Min \quad & c^T x + \Theta(x) \\ s.t. \quad & Ax = b \\ & x \geq 0 \end{aligned} \tag{2}$$

The function $\Theta(x)$ is called recourse function. In standard version of stochastic linear programming, the recourse function $\Theta(x)$ is the expected value of recourse for all different actions corresponding to different realizations of uncertainty. Define sub-problem for scenario s as

$$\begin{aligned} \phi_s(x) = \ & Min \ \ q_s{}^T y \\ & s.t. \quad W_s y = h_s - T_s x \\ & \qquad\quad y \geq 0 \end{aligned} \tag{3}$$

and

$$\Theta(x) = E[\phi_s(x)] \tag{4}$$

In each sub-problem, the scale for problem is reduced. For a particular s, the number of decision variable is only n_2^s and number of constraints is only m_2^s. Another obvious advantage is the sub-problems can be run in parallel if parallel computing equipment is available.

Based on this decomposition approach, Van Slkye and Wets developed a well-known solution scheme, called L-shape method, for the stochastic linear programming. While utilizing the decomposable property of the model, they incorporated cutting algorithms in the solution scheme. The lower bound of $\Theta(x)$ is denoted by a parameter θ. The value of $\Theta(x)$ in the master problem is then approximated by sequentially adding cutting planes to the problem until the feasibility and optimality are reached. The cutting planes are expressed in terms of θ and x with coefficients obtained from the sub-problems.

In each iteration, each sub-problem of linear optimization is feasible if and only if the following condition is satisfied.

$$\pi_s(h_s - T_s x) \leq 0 \tag{5}$$

where π_s is the dual multipliers for the sub-problem of scenario s. In case the sub-problem is infeasible, a constraint (5), called feasible cut, is then added to the master problem. The revised master problem is then solved to generate a new value of x for each sub-problem.

Since θ is the lower bound for the $\Theta(x)$, the value of θ can be further improved until the following condition is satisfied.

$$\theta = \sum_{s \in \Omega} p_s \pi_s (h_s - T_s x) \tag{6}$$

Otherwise, the following optimal cut is then added to the master problem

$$\theta \geq \sum_{s \in \Omega} p_s \pi_s (h_s - T_s x) \tag{7}$$

The master problem is then re-solved.

Owing to the property of separable by scenario in the second stage. Decision maker needs only to solve a series of small-scale linear optimizations. The expectation is then taken after the linear optimizations. This property makes the L-shape method becomes efficient in computations.

ROBUST FORMULATION IN TWO-STAGE STOCHASTIC PROGRAM

Robust optimization (RO) can be considered as an integration of multi-objective programming and stochastic programming. In the second stage, the recourse function is considering both the expected value of recourse costs and the variability of these costs.

A general model of RO in two-stage stochastic program can be formulated into similar decomposed form as the following.

[RO_Master]

$$\begin{aligned} Min \quad & c^{T}x + \widetilde{\Theta}(x) \\ s.t. \quad & Ax = b \\ & x \geq 0 \end{aligned} \tag{8}$$

The basic idea of robust optimization does not limit the choice for the recourse function $\hat{\Theta}(x)$. The robust optimization can be considered as an aggregate objective concerning both "solution robustness" and "model robustness". The terminologies "solution robust" and "model robust" are defined as the following.

Definition1 Solution Robustness: An optimal solution is solution robust with respect to optimality if it remains "close" to optimal for any scenarios $s \in \Omega$.

Definition 2 Model Robustness: An optimal solution is model robust with respect to optimality if it remains "almost" feasible for any scenarios $s \in \Omega$.

Similar to literatures in financial optimization and risk management, one of common-used functions for recourse function is in a mean-variance setting. The second-stage recourse function $\widetilde{\Theta}(x)$ can be expressed in mean and variance of recourse costs. Define, z is a random variable of recourse cost. We use $g(E[z], Var[z])$ for notation as the following.

$$g(E[z], Var[z]) = E(z) + \lambda Var(z) \tag{9}$$

The coefficient λ in the recourse function is a trade-off factor. The higher the value of λ, the decision maker emphases more on the variance term (i.e. solution robustness) than the expected term (i.e. model robustness), and vice verse.

Further arrange, we have an equivalent function in the two-moment formulation.

$$g(E[z], Var[z]) = \widetilde{g}(E[z], E[z^2]) \equiv E[z] + \lambda E[z^2] - \lambda (E[z])^2 \tag{10}$$

This recourse function $\widetilde{g}(E[z], E[z^2])$ is however, not separable by scenarios. Specifically, the non-separable is due to the existence of the term $(E[z])^2$. To solve the minimization of this recourse function, we have to solve a large-scale quadratic program. The decision variables under all scenarios have to be considered. All constraints in all scenarios have to be considered too. The dimension for decision variable will become $n_2^1 \times n_2^2 \times \Lambda \times n_2^S$. And number of constraints is $m_2^1 \times m_2^2 \times \Lambda \times m_2^S$. In this case, we can see we gain little to working decompose the problem than working on its extensive form directly.

Iterative Parametric Separation Scheme.

To utilize the efficiency of L-shape, i.e. separating into a series of small-scale problem, we have to look for a separation scheme so that the huge nonlinear optimization can be broken down. In this sub-section, we consider in the space of first-stage decision variable (i.e for a given x). Since $\widetilde{g}(E[z], E[z^2])$ is a convex function of $E[z]$ and $E[z^2]$. Define

$$\Delta_1 = \frac{\partial \widetilde{g}(E[z], E[z^2])}{\partial E[z]} = 1 - 2\lambda E(z) \tag{11}$$

$$\Delta_2 = \frac{\partial \widetilde{g}(E[z], E[z^2])}{\partial E[z^2]} = \lambda \tag{12}$$

Denote Π_1 as the optimal solution set for all y minimizing the problem with objective function $\widetilde{g}(E[z], E[z^2])$. We construct an auxiliary problem with objective function as the following

$$\hat{g}_{\mu}(E[z],E[z^2]) = E\{\mu(z)+\lambda(z^2)\} \tag{13}$$

The most prominent feature of the auxiliary problem is that the formulation is now separable by scenario. We can separable the problem into a series of quadratic programs for each scenario. Denote $\Pi_2(\mu)$ as the optimal solution set for all y minimizing the auxiliary problem for a particular μ.

Theorem 1: The optimal solution set to the original problem Π_1 is a sub-set of optimal solution set to the auxiliary problem $\Pi_2(\Delta_1)$

Proof. Assume $y^* \in \Pi_1$ but $y^* \notin \Pi_2(\Delta_1)$. Then there exist a y such that $\tilde{g}(E(z),E(z^2))|_{y^*} \le \tilde{g}(E(z),E(z^2))|_{y}$, but $\hat{g}_{\Delta_1}(y^*) > g_{\Delta_1}(y)$ .i.e.

$$[\Delta_1 \quad \lambda]\begin{bmatrix} E[z] \\ E[z^2] \end{bmatrix}\Bigg|_{y^*} > [\Delta_1 \quad \lambda]\begin{bmatrix} E[z] \\ E[z^2] \end{bmatrix}\Bigg|_{y} \tag{14}$$

Since $\hat{g}_{\Delta_1}(E[z],E[z^2])$ is a convex function of $E[z]$ and $E[z^2]$, the following property holds:

$$\hat{g}_{\Delta_1}(E[z],E[z^2])\Big|_{y} \ge \hat{g}_{\Delta_1}(E[z],E[z^2])\Big|_{y^*} + [\Delta_1 \quad \Delta_2]\left[\begin{bmatrix} E[z] \\ E[z^2] \end{bmatrix}\Bigg|_{y} - \begin{bmatrix} E[z] \\ E[z^2] \end{bmatrix}\Bigg|_{y^*}\right] \tag{15}$$

Combing $\Delta_2 = \lambda$, (14) and (15). This however, is contradicting the assumption of $y^* \in \Pi_1$

Theorem 2: An optimal solution y^* to the auxiliary problem (i.e. $y^* \in \Pi_2(\mu)$) is also optimal to the recourse function in two-moment form (i.e. $y^* \in \Pi_1$). A necessary condition is $\mu^* = 1 - 2\lambda E[z]|_{y^*}$.

Proof: All value of recourse cost is parameterized by y ($z = q_s^T y, s = 1,2,3,..,S$). While y is parameterized by μ. In other words, we can express all z in terms of μ as $z(\mu)$. The original problem is thus converted into the following abstract form.

$$\underset{\mu}{Max}\ \tilde{g}_{\mu}(E[z(\mu)],E[(z(\mu))^2]) = E[z(\mu)] + \lambda E[(z(\mu))^2] - \lambda(E[z(\mu)])^2 \tag{16}$$

A first-order condition for optimal μ^* is $\dfrac{\partial\left(\tilde{g}_{\mu}(E[z(\mu^*)],E[(z(\mu^*))^2])\right)}{\partial\mu} = 0$. i.e.

$$\frac{\partial\left(\tilde{g}_{\mu}(E[z(\mu^*)],E[(z(\mu^*))^2])\right)}{\partial E[z(\mu^*)]}\frac{\partial E[z(\mu^*)]}{\partial\mu} + \frac{\partial\left(\tilde{g}_{\mu}(E[z(\mu^*)],E[(z(\mu^*))^2])\right)}{\partial E[(z(\mu^*))^2]}\frac{\partial E[(z(\mu^*))^2]}{\partial\mu} = 0$$

$$(1 - 2\lambda E(z(\mu^*)))\frac{\partial E[z(\mu^*)]}{\partial\mu} + (\lambda)\frac{\partial E[(z(\mu^*))^2]}{\partial\mu} = 0 \tag{17}$$

On the other hand, for $y^* \in \Pi_2(\mu)$, we have the following necessary condition from $\hat{g}_{\mu}(E[z],E[z^2]) = E\{\mu(z)+\lambda(z^2)\}$,

$$\mu^* \frac{\partial E[z(\mu^*)]}{\partial \mu} + \lambda \frac{\partial E\left[(z(\mu^*))^2\right]}{\partial \mu} = 0 \qquad (18)$$

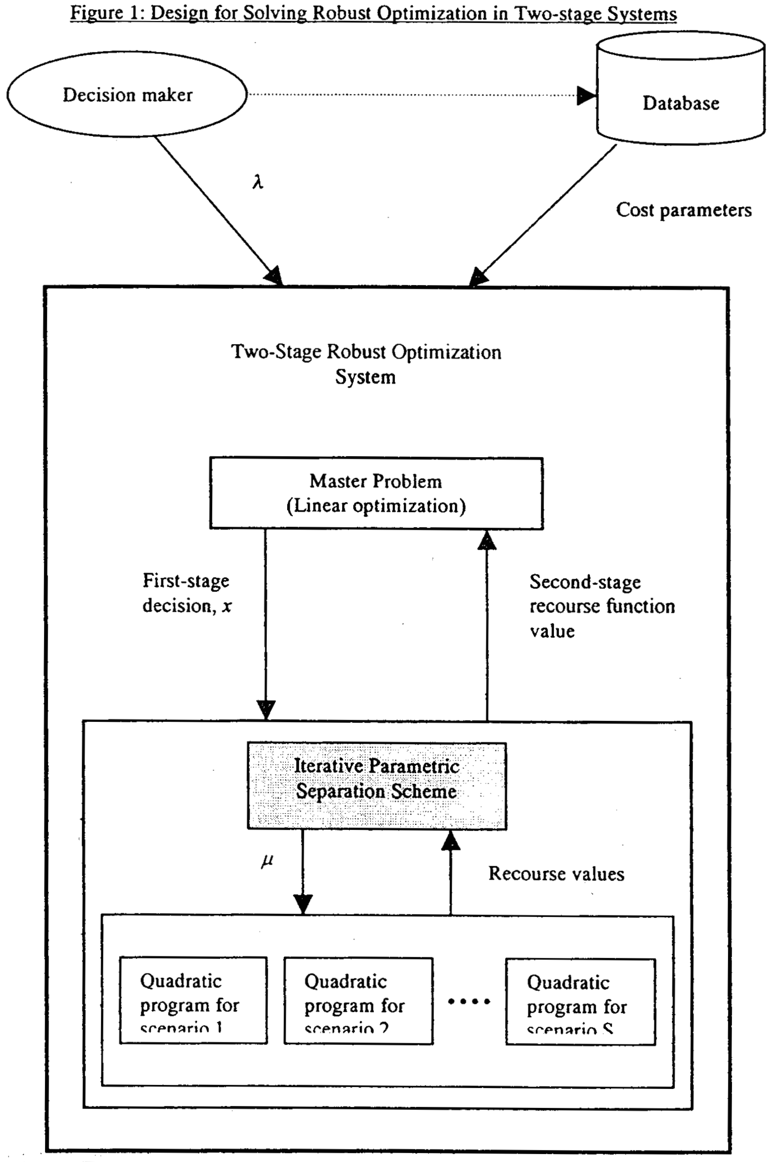
Thus the vector $\begin{bmatrix} (1 - 2\lambda E[z(\mu^*)]) \\ \lambda \end{bmatrix}$ is proportional to $\begin{bmatrix} \mu \\ \lambda \end{bmatrix}$. Since their second elements are both equal to λ. We further have $\mu^* = 1 - 2\lambda E[z]_y$.

Combining the two theorems, we can solve the second-stage robust optimization (a non-separable problem) via the well-constructed auxiliary problem. The remaining task is a one-dimension searching for the newly introduced parameter μ. The gradient for μ, ∇_μ = $(1 - 2\lambda E(z(\mu))) \frac{\partial E[z(\mu)]}{\partial \mu} + (\lambda) \frac{\partial E\left[(z(\mu))^2\right]}{\partial \mu}$. We can then setup a gradient-based searching scheme

MODULAR DESIGN FOR SOLUTION SCHEME

The Figure 1 summaries our solution scheme architecture.

Figure 1: Design for Solving Robust Optimization in Two-stage Systems



Risk attitude of parameter λ is required to input by the decision maker. Cost coefficients, and constraint coefficients are stored in a database. Scenario classification and values of probability for each scenario may be obtained from historical data. Sometime it may need from expertise and beliefs of decision makers. These inputs are then fed into the system via a *user-input module* and *data-retrieval module* for robust optimization in two-stage setting.

A *decomposition module* is required to break down the problem into a mater problem and a sub-problem (non-separable yet). *A linear optimization module* is used to solve the master problem to generate a first-stage decision variable. A *parametric separation module* then separates the sub-problem as we proposed in this paper. Each separated sub-problem is a quadratic programming. A *quadratic optimization module* is then called for solving each problem. If the problem is infeasible, a *feasible cut insertion module* is called to add to master problem. This looping ends until the sub-problems are all feasible. A *linear searching module* will be used to search for optimal parameter μ^*. The searching ends when the condition in theorem 2 is met. The solution then compares with the lower bound in master problem. This looping will end up when optimality is reached, otherwise, *optimal cut insertion module* will add cut to master problem.

CONCLUSIONS AND RECOMMENDATIONS

Robust optimization is an integration of multi-objective programming and stochastic programming. Robust optimization considers not only the value recourse cost but also the variability of this cost. However, a drawback to the formulation is the second-stage problem cannot by separated by scenario. Traditional tool, like L-shape method, become handicapped for solving the problem.

In this paper, we developed a separation scheme for robust optimization. The non-separable part of the model is embedding into a parameter, the second-stage problem is solved via a well-construct auxiliary problem, which is separable. The efficiency of tradition L-shape method can then be applied after implementation of the separation scheme. A basic solution scheme architecture is also proposed with major modules are being highlighted.

REFERENCES

[1]Birge, J. R. (1982), The Value of the Stochastic Solution in Stochastic Linear Programs With Fixed Recourse. Mathematical Programming, 24, 314-325.

[2]Birge, J. R. and Louveaux, F. (1997), Introduction to Stochastic Programming, Springe Verlag.

[3]Maarten H. van der Vlerk. (1996-2002), Stochastic Programming Bibliography. World Wide Web, http://mally.eco.rug.nl/biblio/stoprog.html.

[4]Van Slyke, R and Wets, R. J . B. (1969), "L-Shaped linear programs with application to optimal control and stochastic programming," SIAM Journal on Applied Mathematics, 17, 638-663.

[5]Mulvey J. M., Vanderbei, R. J. and Zenios, S. A (1995), Robust Optimisation of Large-Scale Systems, Operation Research, 43, 264-281.