Constant bounded approximation algorithms for stochastic inventory control

Andrew Benton

December 18, 2017

1   Introduction

The typical approach for stochastic inventory control models are based on Markov Decision Processes, which are solved through dynamic programming. Although this model is flexible, it can often result in uncomputable solutions. Due to this, there exists a large literature discussing approximation algorithms which are able to compute policies with reasonable efficiency. Unfortunately, understanding the performance of these heuristics can be difficult.

Levi et al. [2007] present the ideas of *Marginal Cost Accounting* and *Cost Balancing*. Marginal Cost Accounting gives us a method to efficiently compute the full horizon costs incurred by decisions, and thereby avoiding the dynamic programming solution. Cost Balancing allows for an efficient method of solving the resulting equations, and provides bounds on worst case performance. Together, these ideas result in efficient approximation algorithms with bounded worst case performance.

1.1   Marginal Cost Accounting

A cost accounting scheme assigns costs to particular decisions. A marginal cost accounting scheme is one which after a particular decision is made, that cost can be considered to be unalterable by later decisions. Like a myopic policy, marginal cost accounting schemes give us the ability to select optimal decisions in a particular period without any knowledge of later decisions. Unlike a myopic policy, the costs of the decision in future periods are still considered in the current decision.

As an example, we consider holding costs. Holding costs have the property that once an order is placed, the holding costs for the particular items purchased will be incurred until they are consumed by demand. Let h_j be the per unit holding cost in period j, $D_{[t,j]}$ be the cumulative demand from period t to j, and X_t be the inventory in period t. The holding costs incurred by an order of size Q_t in period t are:

$$\sum_{j=t}^{T} h_j (Q_t - (D_{[t,j]} - X_t)^+)^+$$

Note that this equation does not account for any future decisions, allowing the holding costs to be computed for any demand outcome. This allows the expected holding costs to be computed, provided a forecasting distribution over future demands.

1.2   Cost Balancing

Cost balancing is the component of the algorithm which provides us with bounded cost guarantees. Levi et al. [2007] provide a detailed proof which the later papers are built upon - only a summary will be offered here. They let $Z_t = \mathbb{E}[H_t^B \mid F_t] = \mathbb{E}[\Pi_t^B \mid F_t]$. They then establish the following equivalency with the total cost of the balancing policy:

$$\mathbb{E}[\mathcal{C}(B)] = 2 \sum_t \mathbb{E}[Z_t]$$

They then partition the horizon into two sets, $\mathcal{J}_H$ and $\mathcal{J}_\Pi$, where inventory held is above and below optimal respectively. Using this partition, they establish that:

$$H^{opt} + \Pi^{opt} \geq \sum_{t \in \mathcal{J}_H} H_t^B + \sum_{t \in \mathcal{J}_\Pi} \Pi_t^B$$

With these relationships established, it can be shown that $\mathbb{E}[\mathcal{C}(opt)] \geq \sum_t \mathbb{E}[Z_t]$ and therefore:

$$\mathbb{E}[\mathcal{C}(B)] \leq 2\mathbb{E}[\mathcal{C}(opt)]$$

Note that this provides no guarantees about average performance, just worst case performance. Hurley et al. [2007] show that average performance is typically much better, and also provide a set of heuristics for upper and lower bounds for ordering quantity which improve average performance while not violating the constant worst case guarantee.

2   Constant Bounded Approximation Algorithms

Levi et al. [2007] present algorithms for the stochastic inventory control problem and the lot sizing problem. Levi et al. [2008] present algorithms for the capacitated stochastic inventory control problem. Levi et al. [2016] present algorithms for the multi-echelon inventory control problem. For all of these, they present an algorithm as well as worst case performance guarantees.

2.1   Stochastic Inventory Problem

This problem has per unit holding costs h_s and per unit lost sales costs p_s. Given the demand sequence $\{z_t\}_t$, a starting inventory X_t, the period s holding costs $H_s^B(q_s)$ are computed from the current period to the end of the horizon.

$$H_s^B(q_s) = \sum_{j=s}^{T} h_j \big[q_s - \big(\sum_{i=s}^{j} z_i - X_s\big)^+\big]^+$$

The penalty costs $\Pi_s^B(q_s)$ are simplified by the observation that if too few items are bought during the current period, more can be purchased next period. Due to this, penalty costs only need to be computed for a single period.

$$\Pi_s^B(q_s) = p_s[z_s - (X_s^B + q_s)]^+$$

Using these equations, the Balancing Algorithm seeks the order size q_s where these two costs are equal. That is:

$$l_s^B(q_s) = \pi_s^B(q_s)$$

where $l_s^B(q_s) = \mathbb{E}[H_s^B(q_s)\,|f_s]$ and $\pi_s^B(q_s) = \mathbb{E}[\Pi_s^B(q_s)\,|f_s]$.

Levi et al. [2007] show that when this q_s is applied, the incurred costs are guaranteed to be twice of the optimal costs. Furthermore, we are provided with two other policies which are known to bound the optimal policy. The first is the Myopic policy, which minimizes only the next period. This is known to overestimate the optimal order quantity. The second is the Minimizing policy, which minimizes the sum of the marginal holding and penalty cost. Levi et al. [2007] show that this overestimates the optimal order quantity. Hurley et al. [2007] show that combining these bounds with the balancing policy, as well as other bounds, provides us with lower cost policy while keeping the worst case guarantees.

2.2   Lot Sizing Problem

This problem has per unit holding costs h_s, per unit lost sales costs p_s, and a per order fixed cost K. The approach taken by Levi et al. (2007) seeks to separately balance the holding costs and ordering costs as well as the penalty costs and ordering costs. Here, the Balancing Algorithm provides two parameters: the inventory level at which to order, and the level to order up to (analagous to (s, S) in the exact formulation of this problem). An order is made if the backordering costs exceed K:

$$r_s^B = \min_q \{q \ : \ \pi_s^B(q_s) \leq K\}$$

The order size is then:

$$q_s^B = \max_{q_s} \{q \ : \ l_s^B(q_s) \leq K\}$$

Levi et al. [2007] show that when this policy is applied, the incurred costs are guaranteed to be three times of the optimal costs. Unfortunately, these bounds assume that the current period's demand is known deterministically. While this is true in some cases, it does limit the applicability.

2.3   Capacitated Stochastic Inventory Problem

This problem has per unit holding costs h_s, per unit lost sales costs p_s, and a maximum order size u_s. This problem is effectively a generalization of the uncapacitated stochastic inventory

problem, and the balancing algorithm treats it as such. The holding costs are computed the same, but the penalties must now be computed over the whole horizon:

$$\tilde{\Pi}(q_s) = \sum_{j=s}^{T} p_j \left[\left(\sum_{i=s}^{j} z_i - (q_s + X_s + \sum_{i=s}^{j} u_i) \right)^+ - \left(\sum_{i=s}^{j} z_i - (q_s + X_s + \sum_{i=s}^{j} u_i) \right)^+ \right]$$

The balancing algorithm then seeks the order size q_s such that:

$$l_s^B(q_s) = \tilde{\pi}(q_s)$$

where $l_s^B(q_s) = \mathbb{E}[H_s^B(q_s) \,|\, f_s]$ and $\tilde{\pi}_s^B(q_s) = \mathbb{E}[\tilde{\Pi}_s^B(q_s) \,|\, f_s]$. Levi et al. [2008] show that when this q_s is applied, the incurred costs are bounded by two times the optimal costs.

2.4   Multiechelon Stochastic Inventory Problem

The final set of models are described by Levi et al. [2016]. The problem requires the presentation of much notation, so full details will be left to the original paper. This paper shows that solving the multiechelon problems with n stages requires solving n separate balancing policies. These balancing policies are similar to the ones seen above, but with extra holding costs accounted for. For the n stage problem, it is shown that the incurred costs are bounded by twice the optimal costs.

3   Methods for Computing Balancing Policies

Levi et al. [2007], Levi et al. [2008], and Levi et al. [2016] discuss the logic and structure of these algorithms, but offer little advice on how to compute them. Yu [2010] give some detail, but restrict themselves to poisson demand and rely on an graphical method to initialize the algorithm. Hurley et al. [2007] rely on simulation. The following section contains a detailed description of exactly and efficiently computing the balancing policies for independent or multivariate normal demand distributions.

3.1   Analysis of Cost Functions

Marginal Holding Cost $l_s^B(q_s)$

Computing $l_s^B(q_s)$ in the general case requires the computation of a $T-s$ dimensional integral. Traditional cubature algorithms have exponential complexity in dimensions, making such an integral unreasonable to compute. However, some conditions allow it to be reduced to $T-s$ integrals of fewer dimensions. Letting Φ be the (generally multivariate) distribution

of forecast f_s:

$$l_s^B(q_s) = \mathbb{E}[H_s^B(q_s) \mid f_s]$$

$$= \mathbb{E}\left\{\sum_{j=s}^{T} h_j\left[q_s - \left(\sum_{i=s}^{j} D_i - X_s\right)^+\right]^+ \mid f_s\right\}$$

$$= \int_{z_s=0}^{\infty}\int_{z_{(s+1)}=0}^{\infty}\cdots\int_{z_T=0}^{\infty}\sum_{j=s}^{T} h_j\left[q_s - \left(\sum_{i=s}^{j} z_i - X_s\right)^+\right]^+ d\Phi(z_s, z_{s+1}, \ldots z_T)$$

Assuming that the demand of any period would depend only on past demand (rather than future demand), this can be rewritten as a series of $1, 2, \ldots, T-s$ dimensional integrals:

$$l_s^B(q_s) = \sum_{j=s}^{T}\int_{z_s=0}^{\infty}\int_{z_{(s+1)}=0}^{\infty}\cdots\int_{z_j=0}^{\infty} h_j\left[q_s - \left(\sum_{i=s}^{j} z_i - X_s\right)^+\right]^+ d\Phi(z_s, z_{s+1}, \ldots z_j)$$

Letting $\psi_{[s,j]}$ be the (possibly uncomputable) distribution function of cumulative demand, $l_s^B(q_s)$ and $\frac{d}{dq_s}l_s^B(q_s)$ can be expressed as $T-s$ integrals in one dimension:

$$l_s^B(q_s) = \sum_{j=s}^{T} h_j \int_{z_j=X_s}^{X_s+q_s}\left(q_s + X_s - z_j\right)\psi_{[s,j]}(z_j)dz_j \tag{1}$$

$$\frac{d}{dq_s}l_s^B(q_s) = \sum_{j=s}^{T} h_j \int_{z_j=X_s}^{X_s+q_s}\psi_{[s,j]}(z_j)dz_j \tag{2}$$

$l_s^B(q_s)$ is convex increasing with q_s. For nonincreasing h_j, the series in $l_s^B(q_s)$ is nonincreasing with j. This allows the series to be truncated with a bounded error. This is particularly useful for dependent demand distributions, as it allows us to compute the expectation over fewer random variables (often only 1 or 2 if q_s is below or near its optimal value). So long as the bounded error is less than the tolerance of the quadrature algorithm, this approximation will result in trivial error, while allowing faster function evaluations.

For independent demand, $\psi_{[s,j]} = \phi_s \star \phi_{s+1} \star \cdots \star \phi_j$. For distributions with closed-form convolutions (Normal, Exponential, etc), this can be evaluated efficiently. For dependent demand, specialized methods are necessary. If the demand distribution follows a multivariate normal (or can be approximated by one), $\psi_{[s,j]} = N(\sum_{i=s}^{j}\mu_i, \sqrt{\sum_{i=s}^{j}\sum_{k=s}^{j}\sigma_{i,k}^2})$[1]. For other distributions, Arbenz et al. [2011] provide an algorithm which has superior convergence to Monte Carlo methods for the relevant number of dimensions.

Marginal/Forced Backordering Cost $\tilde{\pi}_s^B(q_s)$

The derivations and discussion seen above apply here as well. The equivalent results follow:

$$\tilde{\pi}_s^B(q_s) = \sum_{j=s}^{T}\int_{z_j=X_s+q_s+\sum_{i=s}^{j}u_i}^{\infty} p_j\left(z_j - X_s - q_s - \sum_{i=s}^{j}u_i\right)\psi_{[s,j]}(z_j)dz_j \tag{3}$$

---

[1]This is not true in general for dependent normal random variables. However, it does hold for multivariate normal random variables.

$$\frac{d}{dq_s}\tilde{\pi}_s^B(q_s) = -\sum_{j=s}^{T} p_j \int_{z_j=X_s+q_s+\sum_{i=s}^{j}u_i}^{\infty} \psi_{[s,j]}(z_j)dz_j \tag{4}$$

$\tilde{\pi}$ is convex decreasing with q_s. For nonincreasing p_j, the series in $\tilde{\pi}_s^B(q_s)$ is nonincreasing.

3.2   Computing Balancing Quantities

The balancing policies seek the value q_s^B at which some set of costs are equal to another. For instance, the dual balancing policies seek q_s^B such that:

$$l_s^B(q_s^B) = \pi_s^B(q_s^B) \tag{5}$$

The following will discuss solving equation (5). Adaptations to solve other policies are simple to apply.

If the conditions to compute $\psi_{[s,j]}$ are met, we can exactly compute the cost functions. If $l_s^B(q_s)$, $\frac{d}{dq_s}l_s^B(q_s)$, $\tilde{\pi}_s^B(q_s)$, and $\frac{d}{dq_s}\tilde{\pi}_s^B(q_s)$ are decreasing with j, they can be efficiently computed, with sublinear time complexity in the horizon length.[2]

Levi et al. [2007] suggests using bisection methods to solve this. However bisection methods have linear convergence, which is quite slow for this problem. Fortunately, under our current assumptions, superior methods can be used. Because $l_s^B(q_s)$ is convex and increasing and $\tilde{\pi}_s^B(q_s)$ is convex and decreasing, (5) has one root, one basin of attraction, no stationary points, and is continuously differentiable. Given these conditions, Newton-Raphson is a better choice.

Newton-Raphson begins with an initial estimate x_0, and then iterates the following equation:

$$x_{n+1} = x_n + \frac{f(x_n)}{f'(x_n)}$$

This is a very standard method - implementations and various modifications are widely available. For the dual balancing policy, f and f' would be defined as follows:

$$f(x) = l_s^B(x) - \pi_s^B(x)$$

$$f_s'(x) = \frac{d}{dx}l_s^B(x) - \frac{d}{dx}\pi_s^B(x)$$

Newton-Raphson has quadratic convergence, and as result often requires an order of magnitude fewer function evaluations than the bisection. Furthermore, all but the first Newton-Raphson iteration are "very close" to the optimal value, which, as described in Section 3.1, results in faster function evaluation.

---

[2]No formal proof can be provided, but logically: Assuming the integration is constant per period, computing the entire horizon will be linear with horizon length. If truncated, then there will be some horizon length which, if increased, no further integrations will be required. The only difficulty here is whether each integration is constant. Under the conditions given, this tends to hold.
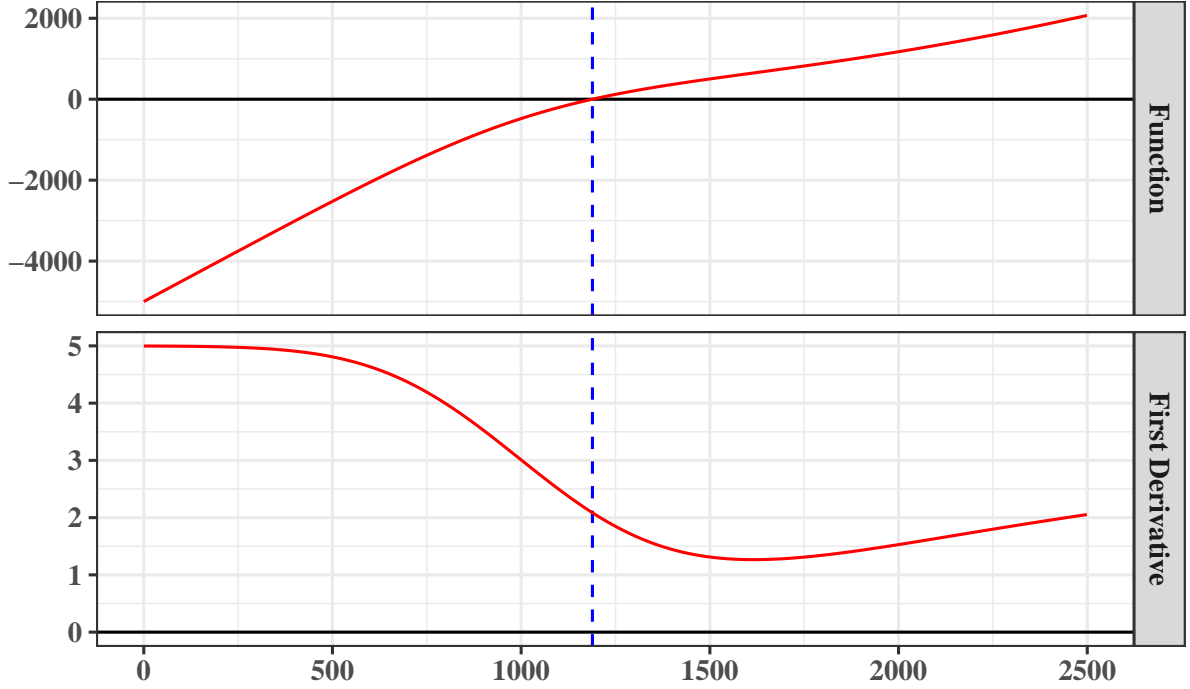
6

Figure 1: Root function for $NID(1000, 300)$ demand forecasts. Optimal q in blue, μ in green. The function is generally smooth.

The only remaining concern when applying Newton-Raphson is the initial estimate x_0. Generally, poor initial estimates can cause subquadratic convergence. There is no general answer to this problem, but numerical experience has indicated that these concerns can be ignored with the initial estimate of $x_0 = \mu_s$.

4 Algorithm Implementation

The scenario studied here is the simplest case, the unconstrainted stochastic inventory problem. The demand forecast is $NID(1000, 300)$ with a penalty cost of $p_j = 5$ for all j and a holding cost of $h_j = 1$ for all j. There are 100 periods in the horizon. These particular demand numbers were selected because the high standard deviation causes longer holding cost evaluations. The truncation error, as well as the tolerance for the quadrature and root-finding algorithms was set to $1 * 10^{-9}$. Such an error should be trivial in the context of inventory control. The initial guess was equal to the one period demand mean, $x_0 = 1000$. No bounding heuristics will be used here - only the balancing policy is applied.

The root function which Newton-Raphson is applied to, as well as the root found by Newton-Raphson, is plotted in Figure 1. Computing this root required 110 nanoseconds. As
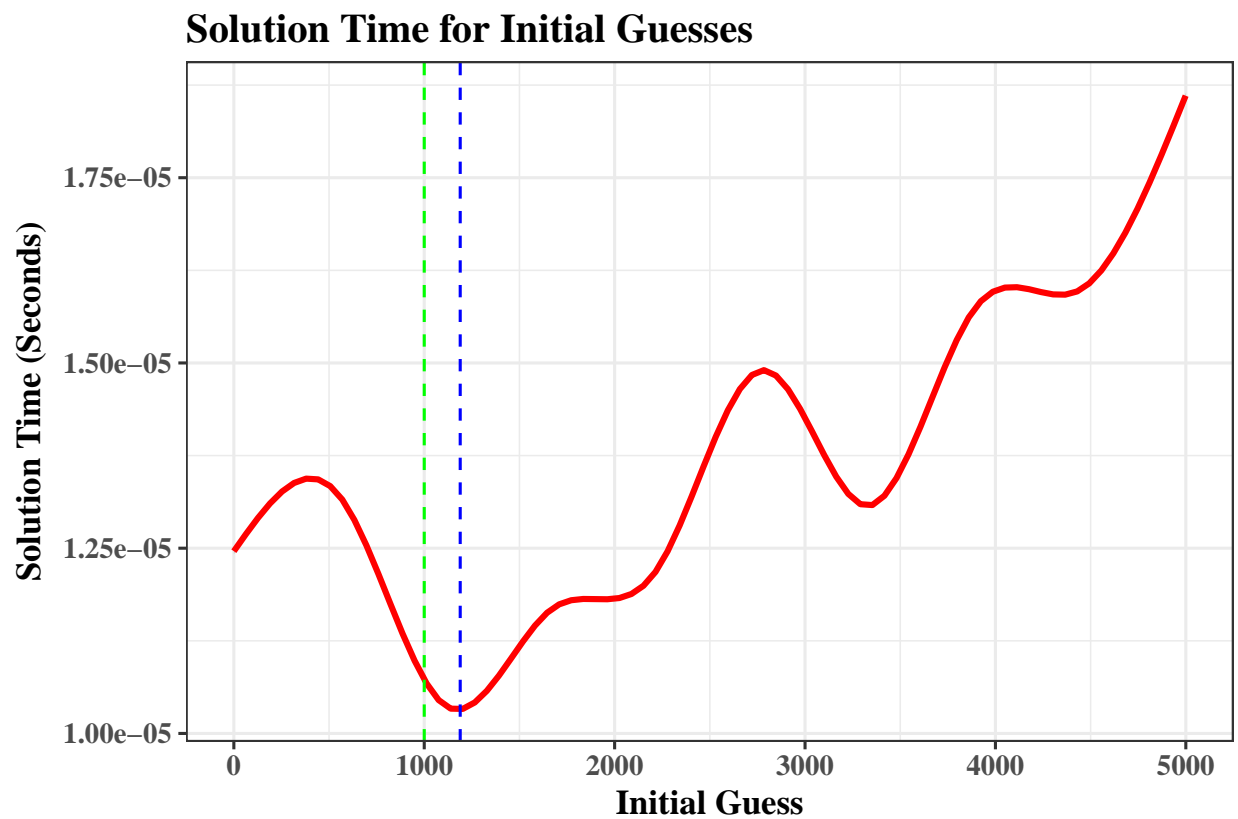
Figure 2: Solution times plotted for various initial guesses. Optimal q in blue, μ in green. Although $x_0 = \mu$ is an effective starting point, the running time for any starting point is small.

discussed in section 3.2.1, the solution time depends upon the quality of the initial guess. However, this relationship is plotted in Figure 2, and shows that differences in solution time are trivial in practice. This plot also confirms that the algorithm converges for a wide range of initial guesses ($x_0 \in \{0, 5000\}$ in this case).

References

P. Arbenz, P. Embrechts, and G. Puccetti. The AEP algorithm for the fast computation of the distribution of the sum of dependent random variables. *ArXiv e-prints*, June 2011.

Gavin Hurley, Peter Jackson, Retsef Levi, Robin O Roundy, and David B Shmoys. New policies for stochastic inventory control models–theoretical and computational results. Technical report, 2007.

Retsef Levi, Martin Pl, Robin O. Roundy, and David B. Shmoys. Approximation algorithms for stochastic inventory control models. *Mathematics of Operations Research*, 32(2):284–302, 2007. doi: 10.1287/moor.1060.0205.

Retsef Levi, Robin O. Roundy, David B. Shmoys, and Van Anh Truong. Approximation algorithms for capacitated stochastic inventory control models. *Operations Research*, 56 (5):1184–1199, 2008. doi: 10.1287/opre.1080.0580.

Retsef Levi, Robin Roundy, Van Anh Truong, and Xinshang Wang. Provably near-optimal balancing policies for multi-echelon stochastic inventory control models. *Mathematics of Operations Research*, 42(1):256–276, 2016.

Qian Yu. *Evaluation of cost balancing policies in multi-echelon stochastic inventory control problems*. PhD thesis, Massachusetts Institute of Technology, 2010.