SPRINGER BRIEFS IN OPTIMIZATION

Urmila Diwekar
Amy David

BONUS Algorithm for Large Scale Stochastic Nonlinear Programming Problems

Springer

SpringerBriefs in Optimization

SpringerBriefs in Optimization showcases algorithmic and theoretical techniques, case studies, and applications within the broad-based field of optimization. Manuscripts related to the ever-growing applications of optimization in applied mathematics, engineering, medicine, economics, and other applied sciences are encouraged.

For further volumes:
http://www.springer.com/series/8918

Urmila Diwekar • Amy David

BONUS Algorithm for Large Scale Stochastic Nonlinear Programming Problems

Springer

Urmila Diwekar
Clarendon Hills
Illinois
USA

Amy David
Krennert School of Business
Purdue University
West Lafayette
Indiana
USA

ISSN 2190-8354 ISSN 2191-575X (electronic)
SpringerBriefs in Optimization
ISBN 978-1-4939-2281-9 ISBN 978-1-4939-2282-6 (eBook)
DOI 10.1007/978-1-4939-2282-6

Library of Congress Control Number: 2014955715

Springer New York Heidelberg Dordrecht London

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

To my husband Dr. Sanjay Joag,
who changed his career to support mine,
for his constant love and support.
–Urmila

Preface

Stochastic programming problems are very difficult to solve as they involve optimization as well as uncertainty analysis. Algorithms for solving large-scale nonlinear stochastic programming problems are very few in number, as are the engineering applications of these problems. This book introduces two algorithms for large-scale stochastic nonlinear problems for both open equation systems and black box models. These algorithms are the Better Optimization of Nonlinear Uncertain Systems (BONUS) algorithm and the L-shaped BONUS algorithm. Real-world applications of these algorithms in the areas of energy and environmental engineering are also detailed. Many have contributed to this book. Researchers who worked with Dr. Diwekar including Dr. Adrian Lee, Dr. Kemal Sahin, Dr. Juan Salazar, and Dr. Yogendra Shastri, as well as collaborators such as Dr. Emil Constantinescu, Dr. Victor Zavala, and Dr. Stephen Zitney have provided the material for this book with their research. Thanks also to our group members Dr. Pahola Benavides, Dr. Berhane Gabreslassie, Dr. Rajib Mukherjee, Shivam Tyagi, and Kirti Yenki who went through the first draft of the book and meticulously pointed out mistakes. Hope you enjoy this work.

Urmila Diwekar and Amy David

Contents

List of Figures

List of Tables

Chapter 1
Introduction

A general optimization problem can be stated as follows.

$$\underset{x}{\text{Optimize}} \quad Z = z(x) \tag{1.1}$$

subject to

$$h(x) = 0 \tag{1.2}$$

$$g(x) \leq 0 \tag{1.3}$$

The goal of an optimization problem is to determine the decision variables x that optimize the objective function Z(Eq. 1.1), while ensuring that the model operates within established limits enforced by the equality constraints h (Eq. 1.2) and inequality constraints g (Eq. 1.3).

Figure 1.1 illustrates schematically the iterative procedure employed in a numerical optimization technique. As seen in the figure, the optimizer invokes the model with a set of values of decision variables x. The model simulates the phenomena and calculates the objective function and constraints. This information is utilized by the optimizer to calculate a new set of decision variables. This iterative sequence is continued until the optimization criteria pertaining to the optimization algorithm is satisfied. If the objective function and constraints are linear and the decision variables involved are scalar and continuous, then it is a linear programming (LP) problem. However, if the objective function and/or constraints are nonlinear then it is a nonlinear programming (NLP) problem. An NLP problem involving integers is a mixed integer nonlinear programming (MINLP) problem.

1.1 Stochastic Optimization Problems

Stochastic optimization gives us the ability to optimize systems in the face of uncertainties. A stochastic optimization or a stochastic programming (SP) problem requires that the objective function and constraints be expressed in terms of some

U. Diwekar, A. David, *BONUS Algorithm for Large Scale Stochastic Nonlinear Programming Problems,* SpringerBriefs in Optimization, DOI 10.1007/978-1-4939-2282-6_1

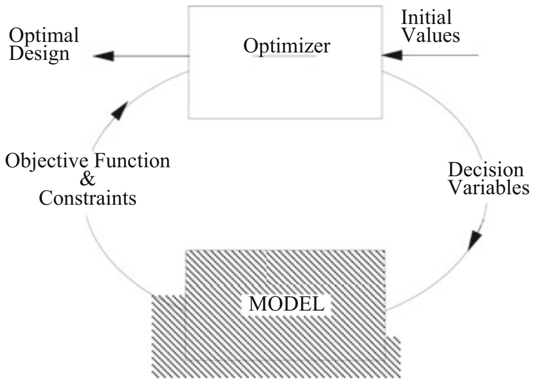

Fig. 1.1 Pictorial representation of the numerical optimization framework [7]

probabilistic representation (e.g., expected value, variance, fractiles, most likely values). For example, in chance constrained programming, the objective function is expressed in terms of expected value, and the constraints are expressed in terms of fractiles (probability of constraint violation), and in Taguchi's offline quality control method ([55], Diwekar and Rubin 1991), the objective is to minimize variance. These problems can be further classified as stochastic linear programming (SLP), stochastic nonlinear programming (SNLP), and stochastic mixed integer linear and nonlinear programming problems. The latter problems are the focus of this book.

A generalized stochastic optimization problem, where the decision variables and uncertain parameters are separated, can then be viewed as:

$$\underset{x}{\text{Optimize}} \quad J = P_1(j(x,u)) \tag{1.4}$$

subject to

$$P_2(h(x,u)) = 0 \tag{1.5}$$

$$P_3(g(x,u) \geq 0) \geq \alpha \tag{1.6}$$

where u is the vector of uncertain parameters and P represents the cumulative distribution functional such as the expected value, mode, variance, or fractiles. Figures 1.2 and 1.3 show the expected value (mean), mode, variance, and fractiles for a probabilistic distribution function.

Unlike the deterministic optimization problem, in stochastic optimization one has to consider the probabilistic functional of the objective function and constraints. The generalized treatment of such problems is to use probabilistic or stochastic models instead of the deterministic model inside the optimization loop. Figure 1.4 represents the generalized solution procedure, where the deterministic model is replaced by an iterative stochastic model with a sampling loop representing the discretized uncertainty space. The uncertainty space is represented in terms of the moments such as the mean, or the standard deviation of the output over the sample space of N_{samp} as given by the following equations (Eqs. 1.7 and 1.8).

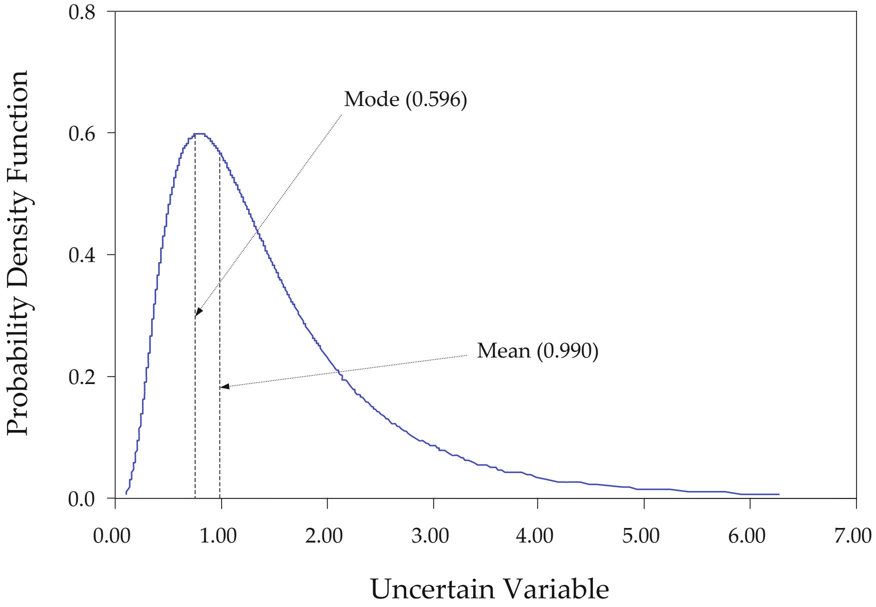

Fig. 1.2 Different probabilistic performance measures (PDF) [7]

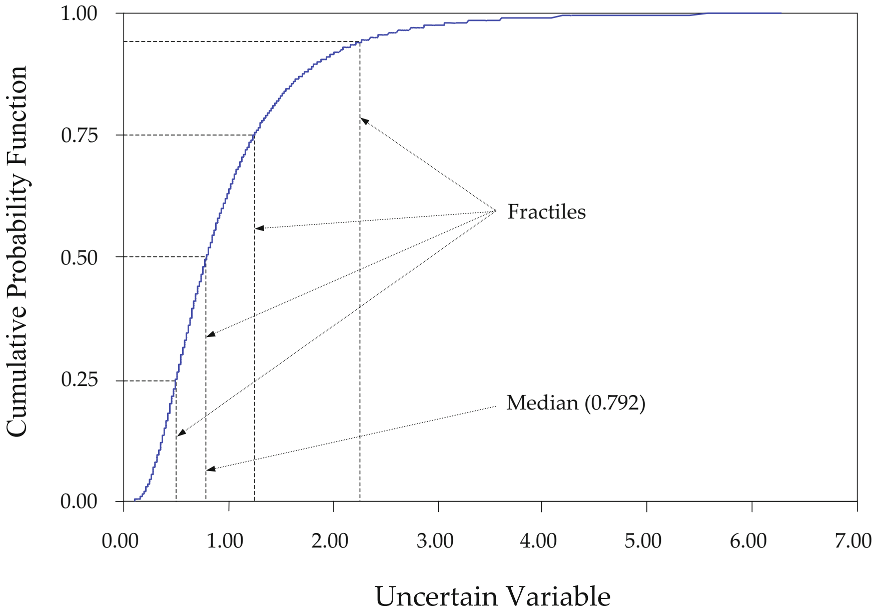

Fig. 1.3 Different probabilistic performance measures (CDF) [7]

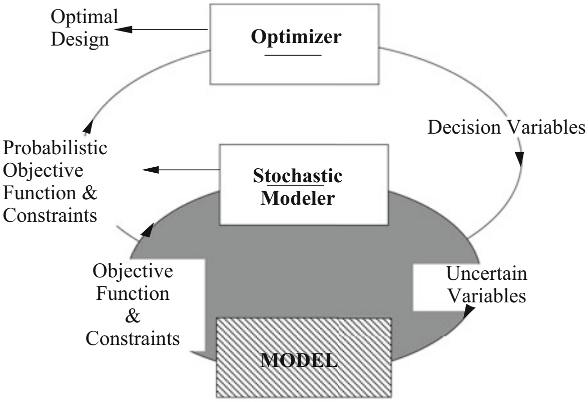

Fig. 1.4 Pictorial representation of the stochastic programming framework

$$E(z(x,u)) = \sum_{k=1}^{N_{samp}} \frac{z(x,u_k)}{N_{samp}} \tag{1.7}$$

$$\sigma^2(z(x,u)) = \sum_{k=1}^{N_{samp}} \frac{(z(x,u_k) - \bar{z})^2}{N_{samp}} \tag{1.8}$$

where $\bar{z}$ is the average value of z. E is the expected value and σ^2 is the variance.

1.2 Stochastic Nonlinear Programming

There are two fundamental approaches used to solve SNLP problems . The first set of techniques identify problem specific structures and transforms the problem into a deterministic NLP problem . For instance, chance constrained programming [4] replaces the constraints that include uncertainty with the appropriate probabilities expressed in terms of moments. The major restrictions in applying the chance constrained formulation include that the uncertainty distributions should be stable distribution functions, the uncertain variables should appear in the linear terms in the chance constraint, and that the problem needs to satisfy the general convexity conditions. The advantage of the method is that one can apply the deterministic optimization techniques to solve the problem.

Decomposition techniques like L-shaped decomposition [2]) divide the problem into stages and generate bounds on the objective function by changing decision variables and solving subproblems that determine the recourse action with respect to the uncertain variables. However, these methods also require convexity conditions and/or dual-block angular structures like the one shown in Fig. 1.5, and are only applicable to discrete probability distributions. For example, Lagrangian-based approaches have been applied to nonlinear SP formulations. The Lagrangian dual ascent

$$\begin{array}{llllllllll}
\min & c_0^T x_0 & + & c_1^T x_1 & + & c_2^T x_2 & + \; \dots \; + & c_N^T x_N & & \\
\text{s.t.} & A x_0 & & & & & & & = & b_0, \\
& T_1 x_0 & + & W_1 x_1 & & & & & = & b_1, \\
& T_2 x_0 & & & + & W_2 x_2 & & & = & b_2, \\
& \vdots & & & & & \ddots & & & \vdots \\
& T_N x_0 & & & & & + & W_N x_N & = & b_N, \\
& x_0 \geq 0, & & x_1 \geq 0, & & x_2 \geq 0, & \dots, & x_N \geq 0. & &
\end{array}$$

Fig. 1.5 Example of a dual block angular structure (LP), each diagonal block is a realization of a random variable(scenario or sample)

method has been proposed by Rockafellar and Wets [38] for problems with finite outcomes for the uncertain variables. Another technique is Regularized Decomposition, which adds quadratic terms to the objective for improved convergence [42] of the L-shaped decomposition method. The augmented Lagrangian method adds a quadratic penalty to ensure convexity, yielding more efficient computation (Dempster, 1988). Rockafellar and Wets also developed a similar technique, the progressive hedging algorithm [39] . These methods have limitations in terms of handling uncertain variables. An alternative approach that can be used to capture uncertainty is through a sampling loop that is embedded within the optimization iterations for the decision variables as shown in Fig. 1.4. This step can be computationally expensive as the model has to be rerun for every sample point. Therefore, we consider efficient sampling techniques. These techniques are described in Chap. 2.

Stochastic approximation methods use upper and/or lower bounds for expected function in a two-stage SP problems. For specific structures (e.g., dual block angular structure) where the L-shaped method is applicable, two approaches consider bounding approximations by embedding sampling within another algorithm without complete optimization. These two approaches are the method of Dantzig [5] , which uses importance sampling to reduce variance in each cut based on a large sample, and the stochastic decomposition method proposed by [20] , which uses the lower bound of the recourse function based on expectation. Again these methods exploit specific structures of the problem and require convexity conditions.

In the BONUS algorithm presented in this book, a generalized approach is proposed that can be used for real world large scale systems without any assumptions. The approach takes advantage of traditional NLP methods such as sequential quadratic programming (SQP) or generalized reduced gradient (GRG2) method that are based on derivative estimation. For real world large scale systems, perturbation derivatives are commonly employed. In order to use derivative-based methods, we need smooth probability density functions; Chap. 3 presents the kernel density estimation (KDE) approach for smooth probability density functions.

Optimization under uncertainty involves iteratively improving a probabilistic objective function. Figure 1.6a shows uncertain input variables with an underlying probability density function $pdf_{in}(x^k, u^k)$, shown as the solid triangular distribution.

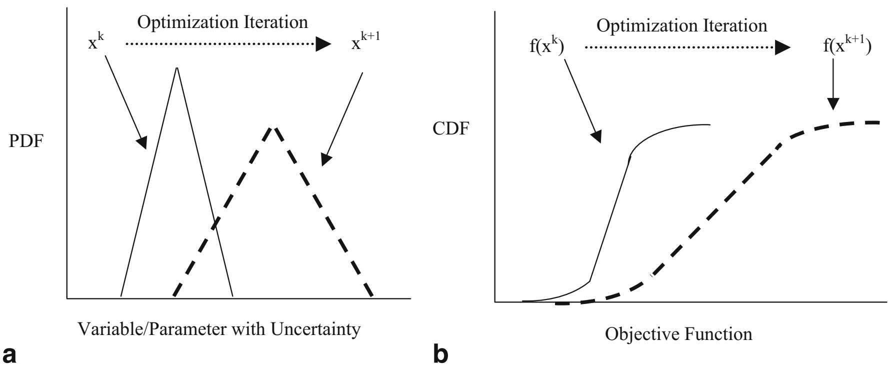

Fig. 1.6 Effect of changes in decision variables

After the model is run, the corresponding output distribution $cdf_{out}(x^k, u^k)$ is generated, shown as the solid line in Fig. 1.6b. As optimization progresses to the next iteration, $k + 1$, moments such as mean, variance, and the probability function can change for the uncertain variables, resulting in a new $\widehat{pdf_{in}}(x^{k+1}, u^{k+1})$, indicated by the dashed line in Fig. 1.6a .

The goal is to identify rapid and efficient techniques that determine an approximation of the properties of the new output distribution, $\widetilde{cdf_{out}}(x^{k+1}, u^{k+1})$, given as the dashed cumulative distribution function in Fig. 1.6b. The advantage of this approach is its bypassing of the model evaluations for successive sampling (the inner loop in Fig. 1.4), which is computationally the most intensive task for optimization under uncertainty. The BONUS algorithm only uses sampling for the first iteration. Details of the algorithm are described in Chap. 4. This is followed by three chapters on application of the BONUS to real world systems. Chapter 8 presents a variant of the BONUS algorithm called L-Shaped BONUS that exploits specific structure problems. Applications of this variant are presented in the Chaps. 9 and 10.

1.3 Summary

A generalized way of solving SNLP is to use sampling-based methods. BONUS exploits the advantages of traditional NLP methods based on derivative information. It uses efficient sampling techniques and uses sampling only for the first iteration. A variant of the the BONUS algorithm, namely, the L-shaped BONUS algorithm uses specific structure of the problem for efficiency improvement. The BONUS algorithm and its variant allows for solution of large scale real world problems.