Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 801734, 9 pages
http://dx.doi.org/10.1155/2013/801734

Research Article

Pareto Optimal Solutions for Stochastic Dynamic Programming Problems via Monte Carlo Simulation

R. T. N. Cardoso,[1] R. H. C. Takahashi,[2] and F. R. B. Cruz[3]

[1] *Departamento de Física e Matemática, Centro Federal de Educaçäo Tecnológica de Minas Gerais, 30510-000 Belo Horizonte, MG, Brazil*
[2] *Departamento de Matemática, Universidade Federal de Minas Gerais, 31270-901 Belo Horizonte, MG, Brazil*
[3] *Departamento de Estatística, Universidade Federal de Minas Gerais, 31270-901 Belo Horizonte, MG, Brazil*

Correspondence should be addressed to R. T. N. Cardoso; rodrigoc@des.cefetmg.br

Received 30 May 2013; Accepted 28 September 2013

Academic Editor: Lotfollah Najjar

A heuristic algorithm is proposed for a class of stochastic discrete-time continuous-variable dynamic programming problems submitted to non-Gaussian disturbances. Instead of using the expected values of the objective function, the randomness nature of the decision variables is kept along the process, while Pareto fronts weighted by all quantiles of the objective function are determined. Thus, decision makers are able to choose any quantile they wish. This new idea is carried out by using Monte Carlo simulations embedded in an approximate algorithm proposed to deterministic dynamic programming problems. The new method is tested in instances of the classical inventory control problem. The results obtained attest for the efficiency and efficacy of the algorithm in solving these important stochastic optimization problems.

1. Introduction

Since Bellman [1] proposed the dynamic programming technique for multistage optimization problems, his method has been very successfully applied to a wide range of problems, as it is well documented (see, for instance, [1–5]). This paper considers a dynamic programming-based method to solve stochastic discrete-time continuous-variable dynamic programming problems. A standard formulation of this problem is given by

$$\min_{u[0],\ldots,u[n-1]} E\left[\sum_{k=0}^{n-1} g_k\left(x\left[k\right], u\left[k\right]\right) + g_n\left(x\left[n\right]\right)\right], \qquad (1)$$

subject to

$$x\left[k+1\right] = f\left(x\left[k\right], u\left[k\right], w\left[k\right]\right), \quad k = 0, 1, \ldots, n-1,$$
$$x\left[0\right] = x_0, \qquad x\left[n\right] = x^*, \qquad (2)$$

in which $k = 0, 1, \ldots, n$ are the time stages, $x[0], \ldots, x[n]$ are the state variables, $u[0], \ldots, x[n-1]$ are the decision variables, $w[0], \ldots, w[n-1]$ are the disturbance variables, $f(x[k], u[k], w[k])$ is the state equation, $g_k(x[k], u[k])$ is the cost associated to stage k, $g_n(x[n])$ is the cost associated to the final stage, n, and $E[X]$ is the expectation of random variable X.

The state variables track the system dynamics throughout the states, and the decision variables are actions that should be taken in order to achieve the optimization objective. Additionally, the disturbance variables are assumed as random variables, from any given distribution. The objective function is assumed separable in all variables and stages.

In discrete-variable dynamic programming problems, a resolution algorithm might involve a sequential search in a graph, the nodes being the admissible states, the arcs being the possible control actions, and the arc costs being the respective transition probabilities. Although accurate, such an algorithm certainly would have a tremendous computational cost, suffering from what is known as the "curse of dimensionality." That is, the processing costs would increase exponentially with the number of possible states (for details, see [1, 2, 6]).

In continuous-variable dynamic programming problems, an exact solution can be obtained only when the system dynamics is linear and the objective function is quadratic. In the general case, only approximate solutions are possible, which are usually based on a state space discretization, such as a uniform grid discretization of points over the state space. However, this is also not an efficient procedure due to the exponential growth of the number of states. In order to cope with this problem, computationally efficient approximate discretization procedures have been tried out already (see, for instance, [7, 8]).

In this paper, we propose numerical heuristic solutions, coupled with Monte Carlo simulations, obtainable within a quite reasonable amount of computational effort. Monte Carlo simulation is a well-known and useful method to determine probabilities by using highly intensive computational experiments [9]. Solving a dynamic programming problem by simulation is not a novelty. However, while widely used in many contexts, Monte Carlo simulations have not been used often in solving dynamic programming problems, perhaps because of their high computational costs. Just recently, now coupled with heuristics and approximations, Monte Carlo simulations started to be considered again. Its prohibitive computational costs were exchanged by solutions without strict guarantee of optimality.

Indeed, neurodynamic programming is a well-known dynamic programming approach that employs Monte Carlo sampling in stochastic settings [10]. Another very successful example is reported by de Farias and van Roy [11], which reformulated the stochastic dynamic programming problem as a linear programming problem and approximated the large resulting number of constraints by Monte Carlo sampling. Also Thompson and Cluett [12] considered Monte Carlo simulation to calculate integrals related to the expected value of the objective function of a unidimensional dual-optimal control problem that has to be decided by iterative dynamic programming [13]. A number-theoretic method, called quasi-Monte Carlo, uses number theory and numerical analysis to generate a point set that is uniformly spaced. This technique has been successfully used by Cervellera et al. [7, 8], along with neural networks, to improve a computationally tractable approximate method of discretization. Those are few examples of well-succeeded reductions in the "curse of dimensionality."

For most of the stochastic dynamic optimization problems considered in the literature, a usual model for disturbances is a zero mean and σ^2 variance Gaussian white noise, although in some cases a nonzero mean may also appear [14]. These assumptions can be suitable for some real-world applications but it certainly will not be the case for many of them. The focus of this paper is stochastic discrete-time continuous-variable dynamic programming problems submitted to non-Gaussian probability distribution functions for disturbances $w[k]$. In those problems, the analytical form of the expected value of the objective function may be really complex or even lead to solutions impossible to be found by means of classic dynamic programming techniques. If disturbances $w[k]$ are Gaussian, it is possible that the best optimization reference is the expected value of the objective function. However, in many practical applications, mainly under non-Gaussian disturbances, it is almost certain that other quantiles, greater or smaller than the median, are better, as we will show shortly. Thus, a multiobjective approach seems to be quite justified [15]. That is, instead of finding the control sequence that optimizes the expected value of the objective function, this paper proposes finding Pareto fronts instead, weighted by all quantiles of the objective function.

In multiobjective optimization problems [16], there may not be a single solution that is the best (or the global optimum) with respect to all the objectives. The presence of multiple objectives in a problem usually gives rise to a family of nondominated solutions, called Pareto optimal set, in which each objective can only be improved by degrading at least one of the other objectives. Since none of the solutions in the nondominated set is absolutely better than the others, any one of them is an acceptable solution. For instance, Li and Haimes [17] presented a survey on multiobjective dynamic programming, and Trzaskalik and Sitarz [18] proposed a procedure that considers a partially ordered criteria structure in dynamic programming. However, we remind that the approach proposed here is out of the traditional multiobjective discrete-time dynamic programming methods.

A highlight of the contributions of this paper is:

(i) We propose a scheme based on Monte Carlo simulations coupled with a deterministic discrete-time dynamic programming method, which is exact for continuous-variable problems and asymptotically accurate for discrete-variable problems. Details will be given shortly, but the main idea of this dynamic programming method is to study the system dynamics as an iteration operated on closed sets, essentially considering the problem from a geometrical point of view, instead of using a more traditional way of studying it in the arcs of the graph. Since the scheme proposed is based on simulations, it will be possible to use non-Gaussian probability distribution functions for disturbances.

(ii) Using the proposed scheme, a multiobjective approach is employed, because why should one ever consider only the expected value as the reference for the optimization when it is possible to take into account all quantiles? To deal with this point, a Pareto front is presented as the answer of the problem, weighted by a function of the empirical quantiles of the decision variables. This is true because quantiles are functions of the random variables, which can be sampled by Monte Carlo simulations.

The proposed methodology and the algorithm are explained in detail in Section 2. A case study based on a classical inventory control problem is conducted and presented in Section 3. Finally, in Section 4, we summarize this paper and conclude it with topics for future research in the area.

```
algorithm
    read input;
    for $i = 1$ until sampleSize do
        generate randomly a sequence of disturbances with a
            given probability distribution function;
        find a sequence of decision variables that optimizes
            the objective-function, as if it were a
            deterministic dynamic programming problem;
    end for
    mount the Pareto front of the decision variables,
        weighted by its quantiles;
    take the box-plot, the average, or any other quantile of
        these variables as the answer of the problem.
end algorithm
```

Algorithm 1

2. Proposed Approach

We propose here a computationally tractable scheme for stochastic dynamic programming problems formulated as in (1). Our heuristic method consists of a multiobjective approach based on Monte Carlo simulations embedded in a deterministic dynamic programming method. For our convenience, the deterministic method chosen here was a geometrical algorithm, described in detail in Cardoso [19] and Cardoso et al. [15], which is exact for continuous-variable problems and asymptotically exact for discrete-variable problems.

In the geometrical algorithm described by Cardoso [19], the system dynamics is supposed to be linear, with real variables, or else with integer variables that could be relaxed. The objective functions could be of any type, not necessarily quadratic, although quadratics are the most used functions in the literature. The geometrical algorithm is inspired by approximate dynamic programming methods, namely, the certainty equivalent control technique and the model predictive control technique (details may be found in [2]). An open-loop optimal control computation is used in conjunction with a rolling horizon, which means that more than one control move is generally calculated. Finally, we remark that the scheme is flexible enough to work under any other fast deterministic dynamic programming method as well. The proposed algorithm may be summarized as shown in Algorithm 1.

We remind that because the proposed procedure is based on simulations, it would be no trouble to consider disturbances from any probability distribution, including those for which an analytical solution is difficult, impossible, or mathematically intractable. For example, it is possible to consider multivariate-Gaussian distributions with different means and variances per coordinate, or even multimodal and asymmetrical distributions. This fact is very convenient in accurately representing real-life phenomena. The multiobjective approach explicitly considers the random distribution of the control variables and implicitly takes into account the random distribution of the state variables and of the objective functions. In other words, all variables are to be treated as random, which is possible because they are functions of the disturbance random variables. As a result, we can find box-plots, histograms, and quantiles for all decision variables, as outcomes of the proposed method for the problem at hand.

As a final remark, according to the formulation presented in (1), Monte Carlo simulations should have been used to compute the sequence of control variables to optimize the expected value of the objective function for several disturbances randomly generated. Instead, the proposed algorithm presents the expected value of a sequence of decision variables to optimize the objective function, for several randomly generated disturbances. In the appendix, we present a theorem that shows that (i) the result found by the proposed methodology is a lower bound for a classical solution method and (ii) the equality is valid if the decision variable minimizes the objective function for almost all disturbances. Under a practical point of view, the equality occurs, for example, if the decision variables are constrained to compact sets, as usually happens in linear problems that are solved by numerical methods. Finally, we remind that the convergence of this algorithm relies on the convergence of Monte Carlo simulations [9].

3. Simulations

3.1. Preliminaries. The algorithm was implemented in MATLAB (MATLAB is a trademark of The MathWorks, Inc.), and all cases were run in a common PC. For all simulations, 5,000 disturbance vectors $w[k]$ were generated. However, with only 100 vectors $w[k]$, convergence could be verified [20]. The purpose of these case studies is just to show a simple and yet interesting application of the proposed methodology. Then, a classical stochastic dynamic programming example is considered, namely, the inventory control. Details may be found in Bertsekas [2], but in few words the problem consists in determining optimal orders to be placed for some items at discrete-time stages so as to meet a stochastic demand. Moreover, it is requested that the available stock at the final stage is null. For such a problem, the variables are inherently discrete since items are counted, but the range of levels for

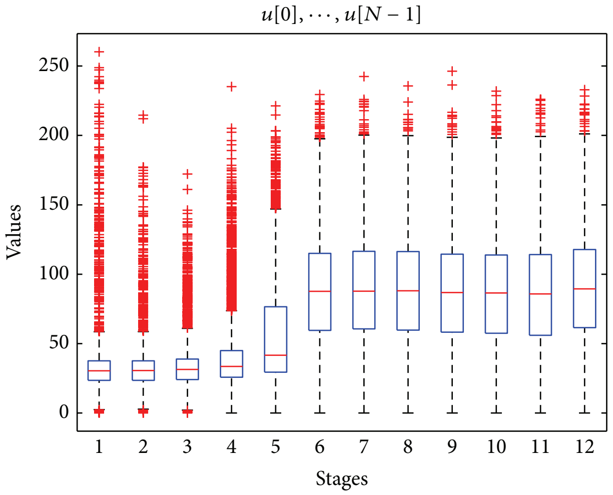

Figure 1: Case 1, box-plots for first product optimal orders.

an item is too large to be practical for a discrete-variable dynamic programming solution. Then, the discrete variables will be relaxed and they will be considered as real numbers. The initial stock will also be a decision variable. This dynamic optimization problem can be formulated as

$$\min_{u[0],\ldots,u[n-1]} E\left[\sum_{k=0}^{n-1}\left(c_k x[k] + d_k u[k]\right) + c_n x[n]\right], \qquad (3)$$

subject to

$$\begin{gathered} x[k+1] = x[k] + u[k] - w[k], \qquad k = 0, 1, \ldots, n-1, \\ x[k] \geq 0, \qquad u[k] \geq 0, \qquad (4) \\ x[0] = x_0, \qquad x[n] = x^*, \end{gathered}$$

in which each stage k corresponds to each month, n is the horizon, each state vector $x[k]$ represents the inventory level (the stock available) at the beginning of stage k, each control action vector $u[k]$ is the amount ordered at the beginning of stage k, and each disturbance vector $w[k]$ is a stochastic customer demand during stage k, given from some probability distribution, each vector having size m corresponding to the number of products considered.

The inventory level evolves according to the linear discrete-time dynamic system:

$$x[k+1] = x[k] + u[k] - w[k]. \qquad (5)$$

A linear cost function per stage is composed by a penalty for each positive stock, represented by a row per unit cost vector c_k, added to the purchasing cost, represented by a row per unit cost vector d_k.

We consider the following three instances of the stochastic inventory control problem, all of them having an optimization horizon $n = 12$, vectors c_k and d_k unitary, and programming goal x^* equal to the null vector. The decision criterion of the multiobjective approach is the trade-off between the probabilities of having some stock at the end and the probability of not attending some demand. The instances are as follows.

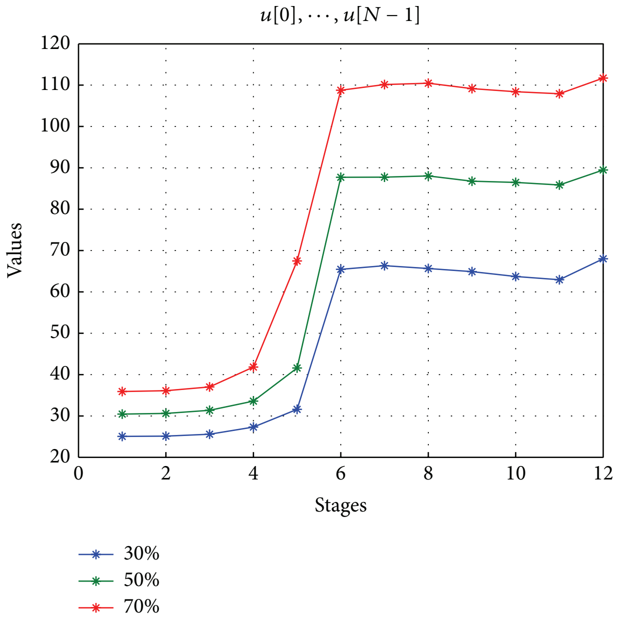

Figure 2: Case 1, band of values for first product optimal orders.

Case 1. Customer demands follow a Gaussian probability distribution, with multiproducts and a coupling constraint. Case 1 is composed by $m = 3$ products. Demands follow Gaussian distributions, with mean 30 and standard deviation 10, for first product, mean 60 and standard deviation 10, for second, and mean 30 and standard deviation 20, for third. The coupling constraint says that 2 units of product 1 and 1 unit of product 2 are necessary to produce each unit of product 3.

Case 2. Customer demands follow a bimodal probability distribution function. Case 2 is composed by just $n = 1$ product, with demands following a mix of Gaussian distributions having mean 100 and standard deviation 10, with probability 0.3, and mean 200 and same standard deviation, with probability 0.7.

Case 3. Customer demands follow an asymmetric probability distribution function. Also, only $n = 1$ product is considered, with demand that follows a zero-truncated log-normal distribution, with mean 2 and standard deviation 1.

3.2. Results for Case 1.

For Case 1, described earlier, the simulation results are presented in Figures 1–5. Only results for the first product are shown, as similar outputs (not presented) were obtained for the other two products. Figure 1 shows box-plots for the first product optimal order, and Figure 2 presents the corresponding band of values, between quantiles 0.3 and 0.7. As expected, it is noticeable here an increase in the uncertainty of the orders, what is evident from

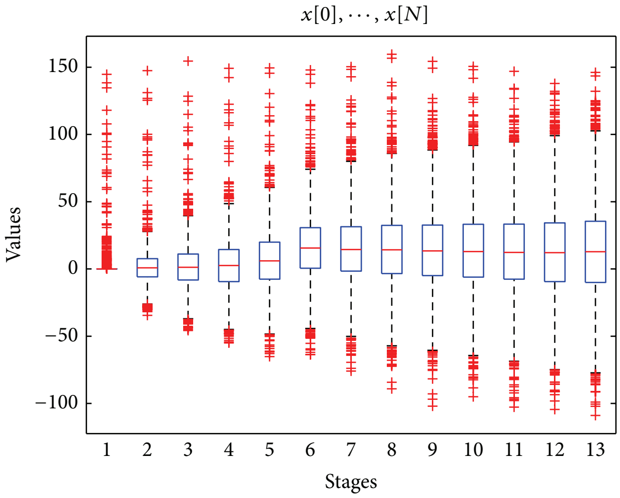

FIGURE 3: Case 1, box-plots for first product inventory levels when ordering by the median.

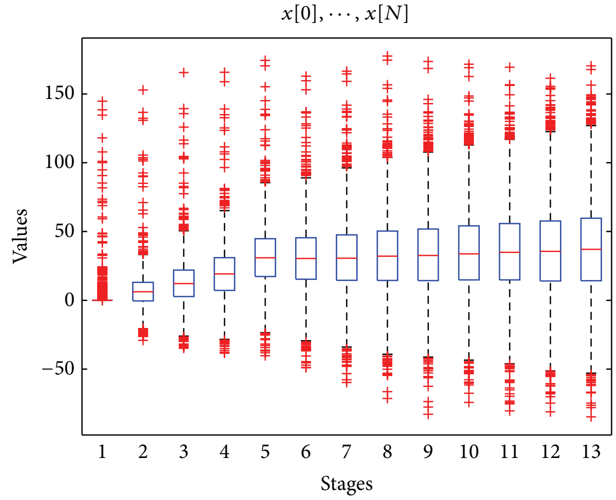

FIGURE 4: Case 1, box-plots for first product inventory levels when ordering by 70th percentile.

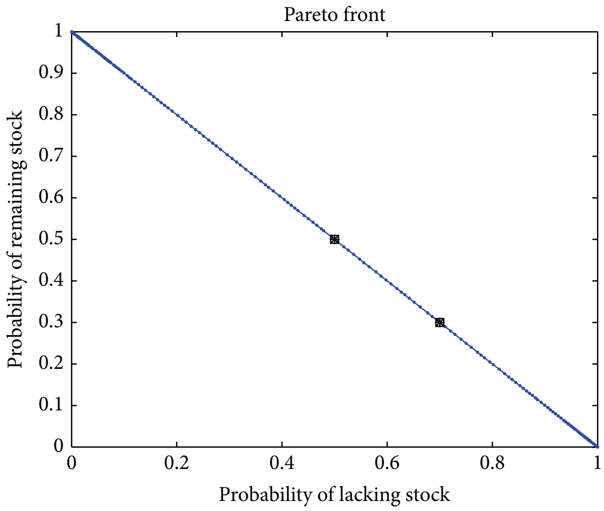

FIGURE 5: Case 1, Pareto front for a trade-off between the probabilities of having and not having stock.

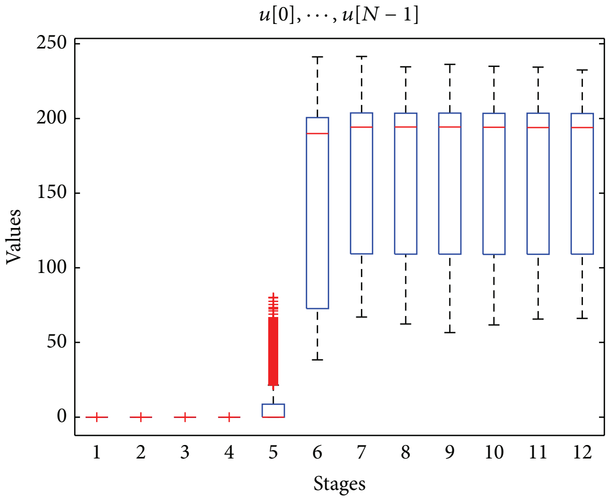

FIGURE 6: Case 2, box-plots of optimal orders.

the band enlargement that is observed as time goes by. Figures 3 and 4 illustrate what is expected to happen to the inventory level, when the orders are made by the median values and by quantile 0.7, respectively. These are values easily obtainable from the algorithm. From Figure 3, we can roughly estimate that 50%-50% are the probabilities of having and not having stock by the end of the planning horizon, which is expected because of the probability distribution of the demands being symmetrical. We see here a reduction in the probability of a negative stock, in case product orders would be placed by quantile 0.7, Figure 4. Actually, the decision maker could play with these two conflicting objectives, namely, (i) a low probability of having stock at the end of the horizon period and (ii) a low probability of not having stock at all, which would mean that some demand would not be attended. Figure 5 presents the respective Pareto front that resulted from our simulations.

3.3. *Results for Case 2.*

Results for Case 2 are depicted in Figures 6–10. The box-plots for optimal orders and respective band of values are presented in Figures 6 and 7, which follow basically the same pattern observed for Case 1; that is, we observe an increase in the width (uncertainty) of the band of values to be ordered along the time. Comparing two strategies of placing orders, that is, from the median values or else from percentile 0.7, leads to the results presented in Figures 8 and 9, from which we can notice that, by the final stage, the probabilities of having and not having stock are not 50%-50% anymore, for orders taken from median values. This results from the asymmetry of demands. On the other hand, ordering from quantile 0.7 will increase significantly the probability of having stock at the final stage. Of course, we could also estimate these probabilities for any ordering strategy we wish, as seen in Figure 10. As a final remark, it can be seen in this case that bimodality in demands creates

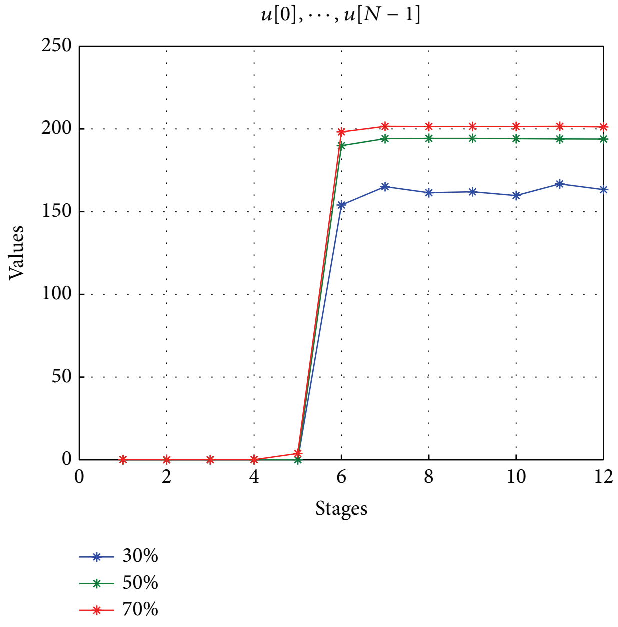

Figure 7: Case 2, band of values for optimal orders.

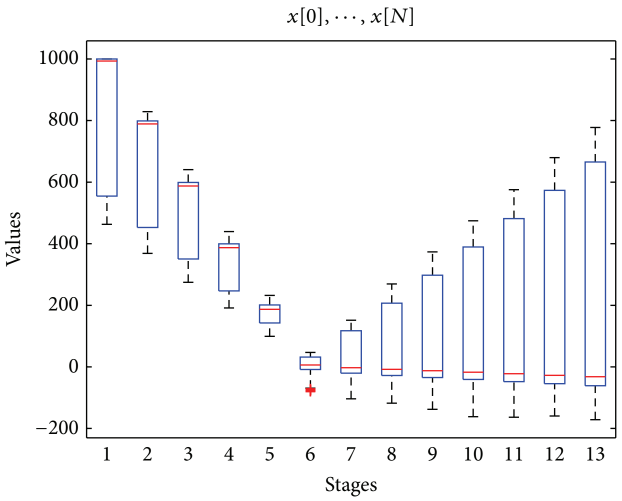

Figure 8: Case 2, box-plots for inventory levels when ordering by the median.

some difficulty for traditional approaches for planning order placements, as a median-based order no longer guarantees 50%-50% probabilities of having and not having stock by the end of the stages.

3.4. Results for Case 3. Figures 11, 12, 13, 14, and 15 show the results for Case 3, which accounts for asymmetrical distributed demands. In this case, orders taken by the median lead to approximately 82% of probability of having stock at the final stage and, of course, 18% of probability of not having stock at all at the final stage. On the other hand, orders taken by quantile 0.7 lead to approximately 37% of probability of having stock at the end of stages and 63% of not having any stock at the end. By means of our methodology, it would be easy to identify, for instance, that the target of 50%-50% would be reached if orders were to be placed from quantile ≈ 0.64.

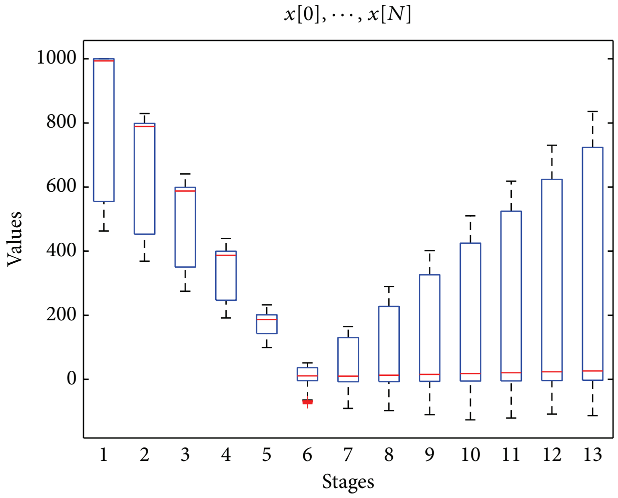

Figure 9: Case 2, box-plots for inventory levels when ordering by 70th percentile.

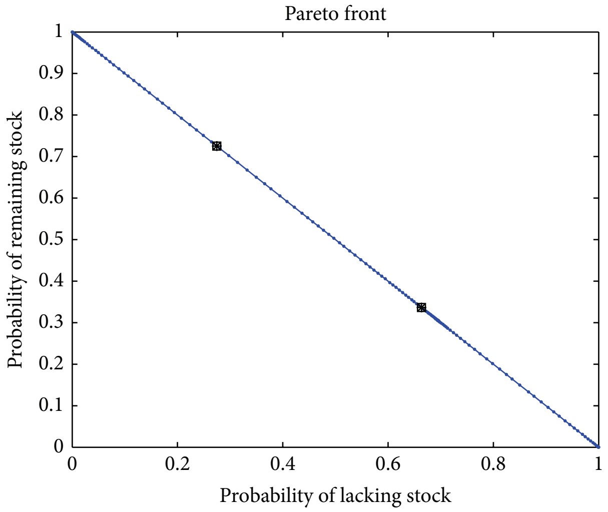

Figure 10: Case 2, Pareto front for a trade-off between the probabilities of having and not having stock.

4. Discussions and Conclusions

In this paper, we proposed a multiobjective approach for stochastic discrete-time real-variable dynamic programming problems, which is based on Monte Carlo simulations coupled with a deterministic dynamic programming algorithm. Our approach was shown to be able to deliver suboptimal (heuristic) solutions by using common desktop computers, within a reasonable amount of computational time. Once a Pareto front for the problem becomes built, decision makers can choose any quantile that is perceived to be advantageous for the specific situation. In addition, a band of values,

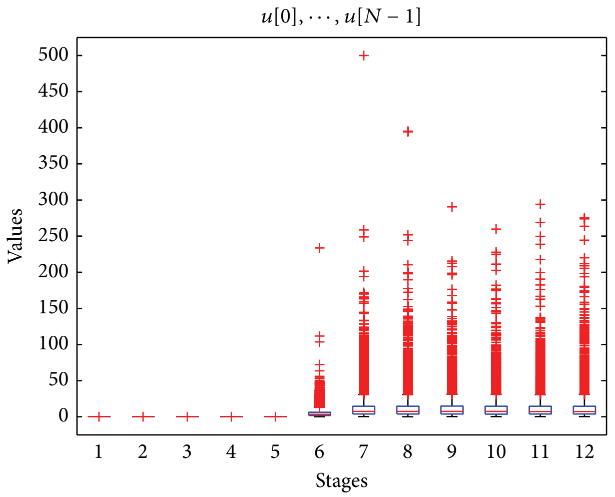

FIGURE 11: Case 3, box-plots of the optimal orders.

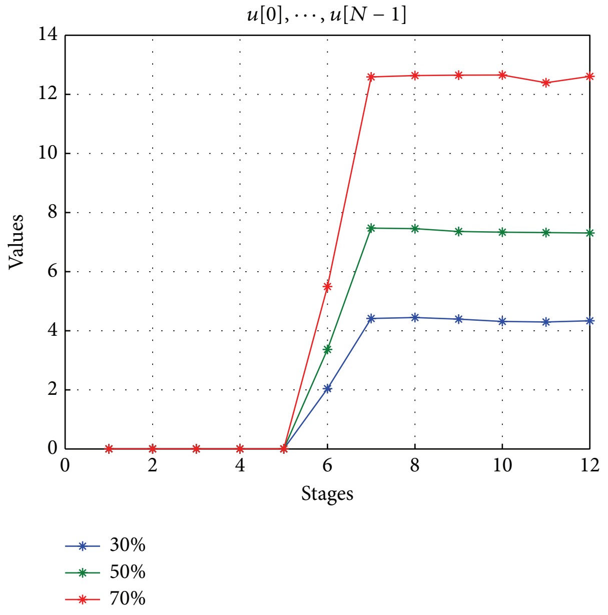

FIGURE 12: Case 3, band of values for optimal orders.

histograms of the decision variables, and their box-plots are provided as outcomes of the proposed method for a given problem, instead of just the decision variable values that optimize the expected value of the objective function. For a more effective understanding of the method proposed and the results delivered, we presented a case study based on a classical inventory control problem. The numerical results obtained in the simulations showed that Monte Carlo simulations were quite effective in solving realistic cases and that the methodology is a promising technique.

Future research will investigate the application of the proposed methodology in instances of real-world dynamic problems larger than those ones treated here. Additionally, more realistic and complex modeling of the random variables

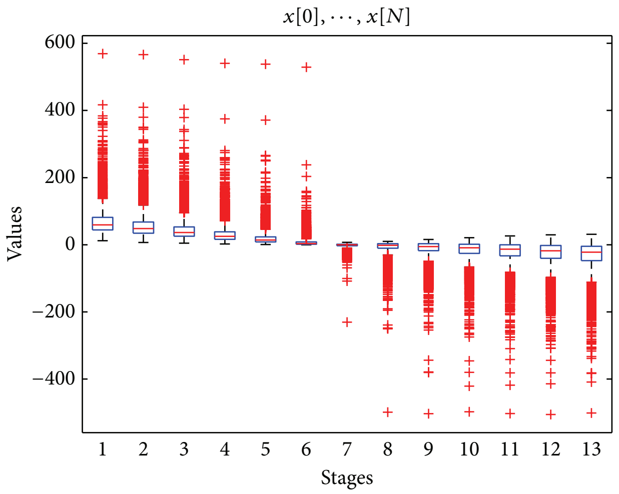

FIGURE 13: Case 3, box-plots for inventory levels when ordering by the median.

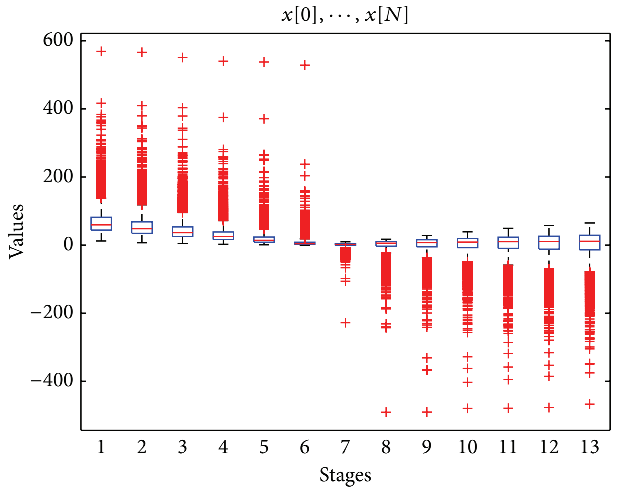

FIGURE 14: Case 3, box-plots for inventory levels when ordering by 70th percentile.

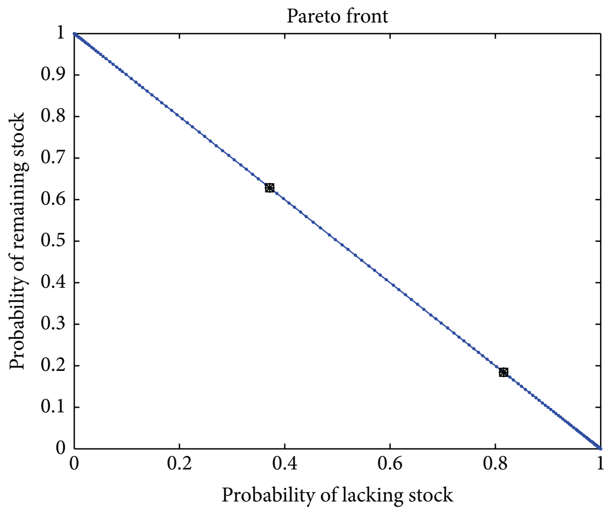

FIGURE 15: Case 3, Pareto front for trade-off between the probabilities of having and not having stock.

will be considered, for instance, with variable-dependency. These are some few topics for future research in this area.

Appendix

This Appendix presents a theorem that validates the proposed algorithm. The random variables are considered to be continuous but the discrete variable case could be treated similarly. The improper integral, which comes from the expected value of the continuous random variable, is supposed to converge. If a minimum does not exist, just replace it by the infimum. All minimization could be for almost all w instead of for all w. The theorem is based on a review of the Fatou Lemma, as follows.

Lemma A.1 (review of Fatou Lemma). *Let $f : \mathbb{R}^2 \to \mathbb{R}$ be an integrable function in the first coordinate and with minimum in the second one. Thus,*

$$\int \min_u f(w,u)\,dw \le \min_u \int f(w,u)\,dw. \tag{A.1}$$

Proof. For each fixed w, consider

$$\overline{u}(w) = \arg\min_u f(w,u). \tag{A.2}$$

Then, one has that

$$f(w,\overline{u}(w)) \le f(w,u), \quad \forall u. \tag{A.3}$$

As the integral operator preserves monotonicity

$$\begin{aligned}\int \min_u f(w,u)\,dw &= \int f(w,\overline{u}(w))\,dw \\ &\le \int f(w,u)\,dw, \quad \forall u.\end{aligned} \tag{A.4}$$

As this expression holds for all u, it holds for

$$u_o = \arg\min_u \int f(w,u)\,dw. \tag{A.5}$$

Thus,

$$\int \min_u f(w,u)\,dw \le \min_u \int f(w,u)\,dw. \tag{A.6}$$

□

In other words, the expected value of the minimum is always a lower bound for the minimum of the expected value.

Theorem A.2 (validation of the algorithm). *Let $f : \mathbb{R}^2 \to \mathbb{R}$ be an integrable function in the first coordinate and with minimum in the second one. If there exists $\overline{u}$ that does not depend on w, such that*

$$f(w,\overline{u}) = \min_u f(w,u), \quad \forall w, \tag{A.7}$$

the following equality holds:

$$\int \min_u f(w,u)\,dw = \min_u \int f(w,u)\,dw. \tag{A.8}$$

Proof. (⇒) Suppose that there exists

$$\overline{u} = \arg\min_u f(w,u), \quad \forall w. \tag{A.9}$$

Consider

$$u_o = \arg\min_u \int f(w,u)\,dw. \tag{A.10}$$

By Lemma A.1,

$$\begin{aligned}\int f(w,\overline{u})\,dx &= \int \min_u f(w,u)\,dw \\ &\le \min_u \int f(w,u)\,dw = \int f(w,u_o)\,dw.\end{aligned} \tag{A.11}$$

As u_o minimizes the integral, it is true that

$$\int f(w,u_o)\,dw \le \int f(w,u)\,dw, \quad \forall u. \tag{A.12}$$

In particular, for $\overline{u}$, which is fixed and minimizes f, for all w

$$\int f(w,u_o)\,dw \le \int f(w,\overline{u})\,dw. \tag{A.13}$$

Therefore, the following equality holds:

$$\int f(w,u_o)\,dw = \int f(w,\overline{u})\,dw. \tag{A.14}$$

In other words,

$$\int \min_u f(w,u)\,dw = \min_u \int f(w,u)\,dw. \tag{A.15}$$

Thus, if minimums exist, they must coincide

$$u_o = \arg\min_u \int f(w,u)\,dw = \overline{u} = \arg\min_u f(w,u). \tag{A.16}$$

(⇐) Suppose that, for all u_o given, there exists a set $\overline{W}$ with positive measure, for which

$$\begin{aligned}u_o &\neq \arg\min_u f(\overline{w},u), \quad \forall \overline{w} \in \overline{W} \Longrightarrow \\ u_o &\neq \arg\min_u f(w,u), \quad \forall w.\end{aligned} \tag{A.17}$$

In other words, $\forall \overline{w} \in \overline{W}$, there exists a $\overline{u}$ which depends on $\overline{w}$, such that

$$\overline{u}(\overline{w}) = \arg\min_u f(\overline{w},u). \tag{A.18}$$

Therefore,

$$f(\overline{w},\overline{u}(\overline{w})) < f(\overline{w},u_o), \quad \forall \overline{w} \in \overline{W}. \tag{A.19}$$

By the monotonicity of the integral

$$\begin{aligned}\int_{\overline{W}} f(w,\overline{u})\,dw &= \int_{\overline{W}} \min_u f(w,u)\,dw \\ &< \int_{\overline{W}} f(w,u_o)\,dw, \quad \forall u_o \Longrightarrow\end{aligned} \tag{A.20}$$

$$\int \min_u f(w,u)\,dw < \int f(w,u_o)\,dw, \quad \forall u_o.$$

In particular,

$$u_o = \arg\min_{u} \int f(w, u)\, dw. \qquad (A.21)$$

Thus, the inequality just holds:

$$\int \min_{u} f(w, u)\, dw < \min_{u} \int f(w, u)\, dw. \qquad (A.22)$$

□

Therefore, the equality is valid, for example, if f is linear and u is considered constrained to compact sets.

Acknowledgments

The authors would like to thank the Brazilian agencies, CAPES, CNPq, and FAPEMIG for the financial support.

References

[1] R. Bellman, *Dynamic Programming*, Dover Books on Computer Science Series, Dover, Mineola, NY, USA, 2003.

[2] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 3, Athena Scientific, Belmont, Mass, USA, 3rd edition, 2007.

[3] B. H. Dias, A. L. M. Marcato, R. C. Souza et al., "Stochastic dynamic programming applied to hydrothermal power systems operation planning based on the convex hull algorithm," *Mathematical Problems in Engineering*, vol. 2010, Article ID 390940, 20 pages, 2010.

[4] Y. Zhang, S. Song, C. Wu, and W. Yin, "Dynamic programming and heuristic for stochastic uncapacitated lot-sizing problems with incremental quantity discount," *Mathematical Problems in Engineering*, vol. 2012, Article ID 582323, 21 pages, 2012.

[5] S. Ji, C. Sun, and Q. Wei, "The dynamic programming method of stochastic differential game for functional forward-backward stochastic system," *Mathematical Problems in Engineering*, vol. 2013, Article ID 958920, 14 pages, 2013.

[6] D. P. Bertsekas, *Dynamic Programming: Deterministic and Stochastic Models*, Prentice Hall, New York, NY, USA, 1987.

[7] C. Cervellera, V. C. P. Chen, and A. Wen, "Optimization of a large-scale water reservoir network by stochastic dynamic programming with efficient state space discretization," *European Journal of Operational Research*, vol. 171, no. 3, pp. 1139–1151, 2006.

[8] C. Cervellera, A. Wen, and V. C. P. Chen, "Neural network and regression spline value function approximations for stochastic dynamic programming," *Computers & Operations Research*, vol. 34, no. 1, pp. 70–90, 2007.

[9] S. M. Ross, *Simulation*, Academic Press, San Diego, Calif, USA, 5th edition, 2012.

[10] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, Belmont, Mass, USA, 1996.

[11] D. P. de Farias and B. van Roy, "Approximate linear programming for average-cost dynamic programming," in *Advances in Neural Information Processing Systems 15*, S. T. Becker and K. Obermayer, Eds., pp. 1587–1594, The MIT Press, Cambridge, Mass, USA, 2003.

[12] A. M. Thompson and W. R. Cluett, "Stochastic iterative dynamic programming: a Monte Carlo approach to dual control," *Automatica*, vol. 41, no. 5, pp. 767–778, 2005.

[13] R. Luus, *Iterative Dynamic Programming*, Chapman & Hall/CRC, New York, NY, USA, 2000.

[14] H. Kushner, *Introduction to Stochastic Control*, Holt, Rinehart and Winston, 1971.

[15] R. T. N. Cardoso, R. H. C. Takahashi, F. R. B. Cruz, and C. M. Fonseca, "A multi-quantile approach for openloop stochastic dynamic programming problems," in *Proceedings of the IFAC Workshop on Control Applications of Optimization (CAO '09)*, vol. 7, pp. 52–57, 2009, http://www.scopus.com/inward/record.url?eid=2-s2.0-79960935120&partnerID=40&md5=7c3ed430fc2a0c425d0c182ac0ecf0d8.

[16] V. Chankong and Y. Y. Haimes, *Multiobjective Decision Making: Theory and Methodology*, Dover, Mineola, NY, USA, 2008.

[17] D. Li and Y. Y. Haimes, "Multiobjective dynamic programming. The state of the art," *Control, Theory and Advanced Technology*, vol. 5, no. 4, pp. 471–483, 1989.

[18] T. Trzaskalik and S. Sitarz, "Discrete dynamic programming with outcomes in random variable structures," *European Journal of Operational Research*, vol. 177, no. 3, pp. 1535–1548, 2007.

[19] R. T. N. Cardoso, *Algorithms for dynamic programming based on invariant families [M.S. thesis]*, Departamento de Matemática, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil, 2005, (Portuguese), http://www.mat.ufmg.br/intranet-atual/pgmat/TesesDissertacoes/uploaded/Diss109.pdf.

[20] S. Robinson, "A statistical process control approach to selecting a warm-up period for a discrete-event simulation," *European Journal of Operational Research*, vol. 176, no. 1, pp. 332–346, 2007.