Computers and Operations Research 92 (2018) 98–110

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

A two-step gradient estimation approach for setting supply chain operating parameters

Philip M. Kaminsky[a,*], Stewart Liu[a,b]

[a] Department of Industrial Engineering and Operations Research, University of California, Berkeley, USA
[b] Department of Decision Sciences, San Francisco State University, USA

ARTICLE INFO

Article history:
Received 10 March 2017
Revised 18 November 2017
Accepted 1 December 2017
Available online 12 December 2017

Keywords:
Supply chain disruption
Retrospective optimization
Sample path optimization
Stochastic gradient search

ABSTRACT

In earlier work, we found retrospective optimization to be effective for setting policy parameters in supply chains with relatively simple structures. This method finds these parameters by solving an integer program over a single randomly generated sample path. Initial efforts to extend this methodology to more complex settings were in many cases too slow to be effective. In response to this, in this research we combine retrospective optimization over a relatively short time horizon with stochastic approximation gradient search algorithms, an approach that proves to be fast and effective. We compare this approach to retrospective optimization without gradient search on simple serial supply chains where the solution is known, and then use it for effective inventory positioning in more complex biopharmaceutical supply chains.

1. Introduction

Firms often face the need to quickly and efficiently set supply chain operating parameters in complex, stochastic supply chains. It turns out that many of these problems share a specific structure: *given a sample path of realizations of the random variables in the problem over some finite horizon*, optimal policy parameters *for that specific sample path and that horizon* can be found by solving a (potentially very large) mixed integer linear program (MILP). Given this insight, we have developed an efficient two-step heuristic approach for finding effective operating parameters for these large stochastic problems, which we call ROGS, based on a hybrid of integer-programming-based *Retrospective Optimization* and *Stochastic Gradient Search*. Although these two techniques have separately (and in a few cases, together) been the focus of much research over the past decades, this is the first work, as far as we know, to combine these algorithms with simulation in a way that is flexible, relatively easy to implement in many settings, and applicable to the complicated and generally non-convex models characterized above. Indeed we have had success using our approach to quickly and effectively set supply chain policy parameters in a number of settings for biopharmaceutical firms that are members of the NSF-supported Center for Excellence in Logistics and Distribution.

Firms in the biopharmaceutical industry have supply chains that involve multiple stochastic elements including random demand, yields, fermentation times, filtration times, quality control times, etc. In addition, these supply chains are subject to disruptions due to natural disasters and human errors. There are typically a large number of supply chain operating policy parameters to set in these supply chains, where, for instance, an order-up-to policy might dictate raising inventory to y at a specific site each period, or a periodic shipping policy might dictate shipping available inventory from one site to another every x days. Setting policy parameters (such as the x or y in the examples above) in this sort of complex, stochastic environment typically requires complicated mathematical models with solution approaches that are very specific to the exact structure of the model (particularly when a large number of system parameters need to be determined simultaneously.) Much to our surprise, Retrospective Optimization via integer programming proved effective for solving these real-world problems.

Retrospective Optimization (Healy and Schruben, 1991) is a type of sample path optimization (Robinson, 1996) commonly used for finding solutions to stochastic optimization problems. This approach is inspired by retrospective analysis, where managerial decisions are made through an exploration of what *would have been* the best decisions in the past based on subsequently realized data and performance (Healy and Schruben, 1991). To apply retrospective optimization, we simulate a potential future, and then determine the retrospective solution for this projected future. Specifically, given distributions for a problem's random variables, one

* Corresponding author.
E-mail address: kaminsky@berkeley.edu (P.M. Kaminsky).

https://doi.org/10.1016/j.cor.2017.12.001
0305-0548/

possible realization for all variables is generated, and the objective function is optimized given this now-deterministic sample path. In other words, we are solving the following problem: If we know the realization of future random variables, what is the most effective policy? In "retrospective" terms, suppose that we are at some point in the future having seen the value that the random system parameters have taken on. Given this scenario, what *would have been* the best policy to implement?

For the types of supply chain policy parameter problems we consider where it is reasonable to assume a linear cost function, given a sample path, parameters can typically be optimized using a mixed-integer program, where integer variables are necessary to model the implementation of *policies*. Recall we are optimizing policy parameters, and not decisions, and consider the example of setting inventory policy in a simple inventory system. If future demand is known and our objective is to minimize inventory costs by determining shipment quantities in each period, shipments would equal future demand, and solving this class of models would require solving linear programs in many cases. However, this approach would not give insight into how to actually manage the supply chain, because in reality, future demand is not known. However, if stationary policy parameters (such as base stock levels for example) are optimized, the resulting mathematical program is a MILP, and the resulting policy is *implementable* because it doesn't directly depend on specific demand realizations.

To illustrate this, consider the example of a simple single stage k-period finite horizon periodic review inventory model with deterministic lead time and random demand. Assuming no loss sales (i.e. backorder is allowed with a penalty cost), such system can be stochastically optimized under a basestock policy. However, with a demand sample path, d_t in period $t = 1, 2, 3, \ldots, k,$ this system operating under the basestock policy can also be modeled as a retrospective MILP.

We let π be the backorder cost and h be the unit holding cost. Let the variable s represent the basestock level, which we also assume to be the inventory level at the beginning of time. Let I_t be inventory at time t, T_t be demand met from inventory at time t, B_t be backorder at time t, and s be the stationary base stock level. Given a vector of demand $\bar{X}_k = [d_1, d_2, \ldots, d_k]$, the optimal basestock level, s^*, for that vector of demand can be found by solving the following integer program:

$P1(\bar{X}_k)$:

$$\min \quad \frac{1}{k}\sum_{t=1}^{k}\left[\pi B_t + hI_t\right]$$

$$s.t. \quad T_t = \min\{s, d_t + B_{t-1}\} \qquad t = 1, 2, \ldots, k \tag{1}$$

$$B_t = d_t + B_{t-1} - T_t \qquad t = 1, 2, \ldots, k \tag{2}$$

$$I_t = s - T_t \qquad t = 1, 2, \ldots, k \tag{3}$$

$$T_0 = 0, B_0 = 0, \tag{4}$$

$$T_t \geq 0, B_t \geq 0, I_t \geq 0 \qquad t = 1, 2, \ldots, k \tag{5}$$

recalling that the minimization constraint (1) can equivalently be formulated as a set of inequalities using a binary variable and a sufficiently large constant. (We detail this in Section 4.1.1.)

In the specific projects that motivated this work, our goal was to set stationary policy parameters in what is effectively an infinite horizon setting. We did this using retrospective optimization by solving the integer program resulting from generating a "sufficiently long" sample path. While we were able to generate effective policies by solving these integer programs for very long time horizons (many years of daily decision-making), these integer programs take a long time to solve (sometimes days), making sensitivity analysis challenging. To reduce solution times, we have had success combining this integer-programming-based retrospective optimization approach with an approach based on Stochastic Gradient Estimation and Search. In our supply chain problems, parameter decision variables take on continuous values. Thus, we attempt to set these parameters by assigning initial values, estimating gradients, and searching in improving directions. This approach can be used to obtain an approximation of a local optimum (Fu, 2006). However, the cost functions of supply chain problems such as the ones we are dealing with are in general not convex functions of policy parameter settings. Thus, we are motivated to utilize a two-stage approach for finding effective parameter settings: first, we solve a relatively computationally inexpensive retrospective optimization integer program to find a "ballpark solution" – a solution that we hope is sufficiently close to the optimal one that local search will prove effective. Then, we use this solution as the starting point for the gradient search. For the examples we have tried, this approach is be nearly as effective as solving the large, complex MILP's described above, but takes orders of magnitude less time, thus allowing us to apply this approach to even more complex supply chains, and to more easily support "what-if" analysis.

While there are no doubt other settings where retrospective optimization problems can be expressed as MILPs, and our approach may also work for those, we developed our approach of combining retrospective optimization MILP's with Stochastic Gradient Search, which we call ROGS, in the context of supply chain problems, and in this paper we explore this approach. Specifically, we present a series of computational experiments developed to better understand the performance of ROGS, and to refine the details of its implementation. In the next section, we review relevant literature. In Section 3, we present our two-step ROGS approach in more detail. In Section 4, we present experiments designed to validate the performance and accuracy of ROGS under different operational parameters. Finally, in Section 6, we explore additional potential applications and extensions of ROGS.

2. Literature review

There are a variety of relevant streams of literature in simulation optimization and supply chain inventory optimization. Below, we consider approaches for optimizing stochastic objective functions based on gradient estimation and search algorithms, as well as retrospective optimization. Because both our motivating problems and computational testing focuses on supply chain inventory position, we also briefly explore the supply chain inventory positioning literature.

2.1. Stochastic gradient estimation

In a typical constrained stochastic optimization problem, the objective is either directly or indirectly a function of both random variables and decision variables. We seek to optimize the decision variables, but must account for random problem parameters. Clearly, given a set of decision variables, we can obtain an estimate of the objective function value by sampling one or more instances of the random variables.

A traditional approach for solving this type of problem involves estimating the gradient of the stochastic objective function, and searching in improving directions until the objective function value improves by less than a predetermined threshold. A simple way to estimate the gradient involves computing the differences in the objective in each individual search direction. This approach is known