STOCHASTIC OPTIMIZATION FORMULATIONS FOR STRUCTURE DESIGN

Shivanandini V. Tanuku
Department of Mechanical Engineering
Old Dominion University
Norfolk, VA 23529
757-683-3728/757683-5344

May T. Hou
Department of Computer Science
Norfolk State University
Norfolk, VA 23504
757-823-9455/757-823-9229

Gene J.-W. Hou
Department of Mechanical Engineering
Old Dominion University
Norfolk, VA 23529
757-683-3728/757683-5344

ABSTRACT

This paper uses a simple three-bar truss problem as a vehicle to investigate the qualities of the optimal solutions resulted from various stochastic programming formulations. The specific stochastic programming techniques selected in this study include the worst case design, the chance programming and the design with recourse. These methods have been widely studied in operations research, but seldom used in engineering design. The three-bar truss problem has three design variables. The structural weight is considered as the design objective, which is related to the production cost. The truss structure is subjected to nine constraints ranging from the yielding to the resonance condition. The direction of the load, which is applied to the tip of the structure, can be varied between $-45^0$ to $90^0$. The worst case design treats the direction of the load as a problem parameter, while the chance programming and the design with recourse treat it as a uniformly random variable. Numerical integration, Monte Carlo simulation and the optimizer provided by MATLAB code are used in this study. Most of the stochastic programming formulations used here arrive at similar optimal solution. The paper is concluded with comments on each of the formulations in respect to the efficiency in computation and the accuracy in representing design consideration.

INTRODUCTION

A product inevitably faces many uncertainties from the on-set of its design through its entire life cycle. These uncertainties may come from material characterization, manufacturing process, operational environment and in-service conditions. Furthermore, many in-service practices, such as regular maintenance, inspection and repair or replacement, new mission requirement and technology advancement may prolong, restore or enhance the product in its service. For example, Jones et al used biological optimization process, in conjunction with the non-destructive inspection, to rework the geometry of the component so as to extend its fatigue life. Kirby and Mavris (2002) argued that the product using the current technology to satisfy today’s needs may
become obsolete when the product is actually fielded. Potential technologies have to be identified, found and incorporated into design process for the future product. To achieve a better product, these uncertainties and recourses have to be considered in the design and analysis processes. Design under uncertainty is a discipline that deals with such a problem. Mathematically, design under uncertainty can be formulated as a stochastic programming problem whose objective and constraints involve uncertainty terms.

Stochastic programming has been widely used for risk management, such as financial planning and control, investment expansion, aircraft routine allocation, production planning, water resource modeling, public policy of energy, etc. (Birge and Louveaux 1997, Kleywegt 2001 and Shapiro 2000). Most of these applications involve recourses. Their stochastic programming formulations include the expected value is a part of the objective. References of Shapiro and his colleagues derived the conditions under which the expectation of a derivative is equal to the derivative of the expectation. If these conditions are held, the sample averages of the function as well as its derivative converge to their true values with probability one as the number of samples goes to infinite. Once the averages of the function and its derivative can be estimated, the stochastic programming problem becomes deterministic and any proper mathematical programming method can be applied. This is the basis of the sample average approximation (SAA) method, which is a Monte Carlo sampling technique. The method is called by various names by different names; the sample path method, the stochastic counterpart method. Statistical inference of the estimators of the optimal solutions has been established by Shapiro et al, which helps to construct the stopping criterion, validation analysis and error bounds of the method. Variance Reduction methods are also suggested to improve the efficiency of the method. There are cases where the expected value appearing in the objective function is a result of an optimization problem. Note that the decision variable in the sub-problem is different from that of the primary objective function. The decision variable can be realized only if the random variable is known. Such a problem is called as a Stochastic Mathematical Programming with Equilibrium Constraints (SMPEC) problems, which is in fact a here-and-now type of two-stage stochastic programming problems with recourse. Shapiro (2008) showed that, with some regularities of the sub-optimization problem, the expected value is continuous and directional differentiable. The SAA method discussed in (Shapiro 2000b) can be extended to the SMPEC problem. The paper further showed that the solution of the SAA method can converge to that of the SMPEC, once the number of the samples approaches infinite. One alternative of the SMPEC formulation is the robust design in which constraints are imposed for every possible realization (Shapiro 2004). This formulation treats all the constraints as hard constraints, which are satisfied with 100% probability. The other is to satisfy the constraints with a given probability. This leads to the chance programming or probabilistic constraint formulation.

Engineers have long recognized the importance of including uncertainty in their designs and analyses. For examples, Monohar and Saha (2005) considered the most critical loading scenario that can exploit the maximal failure for the weakest designed structure. Most of the engineering applications formulated and solved their in the form of chance or probabilistic programming. Some of the research work, however, investigated the worst case design.

Reinhart (1998) used the stochastic approximation method to find the solution of stochastic programming problems, in which the objective is the expectation of a random function defined in
the feasible space. It was suggested to use the response surface method to approximate the
expectation. The important sampling method was used to evaluate the averaged value at the
design points. The gradient of the averaged value of the function with respect the design variable
was then obtained approximately. Reinhart performed error analysis in order to control the error
in gradient approximation. Eschenauer and Vietor (1995) conducted shape optimization of
mechanical components with a brittle failure criterion, which is described by a Weibull-
distribution. The problem was formulated in form of chance programming in which the
constraint is expressed in terms of the probability of failure. The probability of failure was
evaluated through numerical integration and the entire process was supported by a commercially
available finite element code. Gasser and Schueeller (1998) studied the reliability-based design
optimization formulations in which both the objective and the constraints are expressed in terms
of probability measures. They recommended the use of the response surface method to model the
limit state constraints in terms of design variables. The corresponding probability of failure can
then be estimated efficiently using MC simulation. Choi et al (1999, 2001) used the Most Proper
Point approach to speed up the computation of the reliability index and its design derivatives,
with which they developed an efficient finite element-based approach to solve the reliability-
based design optimization problems.

Evgrafov et al (2003) considered stochastic topology design optimization which is an
element of the SMPEC problems. Their problem includes the equilibrium constraints which are in
fact the state equations, results of minimization. The inequality constraints of their problem are
the point-wise stress constraints, which are too restrictive for the problem being stable to the
variation in distribution parameters. Evgrafov and Patriksson (2001) then replaced the point-wise
constraints by an inequality probability constraint. Thus, the SMPEC problem was simplified as
a chance programming, which limited the probability of the stress violation to be less than a
relaxation parameter. The paper continued to prove that the chance programming problem has a
solution and is continuous with respect to the bounds, the relaxation parameter and the
distribution parameters. As the relaxation parameter approaches to zero, the optimal solution of
the chance programming problem approaches to the optimal solution of the SMPEC problem.
This theoretical development leaded to a numerical algorithm, which was successfully validated
by a truss topology design example.

Haug and Arora (1979) designed structures subjected to a range of the direction and the
location of the loads and Li and Padula (2004) conducted airfoil design optimization over a range
of Mach number. Haug and Arora formulated their problems as the worst case design. In their
solution algorithm, however, they replaced the worst case constraint by its first order necessary
condition. Li and Padula also cast their problems as the worst case design. They proposed the
Randomized Multipoint Optimization Method which constructed linear sub-problems at each of
the randomly selected Mach numbers to start each of the design iterations. The sub-problem was
solved by a sequential linear programming for the changes in the design variables and the angle
of attack. The sub-problem was set to minimize the expected value of the drag while ensures the
lift is satisfied and the drag is reduced at every selected Mark number. The design optimization
algorithm was terminated if either the mean of the drag or the drag itself at each of the design
points can not be reduced further. The number of the design variables was in the range of 50 in
their example problems, while the number of the random Mach numbers was 4.
The main objective of this study is to investigate the qualities of the optimal solutions resulted from various stochastic programming formulations. The specific stochastic programming techniques selected in this study include the worst case design, the chance programming and the design with recourse. A simple three-bar truss problem is used as a vehicle to facilitate the investigation. The three-bar truss problem has 3 design variables and 9 constraints. The truss is subjected to a point load whose angle of inclination is varied between $-45^0$ and $90^0$. This angle introduces a uncertainty to the design problem. The constraint reliability as well as the system reliability is the two reliability formulations to be studied in the chance programming, while the expected cost of violations and the cost of the minimal violation are the two recourse formulations to be studied in the design with recourse. Note that this study focuses on numerical feasibility and accuracy, rather than numerical efficiency. Both the Monte Carlo simulation and the numerical integration are used in this study to pursue the optimal solution. Particularly, the concept of the logic function (Ehle et al 1982, Hou et al 2006) is used to aid the numerical integration scheme.

**THREE-BAR TRUSS PROBLEM**

The three-bar truss example, shown in Fig. 1, was used by Haug and Arora to study the worst case design algorithm. The problem is repeated here as the vehicle to study various problem formulations in stochastic programming. The design objective is to find out the cross-sectional areas, $b_1$, $b_2$ and $b_3$ of the members, so that the lightest possible truss can be produced. Here $b_1$, $b_2$ and $b_3$ are the decision or design variables. While designing the truss, the constraints on stress, buckling, deflection and resonance, in addition to the bounds on the design variables, need to be considered.

![Three-Bar Truss Diagram](image)

**Figure 1: Three-Bar Truss**

In a deterministic formulation, a typical structure design optimization problem is expressed as
\[
\begin{align*}
\min_{b \in \mathbb{R}^3} & \quad f(b) \\
\text{subject to:} & \quad g_i(b, z) \geq 0, \quad i = 1, 2, \ldots, 9
\end{align*}
\]  

(P1)

where \( z \) is the state variable whose relation with respect to the design variable, \( b \), is determined by the associated state equation. Specifically, for the three-bar truss problem, the weight of the structure, \( W \), is considered as the cost function. The static response of the structure is the solution of the matrix equation as

\[
K(b)z - s = 0
\]  

where \( z = [z_1, z_2]^T \) gives the horizontal and vertical nodal displacements. The stiffness matrix, \( K(b) \), is a positive definite stiffness matrix and the load vector, \( s \), is given by:

\[
s = P \cos \alpha \quad P \sin \alpha^T
\]

where \( P \) is the applied point load and \( \alpha \) is the angle of load application measured from the horizontal axis. Its value ranges from \(-45^0\) to \(90^0\).

The stress and the buckling constraints are imposed upon each of the three members. Their expressions include the nodal displacements and the design variables. Two displacement constraints are imposed to limit the deflection of the tip of the truss structure. Finally, to avoid the resonance problems, the fundamental frequency of the structure must be constrained, in which the fundamental natural frequency is the solution of the general eigenvalue problem:

\[
K(b)y = \xi M(b)y
\]

where \( K(b) \) is the stiffness matrix and \( M(b) \) is the mass matrix of the truss structure and \( y \) is the associated eigenvector.

**STOCHASTIC PROGRAMMING FORMULATIONS**

Four major design formulations are presented in this study; the deterministic design optimization, the worst case design, the chance programming and the design with recourse. All of the minimization problems are solved by the \textit{fmincon} option of Mathlab.

**The Deterministic Design**

The direction of the load, \( \alpha \), in the design problem defined by Problem (P1), is undetermined. However, to understand the relation between the deterministic optimization solution and the angle, \( \alpha \), Problem (P1) is solved for every increment of \( 1^0 \) between \(-45^0\) and \(90^0\). The problem is solved again for an increment of \( 0.5^0 \). Both give very similar results. Figures 1 and 2 show that the optimal objective and the design variables are functions of the angle, \( \alpha \). As expected, the active constraint set of the optimal solution is varied when the angle, \( \alpha \), is varied between \(-45^0\) and \(90^0\). This variation can cause the discontinuity as observed at \(45^0\). The heaviest design occurs at \(0^0\), which gives the best objective, \(15.999 \text{ in}^2\) and the best \(b_1\), \(b_2\) and \(b_3\) at \(5.6569 \text{ in.}, 1.40\text{E-06 in.} \) and \(5.6569\text{in.}\), respectively. However, this heaviest design does not satisfy the constraints for all of the angles between \(-45^0\) and \(90^0\).
The Worst Case Design

The mathematical formulation of the worst case design is described in Problem (P2).

\[
\min_{b \in \mathbb{R}^3} \ f(b)
\]

subject to:

(P2)

\[
\min_{\alpha} \ \{g_i \mid g_i(z, \alpha) \leq 0\}, \ i = 1, 2, \ldots, 9
\]
An important step in the solution process is to identify the angles, $\alpha$, that yields the largest constraint value. In present study, Monte Carlo simulation is employed in each of design iterations to generate 1,000 samples of $\alpha$, uniformly distributed between $-45^0$ and $90^0$. The worst constraint value among the samples is then identified and selected to represent the constraint value for each of the constraints. The initial run took 41,208 function evaluations to reach the best design at $b = (6.532, 3.381, 6.532)$ with an objective value of 21.854.

It is understood that this approach does not yield same constraint set between consecutive design iterations. In other words, the constraint values are not continuous between consecutive design iterations. Nevertheless, the process converges and produces a better design than that presented by Haug and Arora. Different optimization runs were performed for different numbers of samples and different initial designs. All of them converged to the same solution. That demonstrated the robustness of the solution process.

**The Chance Programming**

The chance programming is a deterministic problem in which the stochastic constraint is measured by its probability or reliability. In the following formulation, each of the constraint is required to have 99% reliability.

$$\min_{b \in R^3} f(b)$$

subject to:

$$R(\bigcup_{i=1}^{9} g_i, z, b) \leq 0 \geq 0.99,$$  \hspace{1cm} i = 1,2,..,9 \tag{P3}$$

The reliability given in Problem (P3) is evaluated in two approaches: Monte Carlo Simulation and the integration method. In the integration method, the logic function is introduced to smoothly approximate the indicator. The Simpson’s rule is then used to compute the integral numerically. Using 10,000 samples, the method took 109,000 function evaluations to reach at the optimal solution, $b = (6.529, 3.370, 6.529)$, with the objective value of 21.841.

A structure is usually considered failed, if it cannot maintain any of the constraint. Therefore, the chance programming should count for the system reliability, not just individual constraint reliability. It leads to the alternative of Problem (P3)

$$\min_{b \in R^3} f(b)$$

subject to:

$$R\left(\bigcup_{i=1}^{9} g_i, z, b\right) \leq 0 \geq 0.99 \tag{P4}$$

The problem is solved again with 10,000 samples for the integration method. Both methods converge to an optimal solution, $b = (6.35, 4.27, 6.35)$ with an objective of 22.242, which is slightly higher than that of Problem (P3).
The Design with Recourse

Modern civil structures are usually equipped with health monitoring facility to inspect and discover the early symptom of failure so as to restore its original performance by repair or replacement. This recourse effort is indicated by the amount of failure incurred in each of the constraints. Thus, in the new formulation, both the construction cost and the cost of repair cost are considered in the object function. The initial construction cost is measured by the weight of the structure, whereas the repair cost which is unknown now is measured by its expectation. Therefore, the new design problem is given as

\[
\min_{b \in \mathbb{R}^3} f \left( \sum\limits_{i=1}^{9} \gamma_i y_i \right) \quad \text{where} \quad Q \left( \bar{\xi}, \alpha \right) = \left\{ \sum_{i=1}^{9} \left( y_i \right)^2 \, | \, y_i = g_i, \xi, \sum_{i=1}^{9} y_i \geq 0 \right\}
\]

Note that no sub-optimization is involved in determining the cost of repair, \( Q \left( \bar{\xi}, \alpha \right) \). The problem (P5) implies that any damage, regardless of single or multiple failures, will be repaired. However, those constraints are not independent to each other. Once a single failure mode is repaired, it may restore the other failure mode as well. It is thus suggested an alternative to Problem (P5) that the repair will be placed on the first occurred failure mode. Mathematically, the new formulation is stated as

\[
\min_{b \in \mathbb{R}^3} f \left( \sum\limits_{i=1}^{9} \gamma_i y_i \right) \quad \text{where} \quad Q \left( \bar{\xi}, \alpha \right) = \min \left\{ y_i \, | \, y_i = g_i, \xi, \sum_{i=1}^{9} y_i \geq 0; i = 1,2,..9 \right\}
\]

In other words, the repair will start with the first possible failure among the 9 constraints. That is, the one constraint that has the lowest positive value. Though they take different weighting coefficient, \( \gamma \), in their respective formulations, Problems (P5) and (P6) reach at the similar optimal solutions.

The Problem (P5) was first solved by 1,000 samples of Monte Carlo simulation. It took 181,000 function evaluations to reach the optimal design, \( b = (\xi, 532 3.381 6.532) \) with the optimal objective of 21.854. The expectation of Problem (P5) was then evaluated by using the 20 and 32-point Gauss-Legendre integration scheme. The minimal positive violations among 9 constraints at the designated Gauss-Legendre quadratures were first identified. Their values were then used to integrate the expectation. The former took 3,220 function evaluations to reach the optimal design, \( b = (\xi, 525 3.385 6.526) \) with the optimal objective of 21.841, while the later took 7,264 function evaluations to reach \( b = (\xi, 525 3.384 6.530) \) with the optimal objective of 21.845. The success of the integration scheme suggested that the expectation of the minimum positive violation is a smooth function of the design variables, even though the minimum positive violation is not continuous as a result of variations in \( \alpha \). The Logic function and the Simpson’s rule were used to approximate and integrate the expectation of Problem (P6). It used...
10,000 data points. The method took 92,000 iterations to reach \( b = 0.328 \) \( 3.325 \) \( 6.320 \) with the optimal objective of 21.464.

**KEY RESULTS AND CONCLUSIONS**

This paper investigated different formulations of stochastic programming for the three-bar truss design problem. All of the tested formulations reached the similar optimal solution. This may be due to the fact that the three-bar truss design problem is well-behaved. Among all of the formulations investigated, the design with recourse of Problem (P6), which includes the expectation of the minimal repair cost, seems the most promising. It is closest to the engineering practice among all. And the paper also demonstrated that it can be solved efficiently with an integration scheme.

**REFERENCES**


