On Solving a class of stochastic multiobjective integer linear programming problems with Interactive Based Approach

Suparni¹, Herman Mawengkang², Opim Salim Sitomput³, Saib Suwilo²

¹ Mathematics Doctorate Program, University of Sumatera Utara, Medan, Indonesia
² Department of Mathematics, University of Sumatera Utara, Medan, Indonesia
³ Faculty of Computer Science and Information Technology, University of Sumatera Utara, Medan, Indonesia

* hmawengkang@yahoo.com

Abstract. Decision problems of stochastic or probabilistic optimization arise when certain coefficient of an optimization model are not fixed or known but are instead, to some extent, stochastic(or random or probabilistic) quantities. This paper focused on multiobjective stochastic optimization. We propose a method for solving a multiobjective chance constraints integer programming problem based on interactive approach. We assume that there is randomness in the right-hand sides of the constraints only and that the random variables are normally distributed. Some examples are presented.

1. Introducing

Decision problems of stochastic or probabilistic optimization arise when certain coefficient of an optimization model are not fixed or known but are instead, to some extent, stochastic(or random or probabilistic) quantities.

In recent years methods of multiobjective stochastic optimization have become increasingly important in scientifically based decision-making involved in practical problems arising in economic, industry, health care, transportation, agriculture, military purposes and technology. We refer the Stochastic programming Web Site (2002) in [1] for links to software as well as test problem collections for stochastic programming. In literature there are many papers that deal with stability of solutions for stochastic multiobjective optimization problems. Among the many suggested approaches for treating stability for these problems in [2], [3], [4] and [5].

More recently, some papers for the author and others have been published in the area of stochastic multiobjective optimization problems, for example, in [6],[1] and [7]. In [6], a solution algorithm is presented for solving integer linear programming problems involving dependent random parameters in the objective functions and with linearly independent random parameters in the constraints. The main feature of the proposed algorithm is based mainly upon the chance-constrained programming technique in [8] along with the cutting-plane method of Gomory in [9]. In [1] reviewed theory and methodology that have been developed to cope with the complexity of optimization problems under uncertainty. The classical recourse-based stochastic programming, robust stochastic programming, probabilistic programming have been discussed and contrasted. In addition, the advantages and shortcomings of these models are reviewed. Applications and the state-of-the-art in computations are also surveyed and several main areas for future development in this field are reported. Stability of solution in multiobjective integer linear programming problems is investigated in [8], where the...
problem involves random parameters in the right-hand side of the constraints only and those random parameters are normally distributed. Some stability notions for such problems have been also characterized.

2. Problem Statement And The Solution Concept
The chance-constrained multiobjective integer linear programming problem with random parameters in the right-hand side of the constraints can be stated as follows:

\[(P): \text{max } F(x), \text{ subject to } x \in X, \]

where

\[X = \left\{ x \in \mathbb{R}^n \mid P\{g_i(x) \leq a_i \} \geq \alpha_i, i=1, 2, \ldots, m, x \geq 0 \text{ and integer, } j=1, 2, \ldots, n \right\} \]

Here x is the vector of integer decision variables and F(x) is a vector of k-linear real-valued objective functions to be maximized. Furthermore, P means probability and i α is a specified probability value. This means that the linear constraints may be violated some of the time and at most 100(1- α) of the time. For the sake of simplicity, we assume that the random parameters b_i (i = 1, 2, …m) are distributed normally with known means E{b_i} and variances Var{b_i} and independently of each other.

Definition 1.
A point x* \in X is said to be an efficient solution for problem (P) if there does not exist another x \in X such that F(x) ≥ F(x*) and F(x) ≠ F(x*) with

\[P\{g_i(x^*) \leq a_i \} \geq \alpha_i, i=1, 2, \ldots, m\]

The basic idea in treating problem (P) is to convert the probabilistic nature of this problem into a deterministic form. Here, the idea of employing deterministic version will be illustrated by using the interesting technique of chance-constrained programming [10]. In this case, the set of constraints X of problem (P) can be rewritten in the deterministic form as:

\[X' = \left\{ x \in \mathbb{R}^n \mid \sum_{j=1}^{n} a_{ij}x_j \leq E\{b_i\} + K_{\alpha_i} \sqrt{\text{Var}\{b_i\}}, i=1, 2, \ldots, m, x_j \geq 0 \text{ and integer, } j=1, 2, \ldots, n \right\} \]

where \(K_{\alpha_i}\) is the standard normal value such that \(\Phi(K_{\alpha_i}) = 1 - \alpha_i\); and \(\Phi(a)\) represents the “cumulative distribution function” of the standard normal distribution evaluated at a. Thus, problem (P) can be understood as the following deterministic version of a multiobjective integer linear programming problem:

\[(MP): \text{max } \{f_1(x), f_2(x), \ldots, f_k(x)\}, \text{ subject to } x \in X'. \]

Now it can be observed, from the nature of problem (MP) above, that a suitable scalarization technique for treating such problems is to use the \(\varepsilon\)-constraint method [2]. For this purpose, we consider the following integer linear programming problem with a single-objective function as:

\[Ps(\varepsilon): \text{max } f_s(x), \text{ subject to } \]

Subject to

\[X(\varepsilon) = \{x \in \mathbb{R}^n \mid f_r(x) \geq \varepsilon_r, r \in K - \{S\}, x \in X' \} \]

where s \in K = \{1, 2, \ldots, k\} which can be taken arbitrary.
It should be stated here that an efficient solution $x^*$ for problem (P) can be found by solving the scalar problem $P_0(\epsilon)$ and this can be done when the minimum allowable levels ($\epsilon_1, \epsilon_2, \ldots, \epsilon_{k-1}, \epsilon_k$) for the (k-1) objectives $f_1, f_2, \ldots, f_{k-1}, f_k$ are determined in the feasible region of solutions $X(\epsilon)$.

It is clear from [11] that a systematic variation of $\epsilon_i$'s will yield a set of efficient solutions. On the other hand, the resulting scalar problem $P_0(\epsilon)$ can be solved easily at a certain parameter $\epsilon = \epsilon^*$ using the branch-and-bound method [7]. If $x^* \in X(\epsilon^*)$ is a unique optimal integer solution of problem $P_0(\epsilon^*)$, then $x^*$ becomes an efficient solution to problem (P) with a probability level $\alpha_i^*, (i = 1, 2, \ldots m)$.

2.1. Problem Statement

The integer goal programming (IGP) problem can be stated mathematically as:

Minimize $c'_1p + c'_2m$

Subject to $Ax + lp - Im = b$

$x$ Integer

where $c_1$ and $c_2$ are vectors of weights placed on the violation of constraints.

$p_i$ and $m_i$ are variables showing by how much a given goal is violated.

Note that in a given constraint either $p_i$ or $m_i$ is certain to be zero in an optimal solution.

As we knew that goal programming is a problem structure of multiobjective programming. Therefore in deriving the method for solving the integer goal programming we start from solving the multiobjective integer programming (MOIP).

Consider the MOIP problem

$$\max [f_k(x), k \in K]$$

Subject to $x \in X = \{x | Ax \leq b, x \in [0, r], x \text{ integer}\}$

where $K = \{1, 2, \ldots, k\}$, $A$ is an $m \times m$ matrix coefficients of the constraints; $b$ is an $m$-vector of the right hand side, $b \in R^m$; $f_k(x), k \in K$, is a linear function of the decision variables, $x$ is an $n$-vector of decision variables, $x \in R^n$ and $r$ is an upper bound on $x$. By max we mean that all objective functions have to be maximized simultaneously.

2.2. Analytic Hierarchy Process

AHP proposed by Saaty [10] and [12] twenty years ago and is a widely used technique for multi-attribute decision making. It is based upon pairwise subjective judgment of elements which are used to complete a matrix. The eigenvalue for each element is then used to assess the contribution of that element to the overall component. As the name suggests, a hierarchy of matrices can be used where components are themselves elements of a higher order component. A typical example might be choosing a supplier on the basis of several criteria such as cost and quality. We would need to determine the relative contributions of cost and quality to the overall decision and also the relative degree to which each supplier possesses each criterion. It is normal to proceed from the more general to the more concrete.

Assume that there are $n$ elements, then we require $(n(n - 1))/2$ pairwise judgments to complete the matrix, where each judgment reflects the perception of the ratio of the relative contributions of elements $i$ and $j$ to the overall component be assessed so $a_{ij} = (w_i / w_j)$, subject to the following constraints: $a_{ij} > 0$, $a_{ij} = 1$ and $a_{ij} = (1/a_{ji})$. Saaty argues that the technique can only be effectively used where the elements are homogeneous, that is within the same order of magnitude, hence the ratios must range from 1/9 to 9.

In order to make the comparison process easier, some researchers attach semantic labels such as “equal” where the ratio is 1, “slightly more important” where it is 2 and so forth. For instance, if we considered quality to be “slightly more important” than cost, one would assign the value two to the appropriate cell in the matrix. In this case, the matrix would be completed as follows:
Each component has a priority scale, that is a derive ratio scale, to measure the contribution of each element to that component. This is based upon the approximate eigenvalue (i.e. divide the sum of the row by n) of each element.

One problem that can occur, especially since the judgments are subjective, is that values assigned are inconsistent. For example, one would expect to observe transitivity. Consistency can be measured as the deviation of the principal eigenvalue of the matrix from the order of the matrix.

The consistency index, CI, is calculated as follows:

\[
CI = \frac{\lambda_{\text{max}} - n}{n - 1}
\]

where \( \lambda_{\text{max}} \) is the maximum principal eigenvalue of the judgments matrix. The nearer CI is to zero the more consistent the judgments. The CI can be compared with the consistency index of a random matrix (RI). The ratio (CI/RI) is known as the consistency ratio (CR). Saaty suggests CR should be less than 0.1, although one should be cautious about attaching undue significance to this value.

Definition. The reference direction is defined by the difference between the reference point given by the DM and the last solution of the problem.

Let \( f_k \) denote an arbitrary value of the objective function of (1) and \( \bar{f}_k \) denote the aspiration level. Further let,

\[
H = \{ k \in K | \bar{f}_k > f_k \} \\
L = \{ k \in K | \bar{f}_k < f_k \} \\
E = \{ k \in K | \bar{f}_k = f_k \}
\]

where \( K = H \cup L \cup E \). To find the next solution of (1), we solve the following single objective surrogate problem:

\[
\max_{x \in X} s_1(x) = \max_{x \in X} \min_{k \in K} (f_k(x) - \bar{f}_k)/(\bar{f}_k - f_k)
\]

subject to

\[
f_k(x) \geq \bar{f}_k + \alpha(\bar{f}_k - f_k), k \in L, \\
f_k(x) = \bar{f}_k, k \in E
\]

where \( \alpha \) is a non-negative parameter and \( \bar{f}_k, k \in K \) is the objective function value found in the last solution.

The objective function (2), \( \max_{x \in X} s_1(x) \) maximizes the smallest standardized difference between the current solution \( f_k(x) \) and the last solution \( \bar{f}_k \) for all objective functions \( k \in H \). That is, it tries to take us as far as possible from the current solution. When we solve (2), the values off the objective functions that belong to set \( H \) increase whereas those that belong to set \( L \) may decrease. This way, function (1) exists only in \( \bar{f}_k > f_k \) for at least one \( k \in K \). This also implies that the reference point \( \bar{f}_k, k \in K \), does not have to be dominated by the previous solution of (1).

Depending upon the values of \( \bar{f}_k \) and \( f_k, k \in K \), sets \( H \) and \( L \) are created which define the single objective problem (2). When the sets \( H \) and \( L \) are non-empty, then the optimal solutions of (2) obtained for various values of \( \alpha \) are weak efficient solutions for (1). It is useful to note that the last solution of (1) is a feasible solution for (2); this is important when solving (2) by an exact algorithm. Further, the feasible solutions of (2) lie close to the efficient surface of (1) which allows us to use an approximate algorithm to solve (2).

Since the objective function of (2) is not linear, no standard algorithm for solving linear or linear integer programming problems can be used to solve it. However, the problem can be started as the following equivalent mixed integer linear programming problem.

\[
\max \ y
\]
The proposed algorithm consists of the following three steps.

Steps 1. Determine an initial (weak) efficient solution.

Steps 2. Show the solution to the DM. If DM is satisfied with the solution, stop; otherwise, ask the DM to specify a new reference point $\tilde{y}_k$, using AHP and go to step 3.

Steps 3. Based on the values of $\tilde{f}_k$ and $f_k$ (the last solution), solve (5) (or (11)) and find a new intermediate weak efficient (or efficient) solution $f_k(x)$; go to Step 2.

The initial weak efficient (or efficient) solution in Step 1 of the proposed algorithm is obtained by solving (5) and (14) for $\alpha = 0$, $f_k = 0$, $k \in K$ and $\tilde{f}_k = 1$, $k \in K$. If the values of some $f_k$, $k \in K$ can be negative, then problem (5) or (14) may be solved by replacing $y$ by $y_1 - y_2$ where $y_1, y_2 \geq 0$.

Since (2) and (14) are mixed integer linear programming problems, they may be solved by any standard exact algorithm. The branch-and-bound algorithms are the most appropriate for this purpose. Since we start with an initial feasible solution for the problem, the solution time in subsequent iterations (Step 3) can be considerably decreased. It may be noted that we start by solving the problem for $\alpha = 0$ and may continue the solution procedure parametrically for several new values of $\alpha$.
Problems (5) and (14) are NP-hard. The exact algorithms may take considerable time to solve problems of large dimensions. Therefore, it may be desirable to use an approximate algorithm to solve the problems in Step 3. It may be noted that, based on the theorems, the feasible solutions obtained by an approximate algorithm lie close to or on the weak-efficient (or efficient) surface; they are also used to formulate problem (5) or (14) for the next iteration. The preceding statements are also true when an exact algorithm is used to solve (5) and (14).

It may be pointed out that if at any iteration the aspiration levels desired by the DM exceed the objective function values (for all objectives) obtained at a previous iteration, i.e. \( \bar{f}_k > f_k \), for every \( k \in K \), the solution of problem (5) (or (14)) will be the same as the one obtained at the previous iteration. There are two ways to avoid this problem. One way is to require the DM to state the aspiration levels such that \( \bar{f}_k < f_k \) for at least one \( k \in K \). However, this puts an extra constraint on the DM. The second way to avoid the problem is to solve (22) (or (26)) instead of (5) (or (14)).

That is,

\[
\max(x_1 - x_2) \tag{22}
\]

Subject to

\[
f_k(x) - (\bar{f}_k - f'_k)(x_1 - x_2) \geq f'_k + \alpha(\bar{f}_k - f_k), k \in K \tag{23}
\]

\[
x \in X \tag{24}
\]

\[
z_1, z_2 \geq 0 \tag{25}
\]

Or

\[
\max(x_1 - x_2 + \beta \sum_{k \in K} y_k) \tag{26}
\]

\[
f_k(x) - f_k = y_k, k \in K \tag{27}
\]

\[
f_k(x) - (\bar{f}_k - f'_k)(x_1 - x_2) \geq f'_k + \alpha(\bar{f}_k - f_k), k \in K \tag{28}
\]

\[
x \in X, \tag{29}
\]

\[
y_k \geq 0, k \in K \tag{30}
\]

\[
z_1, z_2 \geq 0, \tag{31}
\]

where

\[
k_1 = \arg \max \left( (\bar{f}_k - f_k) \right)
\]

and

\[
f'_k = \begin{cases} f_k + \frac{(\bar{f}_k - f_k)}{2}, k = k_1 \\ f_k, \quad k = k_1 \end{cases}
\]

Unfortunately, the last weak-efficient (or efficient) solution may not be feasible for (22) (or (26)); and nothing definite can be said about the feasible solutions of these problems or whether they will lie near the efficient surface for (1).

4. Conclusion

This paper is dealing with proposing a method for solving stochastic multi-objective programming. The idea is to transform the stochastic problem into integer goal programming problem (IGP). Then we solve the IGP equivalent using interactive approach based on analytic hierarchy process.

References

[1] Saad, O. M., 2006. Optimization Under Uncertainty: A State-Of-The-Art, Paper Accepted For Publication In Applied Mathematics And Computation.
[2] El-Banna, A. Z. and Youness, E. A., 1993. *On Some Basic Notions Of Stochastic Multiobjective Problems With Random Parameters In The Constraints*. Microelectronics. Reliability, Vol. 33, No. 13, 1981-1986.

[3] Guddat, J., Vasquez, F., Tammer, K. and Wendler, K., 1985. *Multiobjective and Stochastic Optimization Based on Parametric Optimization*. Akademie-Verlage, Berlin.

[4] Osman, M. S. A. and Saad, O. M., 1994. *On the Solution of Chance-Constrained Multiobjective Linear Programming Problems with A Parametric Study*. Proceedings of the First International Conference On Operations Research and its Applications, Higher Technological Institute, Ramadan Tenth. City, Egypt.

[5] Vogel, S., 1992. *On Stability in Multiobjective Programming-A Stochastic Approach*. Mathematical Programming, 60, 91-119.

[6] Saad, O. M. and Kittani, H. F., 2003. *Multiobjective integer linear programming problems under randomness*. IAPQR TRANSACTIONS, Vol.28, No.2, 101-108.

[7] Sharif, W.H. and Saad, O.M., 2005. *On stability in multiobjective integer linear programming: A stochastic approach*. American Journal of Applied Sciences, Vol. 2, No. 12, 1558-1561.

[8] Seppälä, Y., 1998. *On Accurate Linear Approximations For Chance-Constrained Programming*. Journal of Operational. Research. Society, Vol. 39, No. 7, 693-694.

[9] Taha, H. A., 1980. *Integer Programming: Theory, Applications and Computations*. Academic Press, New York, 1975

[10] Saaty T. L., The Analytic Hierarchy Process. McGraw-Hill, New York.

[11] Chankong, V. and Haimes, Y. Y., 1983. *Multiobjective Decision-Making: Theory and Methodology*. (North Holland Series in System Science and Engineering).

[12] Saaty T. L., Highlights and critical bints in the theory and applications of the analytic hierarchy process. EJOR, 74(3) (1994), 426-447.

[13] Balinski, M., 1961. *An Algorithm For Finding All Vertices of Convex Polyhedral Sets*. SIAM Journal, Vol. 9, No. 1, 72-88.

[14] Osama, Ez-Eldin., 2000. *On Stochastic Multiobjective Integer Linear Programming Problems*. M.SC. Thesis, Helwan University, Cairo, Egypt.

[15] Osman, M. S. A., 1977. *Qualitative Analysis of Basic Notions in Parametric Convex Programming I*. (Parameters in the Constraints), Applied. Math. CSSR Akad. Ved. Prague, 22, 318-332.

[16] Osman, M. S. A., 1977. *Qualitative Analysis of Basic Notions in Parametric Convex Programming II*. (Parameters in the Objective Function), Applied. Math. CSSR Akad. Ved. Prague, 22, 333-348