Provably Near-Optimal Approximation Schemes for Implicit Stochastic and for Sample-Based Dynamic Programs

Nir Halman
Hebrew University of Jerusalem
halman@huji.ac.il

March 30, 2017

Abstract
In this paper we address two models of non-deterministic discrete-time finite-horizon dynamic programs (DPs): implicit stochastic DPs – the information about the random events is given by value oracles to their CDFs; and sample-based DPs – the information about the random events is deduced by drawing random samples. Such data-driven models frequently appear in practice, where the cumulative distribution functions of the underlying random variables are either unavailable or too complicated to work with. In both models the single period cost functions are accessed via value oracle calls and are assumed to possess either monotone or convex structure. We develop the first near-optimal relative approximation schemes for each of the two models. Applications in stochastic inventory control, i.e., several variants of the so called Newsvendor problem are discussed in detail. Our results are achieved by a combination of Bellman equation calculations, density estimation results, and extensions of the technique of $K$-approximation sets and functions introduced by Halman et al. [Math. Oper. Res., 34, (2009), pp. 674–685].

Keywords: Approximation algorithms, inventory control, $K$-approximation sets and functions, sample average approximation.

1 Introduction

Newsvendor problem (NV). In this paper we consider a fundamental single-period stochastic inventory control problem called the newsvendor problem. A vendor needs to decide how many units $x$ of an item with short life cycle (such as newspapers, fashion items and electrical circuits) to order based on the known demand distribution, the ordering cost, the holding cost, and the lost sales cost. Let the cost of ordering $x$ units of the item in the beginning of the period be $c(x)$, and the cost of having $x$ unsold units at the end of the selling period be $h(x)$. Let $b(x)$ be the lost sales cost, which is incurred when there is unfulfilled demand of $x$ units. All these functions are nonnegative and nondecreasing. The stochastic demand $D$ for the item is a discrete random variable with finite support $[A, \ldots, B]$, and is described by its cumulative distribution function (CDF) $F(y) = F_D(y) = \text{Prob}(D \leq y)$. For the sake of simplicity, we assume from now on that $h(x) = b(x) = 0$ for all $x \leq 0$. Researchers have followed two approaches to solving NV. In the first approach, the expected cost of overestimating and underestimating demand is minimized. In the second approach the expected profit is maximized. Both approaches yield the same optimal solution. In this paper we follow the first approach: at the beginning of the period, the vendor decides on the number $x$ of items to order in order to minimize her costs, i.e.,
\[ \min_{x \geq 0} C(x) \]  

where \( C(x) := c(x) + E_D[h(x - D) + b(D - x)]. \)  

[AHM51] showed that if all costs are linear, i.e., \( c(x) = cx,\) \( h(x) = hx,\) \( b(x) = bx,\) then this problem permits a simple solution policy: Determine a base stock \( S := \arg \max \text{Prob}(D \leq S) \leq \frac{b}{b+h} \) and order enough to bring the stock level to \( S, \) the so-called base stock policy. \( \frac{b}{b+h} \) is called the critical ratio. See also [Kho99, QWV+11], [SCB14, p. 120] or any standard reference on inventory management. Interest in the NV has greatly increased over the past 60 years. This interest stems from the fact that the problem serves as a building block in many inventory and supply chain models as well as its relevance to practice, e.g., in service industries such as air transportation and hospitality. Moreover, the increase in product variety along with shrinkage of product lifecycles and relative steadiness of production and transportation leadtimes make single-period inventory models more relevant than ever [Kho99].

**Nonlinear newsvendor problem (NNV).** In the nonlinear newsvendor problem not all cost functions are necessarily linear, e.g., in the more realistic settings when the ordering cost includes a setup cost and quantity discounts such as truckload discounts and all-units discounts, or increasing unit replenishment costs, see Figure 1. Unlike its linear counterpart, in NNV function \( C(\cdot) \) is not necessarily convex. During the last half century a large body of work has been focused on the structure of the optimal policy for NNV as a function of the initial inventory level, under various assumptions on the cost functions (refer to the excellent extensive survey of [Por90] and the references therein for more detail).

![Figure 1: Examples for nonlinear ordering costs](image-url)

**Sample-based NV (SNV, SNNV).** Assuming the demand distribution is given explicitly, or even as an oracle to its CDF, is not always realistic. Indeed, firms typically maintain a database that includes historical customer demand information and update it with daily or weekly point of sale (POS) data. In such an environment, there may not be a function representing the demand distribution. Rather, the database provides the empirical probability that demand (or more precisely, sales) is smaller than a certain value, for any value inspected by the user. In the sample-based NV the goal remains (1) but the optimization is based only on observed independent samples (also called observations) of the demand, without any access to the true demand distribution.

**Non-deterministic dynamic programs.** Although the relevance of NV has increased in the last decades, there are still many problems that can only be formulated as multiple periods problems. Examples include the multi-period NV which is discussed in the next paragraph (see also [LRS07]), and the discrete-time
Formally, we want to determine: a stochastic dynamic program minimizes the expectation of the total incurred cost. This problem is called a multi-period NV problem. In our context, the system dynamics of the random variables $D_t$ are given in [Ber05] and Appendix B of this paper. In the DP that is defined in [Ber05], the system dynamics are of the form: $I_{t+1} = f(I_t, x_t, D_t)$ for $t = 1, \ldots, T$, where $t$ is the discrete time index, $I_t \in S_t$ is the state of the system at time $t$ ($S_t$ is the state space at stage $t$), $x_t \in A_t(I_t)$ is the action or decision to be selected at time $t$ after observing state $I_t$ ($A_t(I_t)$ is the action space at stage $t$ and state $I_t$), $D_t$ is a discrete random variable over the sample space $D_t$, and $T$ is the number of time periods. In the context of this paper, $D_t$ represents an exogenous information flow. The cost function $g_t(I_t, x_t, D_t)$ gives the cost of performing action $x_t$ from state $I_t$ at time $t$ for each possible realization of the random variable $D_t$. In this expression, $g_{T+1}(I_{T+1})$ is the cost paid if the system ends in state $I_{T+1}$, and the sequence of states is defined by the system dynamics. The problem is that of choosing a sequence of actions $x_1, \ldots, x_T$ that minimizes the expectation of the total incurred cost. This problem is called a stochastic dynamic program. Formally, we want to determine:

$$z^*(I_1) = \min_{x_1 \in A(I_1), \ldots, x_T \in A(I_T)} E \left[ g_1(I_1, x_1, D_1) + \sum_{t=2}^T g_t(f(I_{t-1}, x_{t-1}, D_{t-1}), x_t, D_t) + g_{T+1}(I_{T+1}) \right],$$

where $I_1$ is the initial state of the system and the expectation is taken with respect to the joint probability distribution of the random variables $D_t$.

**Multi-period NV problems.** All NV problems mentioned above can be extended to $T$ time periods in the following way [LRS07]. For each time period $t = 1, \ldots, T$ let the demand be denoted by the random variable $D_t$ and the ordering cost by $c_t(\cdot)$. Excessive inventory at the end of time period $t$ is carried over to the next time period at a cost of $h_t(\cdot)$. Unmet demand is backlogged to the next time period at a cost of $b_t(\cdot)$. The inventory level at the end of period $T$ incurs a penalty cost of $g_{T+1}(\cdot)$. For the sake of simplicity we define a holding/backlogging cost function $h_t(\cdot)$ as follows: $h_t(x) := h^t(x)$ for $x \geq 0$ and $h_t(x) = b^t(-x)$ for $x < 0$. We note that $h_t$ is a unimodal function which achieves its minimum at 0. Moreover, if both $h^t(x), b^t(x)$ are linear functions for nonnegative $x$’s, then $h_t$ is also convex and “V”-shaped. This problem can be formulated as a DP with $g_t(I_t, x_t, D_t) = c_t(x_t) + h_t(I_t + x_t - D_t)$ and $f(I_t, x_t, D_t) = I_t + x_t - D_t$.

**On the representation of the random events.** We distinguish between three variants in the representation of uncertainties: (i) explicit stochastic DPs, where the distribution of each random variable $D_t$ is given explicitly as a list of scenarios $(d^i_t, \text{Prob}(D_t = d^i_t))$, $i = 1, \ldots, n_t$, (ii) implicit stochastic DPs, where it is given as a value oracle to its CDF, and (iii) sample-based DPs where the distribution of $D_t$ is not given, and the only information available is a set of independent random draws from $D_t$. It is well known that problem (2) can be solved through a recursion.

**Theorem 1.1 (Bellman optimality equation [BD62]).** For every initial state $I_1$, the optimal value $z^*(I_1)$ of the DP is given by $z_1(I_1)$, where $z_1$ is the function defined by $z_{T+1}(I_{T+1}) = g_{T+1}(I_{T+1})$ together with the recursion

$$z_t(I_t) = \min_{x_t \in A_t(I_t)} E_{D_t} \left[ g_t(I_t, x_t, D_t) + z_{t+1}(f(I_t, x_t, D_t)) \right], \quad t = 1, \ldots, T,$$

Assuming that $|A_t(I_t)| = |A|$, $|S_t| = |S|$ and $|D_t| = |D|$ for every $t$ and $I_t \in S_t$, gives a pseudopolynomial algorithm that runs in time $O(T|A||S||D|)$. A polynomial algorithm should run in time polynomial in $T, \log |A|, \log |S|$, and $|D|$. If the representation of the random events is implicit – the running time should be only polylogarithmic in $|D|$.
**Approximation schemes.** Given a minimization problem and a relative error $\epsilon > 0$, a $(1+\epsilon)$-approximation algorithm finds a solution, where the value of the objective function is at most $(1 + \epsilon)$-times the optimal value. If the running time of the algorithm is polynomial in the input size and $1/\epsilon$, $\forall \epsilon > 0$, this algorithm is called a fully polynomial time approximation scheme (FPTAS). FPTASs are considered as the “holy grail” among approximation algorithms, because we have the ideal situation where we can balance between solution quality (an upper bound on how far we are from the optimum solution) and solution speed (an upper bound on the running time of the algorithm).

1.1 Our results

The only FPTAS framework in the literature about non-deterministic DPs deals with explicit stochastic DPs [HKL+14]. Although the assumption that the random variables are given explicitly is frequently used in the literature, it is limiting once one wants to address more realistic settings where the uncertainty model is very rich, e.g., problems for which either knowing the probability distribution or even computing expectations is impractical. This paper addresses these two settings and generalizes the framework of [HKL+14] to the case where the random event in each period is only implicitly specified, or only available via independent samples. For implicit stochastic DPs we achieve an FPTAS.

**Theorem 1.2.** Every implicit stochastic minimization DP satisfying some technical conditions (Conditions 5, 6 and 7 in this paper) admits an FPTAS.

The set of technical conditions required in the above theorem is similar to the one required in the framework of [HKL+14]. These conditions are essentially about the structure of the cost functions, being either monotone or convex; the transition function; the state and action spaces; and how the input is given. (We give detail on the main differences between these two sets of conditions in Section 6.)

For sample-based DPs we achieve a first approximation scheme framework under conditions similar to the ones in Theorem 1.2. Before formally stating this result we give a definition. For any $\gamma > 0$ we say that $\gamma$ is a bound on the CDF of the demand $D$ if $\Pr(D \leq x) > 0$ implies $\Pr(D \leq x) \geq \gamma$ and $\Pr(D \leq y) < 1$ implies $\Pr(D \leq y) \leq 1 - \gamma$, $\forall x, y$. (Note that because $\gamma$ is (merely) a bound, it does not limit the distribution of the demand to be symmetric.) The base two logarithm of $z$ is denoted by $\log z$. For ease of notation, we assume throughout that $\log_b z = \max\{\log_b z, 1\}$ for any basis $b$ and positive value of $z$.

**Theorem 1.3.** Given an instance of a sample-based DP satisfying some technical conditions (Conditions 5, 7 and 9 in this paper), a relative error $\epsilon > 0$, and an uncertainty parameter $\delta > 0$, there exists an algorithm which computes a solution that with probability of at least $1 - \delta$ has value within ratio of $1 + \epsilon$ of an optimal solution. The algorithm performs $N = O(T^2 \log(T/\delta))$ independent samples of each $D_t$ and runs in time $O(\text{poly}(\text{inp}) + TN \log N)$, where $\text{inp}$ is the binary input size.

To show that the polynomial dependence on $1/\gamma$ in Theorem 1.3 is in general unavoidable we note that (linear) SNV can be cast as a single-period DP satisfying the conditions of Theorem 1.3 and give Theorem 1.4 below. We need some definitions first. For every pair $A, B$ of integers, $A < B$ let $[A, \ldots, B] = \{A, A + 1, \ldots, B\}$ denote the set of integers between $A$ and $B$. Let $V_{\text{max}}$ be the maximal value of a cost function and $V_{\text{min}}$ be the minimal strictly-positive value of such function. We call any constant $M$ that satisfies $M \geq V_{\text{max}} / V_{\text{min}}$ a bound on the cost function.

**Theorem 1.4.** For every given ratio $K > 1$ and an uncertainty parameter $\delta > 0$, calculating with confidence $1 - \delta$ a base stock $S$ that approximates the value of the sample-based linear newsvendor problem within ratio $K$ requires $\Omega(\min\{M, 1/\gamma\} \log 1/\delta)$ independent observations of the demand distribution, where $M$ and $\gamma > 0$ are bounds on the cost functions and the CDF of the demand, respectively. The same number of observations is also needed for approximating the value of the problem within ratio $K$. 

Our techniques and tools also extend beyond DPs. To demonstrate this we study various variants of NNV. Recall that in this case the function $C(\cdot)$ in (1) is not necessarily convex. Clearly, it is neither monotone. Therefore, NNV cannot be cast as a DP satisfying Condition 7 and Theorems 1.2-1.3 do not apply for it. Considering implicit stochastic NNV we can still provide an FPTAS.

**Theorem 1.5.** The nonlinear newsvendor problem in which the demand distribution is given as a value oracle to the CDF, and the cost functions are given as value oracles admits an FPTAS.

We also provide a provably near-optimal solution for SNNV as follows. (For every function $\varphi(\cdot)$ we denote by $t_{\varphi}$ the time needed to evaluate $\varphi(\cdot)$ on a any single point in its domain.)

**Theorem 1.6.** Consider an instance of sample-based nonlinear newsvendor problem in where the support of the demand $D$ is contained in $[A, \ldots, B]$ and the cost functions are given as value oracles. For every given ratio $K = 1 + \epsilon > 1$, uncertainty parameter $\delta > 0$, bound $\gamma > 0$ on the CDF of the demand, bound $M$ on the cost functions, and $N$ independent samples of $D$ with $N \geq \frac{9 \log(4/\delta)}{\gamma^2 \epsilon^2}$, one can compute in

$$O\left( (1 + t_b + t_c + t_h) \frac{\log^2 M + \log(1/\gamma)}{\epsilon^2} \log(B - A) \log\left( \frac{\log(1/\gamma)}{\epsilon} \right) + N \log N + \frac{\log^2(B - A) \log(1/\gamma)}{\epsilon} \right)$$

time an order quantity $q'$ such that with probability at least $1 - \delta$, the expected cost incurred when ordering $q'$ units is at most $K$ times the expected cost incurred when ordering an optimal quantity $q^*$, i.e., $C(q') \leq K c(q^*)$.

Considering the bound in Theorem 1.6, we note that Theorem 1.4 tells us that one cannot hope for running time which is polynomial in the input size, i.e, that also depends polylogarithmically on both $M$ and $1/\gamma$.

**Technique used.** We take one of the most common approaches to learning and stochastic optimization, and solve the sample average approximation (SAA) counterpart [Sha03] (aka - empirical risk minimization), where instead of performing the expectation in (1) with respect to the true underlying probability distribution, we perform it with respect to the empirical probability distribution. We perform a careful analysis of the errors that originate from the empirical CDF and propagate to the objective function value, in order to establish an upper bound on the number of independent samples of $D$ as a function of $\epsilon, \delta$ and $\gamma$. To perform this analysis and develop the approximation schemes, we combine Bellman equation calculations, which are standard tools in dynamic programming, and combinatorial methods in density estimation, which are common tools in supervised learning, with the recent technique of $K$-approximation sets and functions developed in [HKL+14].

### 1.2 Relevance to existing literature

**Constructive frameworks that give approximation schemes for optimization problems.** We are aware of six such frameworks. The ones in [Woe00, PW07, MS13] deal with deterministic discrete DPs only. The one in [SS06] deals with stochastic linear programs (LPs). The one in [HKL+14] deals with explicit stochastic discrete DPs in which the single period cost functions are assumed to possess either monotone or convex structure. The one in [HNO15] works for a subclass of the latter DP model, in which the single period cost functions are assumed to possess convex structure, and provides a faster FPTAS from both theoretical worst-case upper bounds and practical standpoint. We are not aware of any approximation scheme framework for implicit stochastic DPs in general, although [HOS12] provides problem-specific FPTASs for the profit-maximization NNV problem where the distribution of the demand is given implicitly. In this sense we are the first to give approximation scheme frameworks for either implicit stochastic DPs or sample-based DPs.
Markov Decision Processes (MDPs). An $S$-state MDP with $A$ actions has probability $P(i, a, j)$ of going to state $j$, given that we are in state $i$ and execute action $a$. The (possibly random) reward received for executing $a$ from $i$ is $R(i, a)$. When considering MDPs with very large state and action spaces, and especially when the probability distributions $P(i, a, j)$ are described explicitly, the classical planning assumption that the MDP is given explicitly by tables of rewards and transition probabilities becomes infeasible, as it takes $\Omega(S^2 A D)$ space, where $D$ is an average cardinality of the support of a distribution. A number of interesting algorithmic suggestions have been made for coping with such large MDPs while retaining a worst-case additive error guarantee. For example, the phased Q-learning algorithm by [KS99], and the sparse sampling algorithm of [KMN02] (see also the excellent survey of [Kak03]), both run in time sublinear in $S^2 A$. Therefore these algorithms can only produce a very coarse approximation of the MDP itself. However, they give a provably near-optimal additive approximation to the planning problem. Another approach to tackle this representational difficulty is to assume that the MDP has some special structure that permits compact representation, such as MDPs in which the state transition probabilities factor into a small number of components, see [BDG95, KP99, MHK+98]. In this work we follow this second approach. For every time period $t$ the MDP is succinctly induced by a single random event $D_t$: I.e., the probability $P_t(i, a, j)$ is derived from the distribution of $D_t$ and the (a priori given) system dynamics $f(i, a, D_t)$. We also assume some (either monotone or convex) structure on this system dynamics and the single-period cost functions. Note that while in the works cited above the MDPs are stationary and the approximation guarantees are additive, in our DPs the MDPs may be time dependent and our schemes provide relative error guarantees.

Inventory control models. There is ample study on NV with partial information on the underlying demand distribution. The most relevant works are [LRS07, CSL15, LPU15], which all study sample-based linear NV via the SAA approach. In (linear) NV, function $C(\cdot)$ in (1) is convex and hence, optimal order quantities can be characterized in a compact way through first-order information. All three works give approximation schemes that run in time polynomial in $\frac{1}{\epsilon}, \frac{b+h}{\min(b,h)}$ and logarithmic in $\frac{1}{\delta}$. The running time is not polynomial in the input size, though, since $b, h$ are encoded in $\log b, \log h$ bits. [LRS07, CSL15] make no assumptions on the actual demand distribution, while we, as well as part of the work of [LPU15] do make assumptions on the demand distribution. [CSL15] improve upon [LRS07] in the following three senses: they provide hardness results for both the single-period and multi-period NV problems, they deal with the more general problem in where there are capacity limits on the number of items that can be ordered, and they provide smaller upper bounds on the number of independent samples (as a function of $T$) for the multi-period problem. [LPU15] deal only with the single-period NV. In the case of general demand distributions they provide a smaller upper bound than the one of [LRS07] on the number of independent samples (as a function of $\min\{b, h\}/(b + h)$) sufficient for their approximation scheme to work. They also consider log-concave distributions, in which case they further reduce the number of samples. Our work also uses the SAA approach and we also assume a special family (although different) of distributions, but we relax the linearity assumption on the single-period cost functions to arbitrary monotone or convex functions. (Of course, the multi-period NV still fits in our model.) The dependency on $T$ of the number of independent draw samples sufficient for our approximation scheme to work, is by orders of magnitude lower than the ones reported in [LRS07, CSL15]. (Note - in [LRS07, CSL15] there is no assumption on the demand distribution, but on the other hand they deal with the specific multi-period (linear) NV.) Clearly, there is a tradeoff between the assumptions made on the cost functions and those on the distributions, which influences the bound on the number of samples. Because of this tradeoff our work is best viewed as extending the SAA approach to get bounds on the number of samples for DPs with nonlinear cost functions, rather than a one-to-one improvement upon the results reported in [LRS07, CSL15, LPU15]. We summarize the results discussed above in Table 1.
A limitation of our results is our requirement that the RV has discrete and finite support. This raises the question whether one can do without this requirement, i.e., can one extend our results to continuous RVs with bounded support \([A, B]\), while still handling nonlinear cost functions. A recent paper shows that the answer is in the negative even in the implicit stochastic case, and even when the cost functions are Lipschitz continuous and are either monotone or convex, because in such a case no constant factor approximation is possible \([HN16, \text{Thm. 1.3, Thm. 1.5}]\). This means that in a sense our results are “best possible”.

### 1.3 Our contribution

This paper studies variants of NV problems and DPs according to how uncertainties about the input are represented and provides hardness results and near-optimal approximation schemes by carefully quantifying the propagation of errors throughout the algorithms. More specifically, it makes the following contributions. On the negative side, we show in Theorem 1.4 that sample-based (linear) NV cannot be approximated to within any constant multiplicative factor in polynomial time. This implies that the non-polynomial dependency of the running time of \([LRS07]\) on the input size is unavoidable. Our proof is direct and very simple. We note that this negative result is also implied from \([CSL15, \text{Thm. 3.4}]\), but their proof is considerably longer.

We also provide a few positive results, i.e., approximation schemes as follows. First, in Theorem 1.2 we give the first FPTAS framework for implicit stochastic DPs. Second, in Theorem 1.3 we give the first approximation scheme framework for sample-based DPs. The number of samples required by our algorithms is independent of the recursion level \(t = 1, \ldots, T\), and its dependency on \(T\) is only \(O(T^2 \log T)\). It is interesting to note that the multi-period SNV model studied here falls into this framework. We also note that these two contributions can be viewed as extending the framework of \([HKL+14]\) to the implicit stochastic and the sample-based settings. We note that \([HKL+14]\) apply their framework to seven explicit stochastic problems – stochastic knapsack, single-item inventory control, single-item batch dispatch, single-resource revenue management, growth models, lifetime consumption of risky capital and cash management. As a consequence, all of these explicit stochastic problems also admit an approximation scheme when the distributions of the random events are given either implicitly or via independent samples.

Last, we show that our techniques and tools extend beyond DPs by addressing two variants of NNV which cannot be cast in our frameworks above because its value function is neither convex nor monotone: In Theorem 1.2 we give an FPTAS for implicit stochastic NNV and in Theorem 1.6 we provide an approximation scheme for sample based NNV (SNNV) in the case a bound on the CDF of the demand is given.
### Preliminaries

Let $\mathbb{R}, \mathbb{R}^+, \mathbb{Z}, \mathbb{Z}^+, \mathbb{N}$ denote the set of real numbers, nonnegative real numbers, integers, nonnegative integers and positive integers, respectively. For every pair $a, b$ of reals, $a < b$ let $[a, b] = \{x \mid a \leq x \leq b\}$ denote the set of reals between $a$ and $b$. A function $\varphi : D \to \mathbb{R}$ over a linearly ordered set $D$ is called *increasing* if it is nondecreasing on $D$. Similarly, $\varphi$ is called *decreasing* if it is nonincreasing on $D$.

Given a finite set $D \in \mathbb{R}$, let $D_{\min} = \min D$ and $D_{\max} = \max D$. Given $x \in [D_{\min}, D_{\max}]$, for $x < D_{\max}$ let $\text{next}(x, D) := \min\{y \in D \mid y > x\}$, and for $x > D_{\min}$ let $\text{prev}(x, D) := \max\{y \in D \mid y < x\}$. Let $\varphi_{\max} = \max_{x \in D} \varphi(x)$ and $\varphi_{\min} = \min_{x \in D} \varphi(x)$. For simplicity, we assume throughout that $\varphi_{\max} > \varphi_{\min}$. Let $W \subseteq D$ be an arbitrary subset of $D$. We define the *piecewise linear extension of $\varphi$ induced by $W$* as the function obtained by making $\varphi$ linear between successive values of $W$. Function $\varphi$ over $D$ is called *convex over $W$* if its piecewise linear extension induced by $W$ is convex. We define $W^- = \{\text{prev}(x, D) \mid x \in W \setminus \{D_{\min}\}\}$ and $W^+ = \{\text{next}(x, D) \mid x \in W \setminus \{D_{\max}\}\}$.

### 2 A simple impossibility result

In this section we prove Theorem 1.4. We give more impossibility results about NV in Appendix A.

**Proof. (of Theorem 1.4)** Let $A = 0, B = 1$, i.e., the realized demand is always either 0 or 1. Let $\beta = \text{Prob}(D = 0)$. Let the linear cost functions be $b = 1, c = 0$ and $h = MK$ where $M$ is a given large integer. (The critical ratio is then $\alpha = 1/(1 + MK)$. The optimal ordering quantity is $q^* = 0$ if and only if $\beta \in [\alpha, 1]$.)

Consider the case where $\beta = K\alpha$. Clearly, in this case the optimal solution is to order nothing with expected cost $C(0) = 1 - K\alpha = 1 - K/(1 + MK) = (1 + (M - 1)K)/(1 + MK)$. If we order one unit the cost will rise to $C(1) = K\alpha h = MK^2/(1 + MK)$. The ratio between these values is $C(1)/C(0) = MK^2/(1 + (M - 1)K) > K$. Hence, any $K$-approximation solution must order nothing.

We note that in the case of $\beta = 0$, i.e., the demand is always one unit, $C(1) = 0$ and $C(0) = 1$, so in this case any $K$-approximation solution must order one unit.

If the probability distribution is accessible only via sampling, any $K$-approximation algorithm must distinguish between the above two distributions. If this should be done with probability at least $1 - \delta$, we need to choose a number of samples $N$ such that the probability of not getting any “0” sample in the former distribution is less than $\delta$, i.e., $(1 - K\alpha)^N < \delta$. Therefore we get that

$$N > \frac{\log \delta}{\log(1 - \frac{K}{1 + MK})} = \frac{\log \frac{1}{\delta}}{\log(1 + \frac{K}{1 + (M - 1)K})} \geq \frac{(1 + (M - 1)K) \log \frac{1}{\delta}}{2K} > \frac{M - 1}{2} \frac{1}{\log \frac{1}{\delta}},$$

i.e., any $K$-approximation algorithm that succeeds with probability at least $1 - \delta$ must perform $\Omega(M)$ samples. We conclude the proof by noting that $M \approx 1/\gamma$. \qed
In the proof above we have \( M \approx \frac{\max\{h,b\}}{\min\{h,b\}} \). Therefore, Theorem 1.4 implies that the non-polynomial dependency in the input size in the approximation schemes in \([LRS07]\) is unavoidable.

### 3 On K-approximation sets and functions and an FPTAS for NNV

In this section we review the technique of \( K \)-approximation sets and functions introduced by \([HKM^{+}09]\) (see also \([HKL^{+}14]\)) and prove Theorem 1.5, that is, give an FPTAS for NNV.

**Definition 3.1** \((K\text{-approximation functions, Def. 4.1 in [HKL}\^{+}14])\). Let \( K = 1 + \epsilon \geq 1 \) and \( r, \tilde{r} \geq 0 \) be real numbers. We say that \( \tilde{r} \) is a \( K \)-approximation value of \( r \) (or more briefly, a \( K \)-approximation of \( r \)) if \( r \leq \tilde{r} \leq K r \). We say that \( \epsilon \) is the relative error of \( \tilde{r} \). Let \( \varphi, \tilde{\varphi} : D \rightarrow \mathbb{R}^+ \) be real-valued functions over a finite domain \( D \). We say that function \( \tilde{\varphi} \) is a \( K \)-approximation function of \( \varphi \) (\( K \)-approximation of \( \varphi \), in short) if \( \varphi(x) \leq \tilde{\varphi}(x) \leq K \varphi(x) \) for all \( x \in D \) (i.e., if \( \tilde{\varphi}(x) \) is a \( K \)-approximation value of \( \varphi(x) \) for all \( x \in D \)).

The following proposition, called the calculus of \( K \)-approximation Functions, provides a set of general computational rules of \( K \)-approximation functions.

**Proposition 3.2** \((\text{Calculus of } K\text{-approximation Functions, Prop. 5.1 in [HKL}\^{+}14])\). For \( i = 1, 2 \) let \( K_i \geq 1 \), let \( \varphi_i : D \rightarrow \mathbb{R}^+ \) be an arbitrary function over domain \( D \), and let \( \tilde{\varphi}_i : D \rightarrow \mathbb{R}^+ \) be a \( K_i \)-approximation of \( \varphi_i \). Let \( \psi_1 : D \rightarrow D \), and let \( \alpha, \beta \in \mathbb{R}^+ \). The following properties hold:

1. \( \varphi_1 \) is a 1-approximation of itself.
2. (linearity of approximation) \( \alpha + \beta \tilde{\varphi}_1 \) is a \( K_1 \)-approximation of \( \alpha + \beta \varphi_1 \),
3. (summation of approximation) \( \tilde{\varphi}_1 + \tilde{\varphi}_2 \) is a \( \max\{K_1, K_2\} \)-approximation of \( \varphi_1 + \varphi_2 \),
4. (composition of approximation) \( \tilde{\varphi}_1(\psi_1) \) is a \( K_1 \)-approximation of \( \varphi_1(\psi_1) \),
5. (minimization of approximation) \( \min\{\tilde{\varphi}_1, \tilde{\varphi}_2\} \) is a \( \max\{K_1, K_2\} \)-approximation of \( \min\{\varphi_1, \varphi_2\} \),
6. (maximization of approximation) \( \max\{\tilde{\varphi}_1, \tilde{\varphi}_2\} \) is a \( \max\{K_1, K_2\} \)-approximation of \( \max\{\varphi_1, \varphi_2\} \),
7. (approximation of approximation) If \( \varphi_2 = \tilde{\varphi}_1 \) then \( \tilde{\varphi}_2 \) is a \( K_1K_2 \)-approximation of \( \varphi_1 \).

The basic idea underlying the construction of \( K \)-approximation of functions is to keep only a logarithmic number of points in their domain, in what is called a \( K \)-approximation set. We want to only keep points in \( D \) such that the function value “jumps” by less than a factor \( K \) between adjacent points. For simplicity, in the rest of this paper the definitions and results are presented for domains that are contiguous intervals. They can all be extended to general finite linearly ordered domains.

**Definition 3.3** \((K\text{-approximation sets, Def. 4.2 in [HKL}\^{+}14])\). Let \( K \geq 1 \) and let \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be a unimodal function. We say that \( W \subseteq [A, \ldots, B] \) is a \( K \)-approximation set of \( \varphi \) if the following three properties are satisfied: (i) \( A, B \in W \). (ii) For every \( x \in W \setminus \{B\} \), either \( \text{next}(x, W) = x + 1 \) or \( \max\{\varphi(x), \varphi(\text{next}(x, W))\} \leq K \min\{\varphi(x), \varphi(\text{next}(x, W))\} \). (iii) For every \( x \in [A, \ldots, B] \setminus W \), we have \( \varphi(x) \leq \max\{\varphi(\text{prev}(x, W)), \varphi(\text{next}(x, W))\} \leq K \varphi(x) \).

We then use either a step function (for approximating monotone functions) or piecewise linear functions (for approximating convex functions) to determine the function value at points that have been eliminated from the domain.
**Definition 3.4** (Based on Def. 4.4 in [HKL+14]). Let \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be a real-valued function. Let \( W \subseteq [A, \ldots, B] \) be a set that contains \( A, B \). The approximation of \( \varphi \) induced by \( W \) is \( \hat{\varphi}(x) = \varphi(x) \) if \( x \in W \) and \( \hat{\varphi}(x) = \max\{\varphi(prev(x, W)), \varphi(next(x, W))\} \) otherwise. The convex extension of \( \varphi \) induced by \( W \) is the lower envelope of the convex hull of \( \{(x, \varphi(x)) \mid x \in W\} \).

[HKL+14, Alg. 1] design an algorithm, called APXSET (see Appendix C for the algorithm statement) that for every given parameters \( \varphi, D, x^*, K \), returns a \( K \)-approximation set \( W \) for the unimodal function \( \varphi : D \rightarrow \mathbb{R}^+ \) which is minimized at \( x^* \), such that the following proposition holds.

**Proposition 3.5** (Approximation of a unimodal function with direct access, based on Prop. 4.5-4.6 in [HKL+14]). Let \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be a unimodal function which is minimized at a given point \( x^* = \arg\min\{\varphi(x) \mid x \in [A, \ldots, B]\} \). For every given parameters \( \varphi, [A, \ldots, B], x^* \) and \( K > 1 \) the following holds. (i) APXSET constructs a \( K \)-approximation set \( W \) for \( \varphi \) of cardinality \( O(\log K \frac{\text{max}}{\text{min}}) \) in \( O((1 + t_\varphi) \log K \frac{\text{max}}{\text{min}}) \log(B - A)) \) time. (ii) The approximation of \( \varphi \) induced by \( W \), \( \hat{\varphi} \), is a step \( K \)-approximation function of \( \varphi \). (iii) If \( \varphi \) is increasing (decreasing) then \( W \cup W^+ \) (\( W \cup W^- \)) is a \( 1 \)-approximation set of \( \hat{\varphi} \), respectively (iv) If \( \varphi \) is convex then the convex extension of \( \varphi \) induced by \( W, \hat{\varphi} \), is a piecewise linear convex \( K \)-approximation function of \( \varphi \). (v) The values of \( \hat{\varphi} \) and \( \hat{\varphi} \) at any point in \( [A, \ldots, B] \) can be determined in \( O(\log |W|) \) time for any \( x \in [A, \ldots, B] \) if \( W \) is stored in a sorted array \((x, \varphi(x))\), \( x \in W \).

We give some more results on \( K \)-approximation sets and functions which we use in order to obtain our FPTAS for NNV.

\( K \)-approximation sets are very useful for getting succinct approximations for functions that have large domains. The proposition below tells us that when calculating \( \varphi(x) \) as defined in the proposition statement, instead of performing the minimization over the entire set \( C(x) \), one can do so over a subset of logarithmic cardinality while losing only a bit of accuracy.

**Proposition 3.6** (Prop. 5.3 in [HKL+14]). For \( i = 1, \ldots, n \), let \( K_i, L_i \geq 1 \), let \( \varphi_i : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be a function, let \( \hat{\varphi}_i : [A, \ldots, B] \rightarrow \mathbb{R} \) be an \( L_i \)-approximation of \( \varphi_i \), and let \( \psi_i : [A, \ldots, B] \rightarrow [A, \ldots, B] \) be a function such that for any fixed \( x \in [A, \ldots, B] \), \( \hat{\varphi}_i(\psi_i(x, \cdot)) \) is monotone over a contiguous interval \( C(x) \). Let \( m \) be an integer such that \( 1 \leq m \leq n \). For any \( i = 1, \ldots, m \) and any \( x \in [A, \ldots, B] \), let \( W_i(x) \subseteq C(x) \) be a \( K_i \)-approximation set of \( \hat{\varphi}_i(\psi_i(x, \cdot)) \). Let \( \varphi, \hat{\varphi} : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) such that

\[
\varphi(x) = \min_{y \in C(x)} \left\{ \sum_{i=1}^{n} \varphi_i(\psi_i(x, y)) \right\} \quad \text{and} \quad \hat{\varphi}(x) = \min_{y \in \cup_{i=1}^{m} W_i(x)} \left\{ \sum_{i=1}^{n} \hat{\varphi}_i(\psi_i(x, y)) \right\}.
\]

Suppose for every \( x \in [A, \ldots, B] \), the function \( \hat{\varphi}_i(\psi_i(x, \cdot)) \) is monotone in one direction (e.g., increasing) for \( i = 1, \ldots, m \) and is monotone in the other direction (e.g., decreasing) for \( i = m + 1, \ldots, n \). Then, \( \hat{\varphi} \) is a \( \max\{K_1 L_1, \ldots, K_m L_m, L_{m+1}, \ldots, L_n\} \)-approximation of \( \varphi \).

[HOS12] use the notion of \( K \)-approximation sets and functions in order to approximate CDFs and functions that involve expectations via CDF’s. One of their results is:

**Proposition 3.7** (Prop. 2 in [HOS12]). Let \( D \) be an integer-valued random variable and suppose \( \text{Prob}(f(x, D) \geq a_i) \) is monotone in \( x \), \( i = 1, \ldots, n \). Let \( \xi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be a nonnegative increasing function. Let \( K_1, K_2 \geq 1 \), \( \xi(a_0) = 0 \), and let \( S = \{a_1 < \ldots < a_n\} \) be a \( K_1 \)-approximation set of \( \xi \). Finally, let \( \eta_i(x) \) denote \( \text{Prob}(f(x, D) \geq a_i) \) and let \( \hat{\eta}_i(\cdot) \) be a \( K_2 \)-approximation of \( \eta_i(\cdot) \), \( i = 1, \ldots, n \). Then

\[
\hat{\xi}_0(x) = \sum_{i=1}^{n} (\xi(a_i) - \xi(a_{i-1})) \hat{\eta}_i(x) \quad \text{is a} \ K_1 K_2 \text{-approximation of} \ E_D(\xi(f(x, D))).
\]

\(^1\)The range of \( \xi \) in the original statement of Proposition 2 in [HOS12] is \( \mathbb{Z}^+ \). As the proof of the proposition only assumes that the range of \( \xi \) is \( \mathbb{R}^+ \), we choose to state it for this latter range.
Moreover, if $\tilde{F}(\cdot)$ are monotone, then so is $\tilde{\xi}_0(\cdot)$.

By applying Proposition 3.7 with $f(x, D) = x - D$ and noting that $\text{Prob}(x - D \geq a_i) = \text{Prob}(D \leq x - a_i) = F(x - a_i)$ they get an approximation for the expected holding costs $E_D[h(x - D)]$ of NNV, as follows.

**Corollary 3.8** (Corollary 1 in [HOS12]). Let $D$ be an integer-valued random variable and let $F$ be its cumulative distribution function. Let $\xi : [A, \ldots, B] \rightarrow \mathbb{R}^+$ be a nonnegative increasing function. Let $K_1, K_2 \geq 1$, $\xi(a_0) = 0$, and let $S = \{a_1 < \ldots < a_n\}$ be a $K_1$-approximation set of $\xi$. Finally, let $\tilde{F}$ be a $K_2$-approximation of $F$. Then

$$\tilde{\xi}_2(x) = \sum_{i=1}^{n}(\xi(a_i) - \xi(a_{i-1}))\tilde{F}(x - a_i) \text{ is a } K_1K_2\text{-approximation of } E_D(\xi(x - D)).$$

Moreover, if $\tilde{F}(\cdot)$ is increasing, then so is $\tilde{\xi}_2(\cdot)$.

By applying Proposition 3.7 with $f(x, D) = D - x$ and noting that $\text{Prob}(D - x \geq a_i) = \text{Prob}(D \geq x + a_i) = 1 - F(x + a_i - 1)$ we get an approximation for the expected lost sales $E_D[b(D - x)]$ of NNV, as follows.

**Corollary 3.9.** Let $D$ be an integer-valued random variable and let $F$ be its cumulative distribution function. Let $\xi : [A, \ldots, B] \rightarrow \mathbb{R}^+$ be a nonnegative increasing function. Let $K_1, K_2 \geq 1$, $\xi(a_0) = 0$, and let $S = \{a_1 < \ldots < a_n\}$ be a $K_1$-approximation set of $\xi$. Finally, let $\tilde{F}^c$ be a $K_2$-approximation of $1 - F$. Then

$$\tilde{\xi}_1(x) = \sum_{i=1}^{n}(\xi(a_i) - \xi(a_{i-1}))\tilde{F}^c(x + a_i - 1) \text{ is a } K_1K_2\text{-approximation of } E_D(\xi(D - x)).$$

Moreover, if $\tilde{F}^c(\cdot)$ is decreasing, then so is $\tilde{\xi}_1(\cdot)$.

### 3.1 Design and analysis of the approximation scheme

We are now ready to prove Theorem 1.5, that is to state and analyze an FPTAS for NNV, see Algorithm 1. (Recall that the problem input size is $O(\log |A| + \log |B| + \log M + \log \gamma)$, where $M$ is a bound on the cost functions and $\gamma$ is a bound on the CDF of the demand as defined in the Introduction. From here on after we use the notation $z(\cdot)$, where the “$\cdot$” stands for the argument of function $z$. E.g., the value of $z(\cdot - D)$ for variable value 2 is $z(2 - D)$. Put it differently, the function $z$ is shifted by $-D$.)

1. **Procedure FPTASNNV($\varepsilon$)**
   2. $K' \leftarrow 1 + \varepsilon/4$, $S^b \leftarrow \text{APXSET}(b, K')$, $S^h \leftarrow \text{APXSET}(h, K')$
   3. Let $\tilde{\xi}_1$ be as defined in Corollary 3.9 with parameters set to $\xi = b$, $S = S^b$, $K_1 = K'$ and $K_2 = 1$
   4. $\tilde{\phi}_1(\cdot) = E_D[b(D - \cdot)]$
   5. $\tilde{\phi}_2(\cdot) = E_D[h(\cdot - D)]$
   6. Return $\tilde{\phi}(0)$ as defined in Proposition 3.6 with parameters set to $n = 3$, $m = 1$, $\varphi_1(\cdot) = E_D[b(D - \cdot)]$, $\varphi_3(\cdot) = c, \tilde{\phi}_1 = \tilde{\xi}_1, \tilde{\phi}_2 = \tilde{\xi}_2, \tilde{\phi}_3 = c, \psi_i(x, y) = y, i = 1, \ldots, 3$, $L_1 = L_2 = K_1 = K', W_1 = S_1$ and $L_3 = 1$

**Algorithm 1:** An FPTAS for NNV. The demand $D$ has support $[A, \ldots, B]$ and bound $\gamma$ on its CDF $F$ and is given by an oracle to $F$. 

11
Proof. (of Theorem 1.5) Algorithm 1 starts with building $K$-approximation oracles for $E_D[b(D - \cdot)]$ and $E_D[h(-D)]$. Applying Corollary 3.9 we get in line 3 that $\hat{\xi}_1(\cdot)$ is a decreasing $K'$-approximation oracle of $E_D[b(D - \cdot)]$. Similarly, applying Corollary 3.8 we get in line 4 that $\hat{\xi}_2(\cdot)$ is an increasing $K'$-approximation oracle of $E_D[h(-D)]$. Applying Proposition 3.6, the value $\tilde{\varphi}(0)$ returned in line 6 is a $(K')^2$-approximation for NNV. Because $(1 + \epsilon/4)^2 \leq 1 + \epsilon$ we get that $\tilde{\varphi}(0)$ is a $(1 + \epsilon)$-approximation for NNV as needed, i.e., $\tilde{\varphi}(0)$ returned by the algorithm is a $(1 + \epsilon)$-approximation of (1).

We now analyze the running time of the approximation algorithm. Let $M = \max\{h(B - A)/\gamma, b(B)/\gamma\}$ be an upper bound on the ratio between the maximal and minimal possible values of $h$ and $b$. By Proposition 3.5, the cardinality of $S^h, S^b$ is $O(\log_K M)$, and the time needed to build each one of these sets is $O((1 + t) \log_K M \log(B - A))$, where $t = t_h, t_b$, respectively. By the same proposition, the cardinality of $S_1$ is $O(\log_K(M/\gamma))$ (note that if $x'$ is the minimum argument for which $b(x') > 0$, then the minimum strictly positive value of $E_D[b(D - \cdot)]$ is $E_D[b(D - B + x')] = b(x')\text{Prob}(D = B) \geq \gamma b^{\min}$), and the time needed to build $S_1$ is $O((1 + t_{\xi_1}) \log_K(M/\gamma) \log(B - A))$. Due to Corollaries 3.8 and 3.9, each query to $\hat{\xi}_i$ takes $t_{\hat{\xi}_i} = O((1 + t_F + t_e) \log_K M)$ time, $i = 1, 2$. Overall, the time needed to calculate a $(1 + \epsilon)$-approximation for NNV is bounded by $O((1 + t_h + t_e + t_h + t_F) \log_K M \log_K(M/\gamma) \log(B - A)) = O\left((1 + t_b + t_e + t_h + t_F) \log M \log \log(1/\delta) \log(B - A)\right)$ (we use here the equation $\log_K M = O(\frac{\log M}{\epsilon})$), i.e., polynomial in the (binary) input size and $1/\epsilon$.

\section{An approximation scheme for SNNV}

In this section, we design a provably near-optimal solution to SNNV, i.e., prove Theorem 1.6.

\subsection{Approximating cumulative distribution functions via sampling}

We approximate the CDF of an integer-valued random variable $D$ via the empirical distribution function.

**Definition 4.1.** The empirical distribution function $\hat{F}_N$ of a random variable $D$ is the CDF that puts mass $1/N$ at each data point $D_i$. Formally,

$$\hat{F}_N(x) = \frac{1}{N} \sum_{i=1}^{N} I_{D_i \leq x} \quad \text{where} \quad I_{D_i \leq x} = \begin{cases} 1 & \text{if } D_i \leq x; \\ 0 & \text{otherwise}. \end{cases}$$

(4)

The following theorem bounds the (absolute) distance between the true CDF $F$ of $D$ and the one of its empirical distribution function.

**Theorem 4.2** (Dvoretzky-Kiefer-Wolfowitz (DKW) inequality, chap. 3.3 in [DL01]). For any $\epsilon > 0$

$$\text{Prob}\left(\sup_x |F(x) - \hat{F}_N(x)| > \epsilon \right) \leq 2e^{-2\epsilon^2 N}.$$

We next construct a (one-sided multiplicative) $K$-approximation for $F$ ($K = 1 + \epsilon$). We slightly adjust the empirical distribution function to $\bar{F}_N(\cdot) = (1 + \epsilon/2)\hat{F}_N(\cdot)$ and call $\bar{F}_N$ the adjusted empirical distribution function. (Note that here we slightly abuse notation because $\bar{F}_N$ depends on $\epsilon$.) The next proposition uses the two-sided absolute error in (DKW) inequality in order to get a one-sided relative error.

**Proposition 4.3.** Let $D$ be a random variable with a given support $[A, \ldots, B]$. Suppose there exists a positive real number $\gamma > 0$ such that $\text{Prob}(D = A), \text{Prob}(D = B) \geq \gamma$. For every $\epsilon' > 0$, the adjusted empirical distribution function $\bar{F}_N$ satisfies

$$\min_{x \in [A, \ldots, B]} \text{Prob} \left( F(x) \leq \bar{F}_N(x) \leq (1 + \epsilon') F(x) \right) \geq 1 - \delta',$$

where $\delta' = 2e^{-2\epsilon'^2 N}$. 

12
where \( \delta' = 2e^{-2(\frac{\epsilon'}{2+\epsilon'})^2}\gamma^2N \). I.e., with confidence \( 1 - \delta' \), \( \bar{F}_N(\cdot) \) is an increasing \( (1 + \epsilon') \)-approximation function of \( F(\cdot) \).

**Proof.** We first note that \( \bar{F}_N \) is an increasing function. Using DKW inequality with \( \epsilon = (\epsilon'\gamma)/(2+\epsilon') \) we get that with probability at least \( 1 - \delta' \) the second and third inequalities in the following term are valid for all values of \( x \) (the first and last inequalities hold always):

\[
\frac{2}{2+\epsilon'} F(x) = \left( 1 - \frac{\epsilon'\gamma}{2+\epsilon'} \right) F(x) \leq F(x) - \frac{\epsilon'\gamma}{2+\epsilon'} \leq \bar{F}_N(x) \leq F(x) + \frac{\epsilon'\gamma}{2+\epsilon'} \leq \left( 1 + \frac{\epsilon'}{2+\epsilon'} \right) F(x).
\]

Dividing this term by \( \frac{2}{2+\epsilon'} \) we get that with probability at least \( 1 - \delta' \), \( \bar{F}_N(\cdot) \) is an increasing \( (1 + \epsilon') \)-approximation of \( F(\cdot) \).

**Corollary 4.4.** Let \( D \) be a random variable with a given support \([A, \ldots, B]\). For every relative error \( \epsilon' > 0 \), uncertainty parameter \( \delta' > 0 \), bound \( \gamma > 0 \) on the CDF of \( D \), and \( N \) independent samples of \( D \), \( N \geq \frac{\log(2(2')/(2+\epsilon'))^2}{2\gamma^2\epsilon^2} \), the adjusted empirical distribution function \( \bar{F}_N(\cdot) \) is with confidence \( 1 - \delta \) an increasing \( (1 + \epsilon') \)-approximation function of \( F(\cdot) \).

Before moving to the design of the approximation scheme for SNNV, we need to design an efficient construction of a succinct representation of (an approximation for) the CDF of the demand. More generally, we define a function COMPRESSION for the efficient construction of a succinct representation of an increasing function \( \varphi : [A, \ldots, B] \to \mathbb{R}^+ \) that can be accessed directly as follows (for simplicity, we omit the parameter \([A, \ldots, B]\) in COMPRESSION whenever the domain of the function is obvious from the context).

```
1: Function COMPRESSION([\varphi], [A, \ldots, B], K)
2: W \leftarrow \text{APXSET}([\varphi], [A, \ldots, B], K)
3: return \{(x, \varphi(x)) \mid x \in W\} as a sorted array
```

**Algorithm 2:** FUNCTION COMPRESSION returns a succinct representation of a step \( K \)-approximation of an increasing function \( \varphi \).

We similarly define function COMPRESSION for decreasing functions and define function COMPRESSION when \( \varphi \) is increasing and to be COMPRESSION when \( \varphi \) is decreasing. Applying Proposition 3.5 coupled with approximation of approximation (Proposition 3.2(7)) yields the following result.

**Proposition 4.5.** Let \( K, L > 1 \) be real numbers and let \( \varphi, \varphi' : [A, \ldots, B] \to \mathbb{R}^+ \) be monotone functions over \([A, \ldots, B]\). Then function \( \text{COMPRESSION}([\varphi], [A, \ldots, B], K) \) returns in \( O(t_{\varphi} \log K \frac{\varphi_{\max}}{\varphi_{\min}} \log(B - A)) \) time a succinct representation of a step \( K \)-approximation function of \( \varphi \) with \( O(\log K \frac{\varphi_{\max}}{\varphi_{\min}}) \) steps that has query time \( O(\log K \frac{\varphi_{\max}}{\varphi_{\min}}) \). If \( \varphi \) is an \( L \)-approximation function of \( \varphi' \), then the returned succinct representation is also a \( KL \)-approximation of \( \varphi' \).

### 4.2 Design and analysis of the approximation scheme

We are now ready to state and analyze our approximation scheme, see Algorithm 3. A main difficulty in proving Theorem 1.6 is the construction of a \( K \)-approximation for the CDF \( F(\cdot) \) of the random variable \( D \). This stems from the fact that we don’t have direct access to \( F \). Therefore, we will use the adjusted empirical distribution function \( \bar{F}_N \) as an oracle for \( F \). For any fixed \( x \), note that since \( \bar{F}_N \) is constructed via samples, there is always some positive probability that \( \bar{F}_N(x) \) is “far” from \( F(x) \). Therefore, it is generally not possible to build a \( K \)-approximation for \( F \) that is always correct. For this reason we aim at constructing a function \( \bar{F} \) that with high probability will be a \( K \)-approximation of \( F \) for all values of \( x \).
Proof. (of Theorem 1.6) We start by analyzing the approximation ratio. Similarly to Algorithm 1 the algorithm starts by constructing $K'$-approximation sets $S^b, S^h$ for $b(\cdot), h(\cdot)$. Because we don’t have direct access to $F$, the algorithm uses the adjusted empirical distribution function $\bar{F}_N$ instead. By applying Corollary 4.4 with parameters set to $e' = \epsilon/2$ and $d' = \delta/2$, we get with confidence $1 - \delta/2$ that $\bar{F}_N(\cdot)$ is an increasing $K''$-approximation function of $F(\cdot)$. Therefore, by Proposition 4.5 with parameters set to $K = K', L = K''$, $\varphi = \bar{F}_N, \varphi' = F$, we get an oracle $\bar{F}_N$ that is an increasing $K'$-approximation function of $\bar{F}_N$ and with confidence $1 - \delta/2$, is also a $K'K''$-approximation function of $F(\cdot)$. The same holds for the construction of the oracle $\bar{F}^C_N$, that with confidence $1 - \delta/2$ is a $K''K'$-approximation function of $1 - F(\cdot)$. Applying Corollary 3.9 we get in line 8 that $\tilde{x}_1$ is a decreasing $(K')^2K''$-approximation of $E_D[b(D - x)]$ with confidence $1 - \delta/2$. Similarly, by applying Corollary 3.8 we get in line 9 that $\tilde{x}_2$ is an increasing $(K')^2K''$-approximation of $E_D[h(x - D)]$ with confidence $1 - \delta/2$. Therefore, by union bound we get that with confidence $1 - \delta$ both $\tilde{x}_1$ and $\tilde{x}_2$ are $(K')^2K''$-approximation functions of $E_D[b(D - x)]$ and $E_D[h(x - D)]$, respectively. The algorithm then constructs in line 10 a $K'$-approximation set $S_1$ of $\tilde{x}_1$. We last apply Proposition 3.6 to get that $\tilde{\varphi}(0)$ returned by line 11 is with confidence $1 - \delta$ a $(K')^3K''$-approximation for SNNV. Substituting $K'$ and $K''$ with their values we get a $(1 + \epsilon)$-approximation as needed.

We next analyze the running time. Recall that $M$ is an upper bound on the ratio between the maximal and minimal values of the NNV cost functions. Regarding line 5, by Proposition 3.5, the cardinality of each $S^b, S^h$ is $O(\log_K M)$, and the time needed to build both of these sets is $O((1 + b_h + t_h) \log_K M \log(B - A))$. Regarding line 7, by Proposition 4.5, the time needed to build succinct representations for the value oracles $\bar{F}_N, \bar{F}^C_N$ is $O((1 + t_{F_N}) \log_K(1/\gamma) \log(B - A))$ with query time $t_{\bar{F}_N} = t_{\bar{F}^C_N} = O(\log \log_K(1/\gamma))$. By sorting the $N$ observations of $D$, each query to $\bar{F}_N$ takes $t_{\bar{F}_N} = O(\log N)$ time. Considering lines 8-9, by corollaries 3.9-3.8 each query to $\tilde{x}_1, \tilde{x}_2$ takes $O((1 + t_b + t_h + t_{\bar{F}_N}) \log_K M)$ time. We now turn to analyze the time needed to build $S_1$ in line 10. Again, by Proposition 3.5, the cardinality of $S_1$ is $|S_1| = O(\log_K(M/\gamma))$, and the time needed to build $S_1$ is $O((1 + t_{\tilde{x}_1}) \log_K(M/\gamma) \log(B - A)) = O((1 + t_b + \log_K(1/\gamma) \log_K M \log_K(M/\gamma) \log(B - A))$. We turn now to the last line of the algorithm. Due to Proposition 3.6, evaluating $\tilde{\varphi}(0)$ takes $O((1 + t_{\tilde{x}_1} + t_{\tilde{x}_2} + t_c)|S_1|) = O((1 + t_c + t_b + t_h + \log \log_K(1/\gamma) \log_K M \log_K(M/\gamma))$. Overall, the time needed to calculate a $(1 + \epsilon)$-approximation for

1: **Procedure FPTASSNNV** $(N, \epsilon, \delta)$
2: if $N < \frac{2\log \frac{2}{\epsilon}}{\epsilon^2}$ then
3: return “not enough samples of demand to calculate $(1 + \epsilon)$-approximation with confidence $1 - \delta$”
4: end if
5: $K' \leftarrow 1 + \epsilon/10$, $K'' \leftarrow 1 + \epsilon/2$, $S^b \leftarrow \text{APXSET}(b,K')$, $S^h \leftarrow \text{APXSET}(h,K')$
6: Let $\bar{F}_N$ be the adjusted empirical distribution function based on the $N$ samples and accuracy $\epsilon/2$
7: $\bar{F}_N \leftarrow \text{COMPRESSINC}(\bar{F}_N, K')$, $\bar{F}^C_N \leftarrow \text{COMPRESSDEC}(1 - \bar{F}_N, K')$
8: Let $\tilde{x}_1$ be as defined in Corollary 3.9 with parameters set to $\xi = b$, $\bar{F}^C = \bar{F}^C_N$, $S = S^b$, $K_1 = K'$ and $K_2 = K'K''$ /* $\tilde{x}_1(\cdot)$ is a $(K')^2K''$-approximation of $E_D[b(D - \cdot)]$ with confidence $1 - \delta/2$ */
9: Let $\tilde{x}_2$ be as defined in Corollary 3.8 with parameters set to $\xi = h$, $\bar{F} = \bar{F}_N$, $S = S^h$, $K_1 = K'$ and $K_2 = K'K''$ /* $\tilde{x}_2(\cdot)$ is a $(K')^2K''$-approximation of $E_D[h(\cdot - D)]$ with confidence $1 - \delta/2$ */
10: $S_1 \leftarrow \text{APXSET}(\tilde{x}_1, K')$
11: return $\tilde{\varphi}(0)$ as defined in Proposition 3.6 with parameters set to $n = 3$, $m = 1$, $\varphi_1(\cdot) = E_D[b(D - \cdot)]$, $\varphi_2(\cdot) = E_D[h(\cdot - D)]$, $\varphi_3 = c$, $\varphi_1 = \tilde{x}_1$, $\varphi_2 = \tilde{x}_2$, $\varphi_3 = c$, $\psi_i(x,y) = y$, $i = 1, \ldots, 3$, $K_1 = K'$, $W_1 = S_1$ and $L_1 = L_2 = (K')^2K''$, $L_3 = 1$
We give some notation first. Let $SNNV$ be bounded by

$$O \left( (1 + t_{e} + t_{b} + t_{h} + \log \frac{\log(1/\gamma)}{\epsilon}) \log M (\log M + \log(1/\gamma)) \log(B - A) + \log N \left( \frac{\log(B - A) \log(1/\gamma)}{\epsilon} \right) \right),$$

i.e., polynomial in the (binary) input size, in $1/\epsilon$, $1/\gamma$ and logarithmic in $1/\delta$. \hfill $\square$

**Remark.** Applying Theorem 1.6 to SNV (which is a special case of SNNV) and setting $A = 0$, $B = 1$ and $\gamma = \frac{\min(b,h)}{b+h}$, we get that the number of samples taken by the algorithm is identical to the one of [LRS07], up to a constant factor. Note that given historical data with fixed size $N$, our bound on the number of samples $N$ sufficient for our approximation scheme to work indicates what kind of guarantees one can hope for.

## 5 Explicit stochastic DP

We give some notation first. Let $D_{\text{min}} = \{ \min d \mid \text{Prob}(D = d) > 0 \}$, and $D_{\text{max}} = \{ \max d \mid \text{Prob}(D = d) > 0 \}$. Let $X$ be a contiguous interval, and $Y(x)$ a nonempty contiguous interval for all $x \in X$. The set $X \otimes Y = \bigcup_{x \in X} \{ x \}, Y(x) \subset \mathbb{Z}^{2}$ is integrally convex if there exists a polyhedron $C_{XY} \subset \mathbb{R}^{2}$ such that: $X \otimes Y = C_{XY} \cap \mathbb{Z}^{2}$, and the slope of the edges of $C_{XY}$ is an integer multiple of $45^\circ$.

In this section we review the FPTAS framework for explicit stochastic DP as defined in [HKL+14]. They assume the following three conditions hold.

**Condition 1.** $S_{T+1}, S_{t}, A_{t}(I_{t}) \subset \mathbb{Z}$, for $I_{t} \subset S_{t}$ and $t = 1, \ldots, T$. For any set $X$ among these sets, $\log \max_{x \in X}(|x| + 1)$ is bounded polynomially by the (binary) input size, and the $k$th largest element in $X$ can be identified in constant time for any $1 \leq k \leq |X|$. For each $D_{t}$, the number of different values it admits with positive probability is a given integer $n_{t}$, i.e., its support is $D_{t} := \{ d_{t,1}, \ldots, d_{t,n_{t}} \}$, where $d_{t,i} < d_{t,j}$ for $i < j$. For every $d \in D_{t}$ a positive rational probability $\text{Prob}(D_{t} = d)$ is given, i.e., the discrete random variable $D_{t}$ is given explicitly as a set of $n_{t}$ ordered pairs $(d, \text{Prob}(D_{t} = d))$. Last, $D_{t} \subset \mathbb{Q}$ for $t = 1, \ldots, T$.

**Condition 2.** For every $t = 1, \ldots, T+1$, functions $f_{t}, g_{t}$ are either given explicitly (i.e., as explicit formulae) or accessed via oracle calls. Moreover, the values of $g_{t}$ are polynomially bounded by the (binary) size of the input.

**Condition 3.** At least one of the following properties holds:

(i) **Increasing DP** Function $g_{T+1}(\cdot)$ is increasing. For $t = 1, \ldots, T$, function $f_{t}(I, x, d)$ is increasing in $I$ and monotone in $x$, and $g_{t}(I, x, d)$ is monotone in $x$. Moreover, for each $t = 1, \ldots, T$, either $z_{t}(\cdot)$ is increasing, or $g_{t}(I, x, d)$ is increasing in $I$ and $A_{t}(I) \subseteq A_{t}(I')$ for all $I, I' \in S_{t}$ with $I \geq I'$.

(ii) **Decreasing DP** Function $g_{T+1}(\cdot)$ is decreasing. For $t = 1, \ldots, T$, function $f_{t}(I, x, d)$ is increasing in $I$ and monotone in $x$, and $g_{t}(I, x, d)$ is monotone in $x$. Moreover, for each $t = 1, \ldots, T$, either $z_{t}(\cdot)$ is decreasing, or $g_{t}(I, x, d)$ is decreasing in $I$ and $A_{t}(I) \subseteq A_{t}(I')$ for all $I, I' \in S_{t}$ with $I \leq I'$.

(iii) **Convex DP** The terminal state space $S_{T+1}$ is a contiguous interval. For $t = 1, \ldots, T$, the state space $S_{t}$ and the action space $A_{t}(I)$, $\forall I \in S_{t}$, are both contiguous intervals, and $D_{t} \subset \mathbb{Z}$. Function $g_{T+1}(\cdot)$ is convex over $S_{T+1}$. For $t = 1, \ldots, T$, the set $S_{t} \otimes A_{t}$ is integrally convex, function $g_{t}$ can be expressed as $g_{t}(I, x, d) = g_{t}^{I}(I, d) + g_{t}^{x}(x, d) + u_{t}(f_{t}(I, x, d))$, and function $f_{t}$ can be expressed as $f_{t}(I, x, d) = a(d)I + bx + c(d)$, where $g_{t}^{I}(\cdot, d), g_{t}^{x}(\cdot, d), u_{t}(\cdot)$ are univariate convex functions, $a(d) \in \mathbb{Z}$, $b \in \{-1, 0, 1\}$, and $c(d) \in \mathbb{Z}$. 

15
The input data of the problem consists of the number of time periods $T$, the initial state $I_1$, and the explicit description of the random variables as described in Condition 1.

DP formulation (3) that satisfies either Condition 3(i) or Condition 3(ii) is called monotone, and convex whenever it satisfies Condition 3(iii).

**Theorem 5.1** (Thm. 3.3 in [HKL+14]). Every explicit stochastic DP satisfying Conditions 1, 2 and 3 admits an FPTAS.

We note that even though Conditions 1, 2 and 3 may look burdensome, some of their components cannot be relaxed. In particular, [HKL+14] show that a Convex DP where either $b \notin \{-1, 0, 1\}$ or $S_t \otimes A_t$ is not integrally convex, does not admit an FPTAS under the standard complexity assumption that $P \neq NP$.

The challenge is, therefore, to relax Conditions 1, 2 and 3 while still having an FPTAS. For example, [HKL+14, Sec. 10.5] extend the FPTAS to the case where instead of direct access to the (exact) oracles $g_t$, one has an FPTAS for them.

In the next two sections we will relax the assumption on the representation of the random variables $D_t$ as stated in Condition 1. That is, in Section 6 we will permit implicitly defined RVs and in Section 7 we will treat random variables the are accessed via samples.

### 6 Implicit stochastic DP

The main difficulty in the implicit stochastic model lies in the fact that while in the explicit stochastic model the complexity of calculating expectations is polynomial in the input size, i.e., linear in the size of the support of the random variable, calculating expectations in the implicit stochastic model may be intractable. E.g., considering the standard (linear) NV, where the demand distribution is given implicitly as a value oracle to the demand CDF, while the optimal order quantity is defined by the critical ratio (see Introduction) and is polynomially computable via binary search, calculating the expected value of the optimal solution may require exponential number of function evaluations [HOS12, Thm. 1].

Specifically, in the explicit stochastic model the following two propositions are used to build approximate value oracles for the cost-to-go functions of monotone (convex) DPs, respectively (recall from Condition 1 that $n_t$ is the support size of $D_t$):

**Proposition 6.1** (Prop. 7.1 in [HKL+14]). Suppose the DP formulation (3) satisfy Condition 3(i) or Condition 3(ii). Let $K'$, $L'$, $L''$, $t$, and $I_t$ be fixed values, where $K' \geq 1$, $1 \leq L'' \leq K'L'$, $I_t \in S_t$, and $t \in [1, \ldots, T]$. Let $g_t$ be as stated in Conditions 3(i) and 3(ii). Let $\bar{z}_{t+1}$ be a monotone $L'$-approximation of $z_{t+1}$, and $W$ be a $K'$-approximation set of $\bar{z}_{t+1}$. Let $W^{-1}(I_t) = \bigcup_{k=1}^{n_t} f^{-1}_{t,k}(I_t, W)$, where $f^{-1}_{t,k}(I_t, W) = \{ x_k \mid f_t(I_t, x_k, d_{t,i}) \leq w \}, \{ x_k \mid f_t(I_t, x_k, d_{t,i}) \geq w \} \mid w \in W \}$ if $f_t$ is increasing in its second variable, and $f^{-1}_{t,k}(I_t, W) = \{ x_k \mid f_t(I_t, x_k, d_{t,i}) \leq w \}, \{ x_k \mid f_t(I_t, x_k, d_{t,i}) \geq w \} \mid w \in W \}$ if $f_t$ is decreasing in its second variable. Let $\bar{g}_t(I_t\cdot, D_t)$ be a monotone $L''$-approximation of $g_t(I_t\cdot, D_t)$. Let

$$\bar{z}_t(I_t) = \min_{z_t \in W^{-1}(I_t)} E_{D_t} \{ \bar{g}_t(I_t, x_t, D_t) + \bar{z}_{t+1}(f_t(I_t, x_t, D_t)) \}.$$  

Then, $\bar{z}_t(I_t)$ is an unnecessarily monotone $K'L'$-approximation value of $z_t(I_t)$, and it can be determined in $O(n_t(t_{\bar{g}} + t_{f} + t_{\bar{z}_{t+1}})|W^{-1}(I_t)|)$ time if the elements of $W^{-1}(I_t)$ are given.

**Proposition 6.2** (based on Props. 7.2 and 9.1 in [HKL+14]). Suppose the DP formulation (3) is convex (so Condition 3(iii) is satisfied). Let $K'$, $L''$, $t$, and $I_t$ be fixed values, where $K'' \geq K' \geq 1$, $I_t \in S_t$, and $t \in [1, \ldots, T]$. Let $\bar{g}_t(I_t\cdot, d_{t,i})$ be a convex $K'$-approximation function of $g_t(I_t\cdot, d_{t,i})$ for every $i = 1, \ldots, n_t$. Let $\bar{z}_{t+1}$ be a convex $K''$-approximation function of $z_{t+1}$. Let

$$\bar{z}_t(I_t) = \min_{x_t \in A_t(I_t)} E_{D_t} \{ \bar{g}_t(I_t, x_t, D_t) + \bar{z}_{t+1}(f_t(I_t, x_t, D_t)) \}.$$  

16
Then, \( \bar{z}_t(I_t) \) is a convex \( K'' \)-approximation value of \( z_t(I_t) \) and can be determined in \( O(n_t(t_{\bar{g}} + t_f + t_{\bar{z}_{t+1}}) \log |A_t(I_t)|) \) time.

A sufficient condition for approximating the expected value of a single-period cost function, i.e., for approximating \( E_{D_t} g_t(I_t, x_t, D_t) \) is to assume that \( g_t \) has some special structure.

**Condition 4.** \( g_t \) can be expressed as

\[
g_t(I, x, d) = g_t^L(I, x) + g_t^I(f_t^I(I, d)) + g_t^D(f_t^D(x, d)) + g_t^u(f_t^u(I, x, d)),
\]

where for every \( \diamond \in \{I, x, d\} \), \( g_t^\diamond \) is monotone. For \( \diamond \in \{I, x\} \), \( f_t^\diamond \) is either linear and separable in its variables or \( \text{Prob}(f_t^\diamond(\cdot, D) \geq a) \) is monotone for every fixed \( a \) and can be expressed as a linear and separable function of \( F_t \) multiplied by an indicator on the relation between its variable and \( a \). Furthermore, \( f_t^x \) is either linear and separable in its variables or \( \text{Prob}(f_t^x(I, x, D) \geq a) \) is monotone in \( x \) for every fixed \( I, a \) and can be expressed as a linear and separable function of \( F_t \) multiplied by an indicator on the relation between \( I, x, a \).

Note that multi-period NV satisfies this condition when parameters are set to \( g_t^L(x) = c_t(x), f_t^I \equiv f_t^x \equiv g_t^I \equiv g_t^x \equiv 0, g_t^u(\cdot) = h_t(\cdot) \) and \( f_t^u(I, x, d) = I + x - d \). Note also that \( \text{Prob}(f_t^u(I, x, D) \geq a) = F_D(I + x - a) \) and is therefore monotone increasing in \( x \). (Another example of a problem formulation that satisfies this condition is given in Appendix B.4.)

### 6.1 Model statement

We design an FPTAS for implicit stochastic DPs that satisfy the following three conditions.

**Condition 5.** \( S_{T+1}, S_t, A_t(I_t) \subseteq \mathbb{Z} \) for \( I_t \in S_t \) and \( t = 1, \ldots, T \). For any set \( X \) among these sets, \( \log \max_{x \in X} (|x| + 1) \) is bounded polynomially by the (binary) input size, and the \( k \)th largest element in \( X \) can be identified in constant time for any \( 1 \leq k \leq |X| \). For each \( t = 1, \ldots, T \), the support of \( D_t \) is contained in \( [A_t, \ldots, B_t] \).

**Condition 6.** For every \( t = 1, \ldots, T + 1 \), functions \( f_t, g_t \) and the CDF \( F_t \) of \( D_t \) are either given explicitly (i.e., as explicit formulae) or accessed via value oracles. Moreover, the values of \( g_t, F_t \) are polynomially bounded by the (binary) size of the input.

**Condition 7.** At least one of the following properties holds:

(i) **(Increasing DP)** Function \( g_{T+1}(\cdot) \) is increasing. For \( t = 1, \ldots, T \), function \( f_t(I, x, d) \) is increasing in \( I \) and monotone in \( x \), and \( g_t(I, x, d) \) is monotone in \( x \). Either monotone FPTASs to calculate \( E_{D_t} g_t(I, \cdot, D) \) are given for any fixed \( I \), or Condition 4 is met. \( f_t(I, x, D) \) is either linear and separable in its variables, or \( \text{Prob}(f_t(I, x, D) \geq a) \) is monotone in \( x \) for every fixed \( a, I \) and can be expressed as a linear and separable function of \( F_t \) multiplied by an indicator on the relation between \( I, x, a \). Moreover, for each \( t = 1, \ldots, T \), either \( z_t(\cdot) \) is increasing, or \( g_t(I, x, d) \) is increasing in \( I \) and \( A_t(I) \subseteq A_t(I') \) for all \( I, I' \in S_t \) with \( I \geq I' \).

(ii) **(Decreasing DP)** Function \( g_{T+1}(\cdot) \) is decreasing. For \( t = 1, \ldots, T \), function \( f_t(I, x, d) \) is increasing in \( I \) and monotone in \( x \), and \( g_t(I, x, d) \) is monotone in \( x \). Either monotone FPTASs to calculate \( E_{D_t} g_t(I, \cdot, D) \) are given for any fixed \( I \), or Condition 4 is met. \( f_t(I, x, D) \) is either linear and separable in its variables, or \( \text{Prob}(f_t(I, x, D) \geq a) \) is monotone in \( x \) for every fixed \( a, I \) and can be expressed as a linear and separable function of \( F_t \) multiplied by an indicator on the relation between \( I, x, a \). Moreover, for each \( t = 1, \ldots, T \), either \( z_t(\cdot) \) is decreasing, or \( g_t(I, x, d) \) is decreasing in \( I \) and \( A_t(I) \subseteq A_t(I') \) for all \( I, I' \in S_t \) with \( I \leq I' \).

17
(iii) **(Convex DP)** The terminal state space $S_{T+1}$ is a contiguous interval. For $t = 1, \ldots, T$, the state space $S_t$ and the action space $A_t(I)$, $\forall I \in S_t$, are both contiguous intervals, and $D_t \subset \mathbb{Z}$. Moreover, function $f_t$ can be expressed as $f_t(I, x, d) = aI + bx + cd$, where $a(d) \in \mathbb{Z}$ and $b, c \in \{-1, 0, 1\}$. Function $g_{T+1}^t(\cdot)$ is convex over $S_{T+1}$. For $t = 1, \ldots, T$, the set $S_t \otimes A_t$ is integrally convex.

Function $g_t$ can be expressed as $g_t(I, x, d) = g_t^1(f_t(I, d)) + g_t^2(f_t(x, d)) + g_t^3(f_t(I, x, d))$, where $g_t^1(\cdot), g_t^2(\cdot), g_t^3(\cdot)$ are univariate convex functions and $f_t^1, f_t^2$ are linear and separable functions in their variables.

In our analysis, the following notation will be used:

$$
\begin{align*}
\gamma_t &= \min \{ \text{Prob}(D_t^{\max}), \text{Prob}(D_t^{\min}) \}, \ t = 1, \ldots, T = \text{bound on the CDF of } D_t; \\
\gamma &= \min \gamma_t = \text{bound on all CDFs}; \\
U_S &= \max_{t=1,\ldots,T+1} |S_t| = \text{maximal size of the state space}; \\
U_A &= \max_{t=1,\ldots,T} \max_{I \in S_t} |A_t(I)| = \text{maximal size of the action space}; \\
U_D &= \max_{t=1,\ldots,T} (B_t - A_t + 1) = \text{maximal size of a support of a RV}.
\end{align*}
$$

The input data of the problem consists of the number of time periods $T$, the initial state $I_1$, the constant $\gamma$, and endpoints $A_t, B_t$ of the supports of the random variables.

We note that a rather broad class of discrete distributions falls under this model. E.g., when the demand distribution $D$ is Poisson with rate $\lambda$ that is truncated from above by some $m > \lambda$, i.e.,

$$
\begin{align*}
\text{Prob}(D = x) = \begin{cases} 
\text{Prob}(P = x) & \text{if } x < m, \\
1 - \sum_{i=0}^{m-1} \text{Prob}(P = i) & \text{if } x = m, \\
0 & \text{if } x > m,
\end{cases}
\end{align*}
$$

where $P \sim \text{Pois}(\lambda)$. Then $A = 0$, $B = m$ and $\gamma \approx \min \{e^{-\lambda}, e^{-\lambda}(e\lambda)^m/m^m\}$. Or when the demand distribution is discrete Gaussian with mean 200,000 and standard deviation 100,000 truncated from below by 0 and from above by 400,000, i.e.,

$$
\begin{align*}
\text{Prob}(D = x) = \begin{cases} 
\text{Prob}(Z < x + 0.5) - \text{Prob}(Z > x - 0.5) & \text{if } 0 < x < 400,000, \\
\text{Prob}(Z < 0.5) & \text{if } x = 0, \\
\text{Prob}(Z > 399,999.5) & \text{if } x = 400,000,
\end{cases}
\end{align*}
$$

where $Z$ is Gaussian with mean 200,000 and standard deviation 100,000. Then $A = 0$, $B = 400,000$ and $\gamma \approx 0.025$.

[HKL+14, Tab. 2] showed seven stochastic problems to fit into their FPTAS framework. The distribution of the random variables in these problems was assumed to be given explicitly. In Appendix B we consider these problems under the less restrictive assumption of implicitly-defined random variables, and give for them formulations that satisfy Conditions 5, 6 and 7. Consequently, due to Theorem 1.2 all these problems admit an FPTAS.

We note that the FPTAS framework in Theorem 1.2 holds also if instead of direct access to the (exact) CDF of the random variables, one has an FPTAS for them. (In the proof of Theorem 1.2 we will apply Proposition 3.7 with $K_2 > 1$ instead of $K_2 = 1$.) Furthermore, the FPTAS framework in Theorem 1.2 holds also if instead of direct access to the (exact) cost functions $g_t$, one has an FPTAS for them. (The proof for the monotone case is derived from [HKL+14, Sec. 10.5] and Proposition 3.7. The proof for the convex case is derived from [Hal15] and Proposition 6.8.)

Our result also applies to maximization problems, where the DP recursion (3) has a “max” function instead of a “min” function. In order to achieve this, Condition 7 for maximization problems is reformulated as follows.

**Condition 8.** At least one of the following properties holds:
(i) (Increasing DP) Function \( g_{t+1} \) is increasing. For \( t = 1, \ldots, T \), function \( f_t(I, x, d) \) is increasing in \( I \) and monotone in \( x \), and \( g_t(I, x, d) \) is monotone in \( x \). Either monotone FPTASs to calculate \( E_D g_t(I, \cdot, D) \) are given for any fixed \( I \), or Condition 4 is met. \( f_t(I, x, D) \) is either linear and separable in its variables, or \( \text{Prob}(f_t(I, x, D) \geq a) \) is monotone in \( x \) for every fixed \( a, I \) and can be expressed as a linear and separable function of \( F_t \) multiplied by an indicator on the relation between \( I, x, a \). Moreover, for each \( t = 1, \ldots, T \), either \( z_t(\cdot) \) is increasing, or \( g_t(I, x, d) \) is increasing in \( I \) and \( A_t(I) \subseteq A_t(I') \) for all \( I, I' \in \mathcal{S}_t \) with \( I \leq I' \).

(ii) (Decreasing DP) Function \( g_{t+1} \) is decreasing. For \( t = 1, \ldots, T \), function \( f_t(I, x, d) \) is increasing in \( I \) and monotone in \( x \), and \( g_t(I, x, d) \) is monotone in \( x \). Either monotone FPTASs to calculate \( E_D g_t(I, \cdot, D) \) are given for any fixed \( I \), or Condition 4 is met. \( f_t(I, x, D) \) is either linear and separable in its variables, or \( \text{Prob}(f_t(I, x, D) \geq a) \) is monotone in \( x \) for every fixed \( a, I \) and can be expressed as a linear and separable function of \( F_t \) multiplied by an indicator on the relation between \( I, x, a \). Moreover, for each \( t = 1, \ldots, T \), either \( z_t(\cdot) \) is decreasing, or \( g_t(I, x, d) \) is decreasing in \( I \) and \( A_t(I) \subseteq A_t(I') \) for all \( I, I' \in \mathcal{S}_t \) with \( I \geq I' \).

(iii) (Concave DP) The terminal state space \( S_{T+1} \) is a contiguous interval. For \( t = 1, \ldots, T \), the state space \( S_t \) and the action space \( A_t(I) \), \( \forall I \in S_t \), are both contiguous intervals, and \( D_t \subset \mathbb{Z} \). Moreover, function \( f_t \) can be expressed as \( f_t(I, x, d) = aI + bx + cd \), where \( a(d) \in \mathbb{Z} \) and \( b, c \in \{-1, 0, 1\} \). Function \( g_{t+1}(\cdot) \) is concave over \( S_{T+1} \). For \( t = 1, \ldots, T \), the set \( S_t \cap A_t \) is integrally concave. Function \( g_t \) can be expressed as \( g_t(I, x, d) = g_t^l(f_t^l(I, d)) + g_t^z(f_t^z(x, d)) + g_t^n(f_t(I, x, d)) \), where \( g_t^l(\cdot), g_t^z(\cdot), g_t^n(\cdot) \) are univariate concave functions and \( f_t^l, f_t^z \) are linear and separable functions in their variables.

**Theorem 6.3.** Every implicit stochastic maximization DP satisfying conditions 5, 6 and 8 admits an FPTAS.

In the analysis presented in this and next sections we focus on minimization problems. A discussion of the validity of Theorem 6.3 is provided in [HKL+14, Sec. 10.1]

### 6.2 FPTAS for monotone DP

We start this section by showing how to construct succinct representations for the various functions that are computed throughout the execution of the FPTAS (Proposition 6.5), and how to build an oracle for the cost-to-go functions (Proposition 6.6). Considering Propositions 3.5 and 4.5, sometimes, as happens with \( z_t(\cdot) \) in Proposition 6.1 above and Proposition 6.6 below, direct access to the monotone function \( \tilde{\varphi} \) we want to approximate is not possible, and only access to an unnecessarily monotone function \( \tilde{\varphi} \) that \( L \)-approximates \( \varphi \) is possible \((L > 1)\). Since \( \tilde{\varphi} \) is unnecessarily monotone, a \( K \)-approximation set of it is not well defined. However, for this purpose [HKL+14] design a simple procedure called \textsc{IndirectAPXSet} (see Appendix D for the algorithm statement) that for every given parameters \( \tilde{\varphi}, [A, \ldots, B], K \), returns a \( K \)-approximation set \( \tilde{W} \) for a monotone function \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) which \( \tilde{\varphi} \) \( L \)-approximates, such that the following proposition holds.

**Proposition 6.4** (Approximation of an increasing function via \( L \)-approximation oracle, based on Props. 4.5 and 4.7 in [HKL+14]). Let \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be an increasing function. Let \( \tilde{\varphi} \) be an \((\text{unnecessarily increasing})\) \( L \)-approximation function of \( \varphi \) \((L > 1)\). Let \( \tilde{W} \) be the output of function \textsc{IndirectAPXSet} for given parameters \( \tilde{\varphi}, D, \) and \( K > 1 \). Let \( \hat{\varphi} \) be the maximal nondecreasing function that is bounded from above by \( \tilde{\varphi} \) over \( \tilde{W} \). Then, \( \tilde{W} \) is a \( K \)-approximation set of \( \hat{\varphi} \) of cardinality \( O(\log K \frac{\max}{\min} \log (B - A)) \); \( \hat{W} \cup \tilde{W}^+ \) is a 1-approximation set of \( \hat{\varphi} \); \( \hat{\varphi} \) is an increasing step \( KL \)-approximation function of \( \varphi \) that consists of \( O(|\tilde{W}|) \) steps; function \textsc{IndirectAPXSet} computes \( \tilde{W} \) in \( O(t_\varphi \log K \frac{\max}{\min} \log (B - A)) \) time; and the value \( \hat{\varphi}(x) \)
can be determined in \( O(\log \log_K \varphi_{\max}^{\varphi_{\min}}) \) time for any \( x \in [A, \ldots, B] \) if the breakpoints of \( \varphi \) are stored in a sorted array of \( \{(x, \tilde{\varphi}(x)) \mid x \in W\} \).

We define a function \textsc{IndirectCompressInc}, see Algorithm 4, for the efficient construction of a succinct representation of an approximation of an increasing function \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) which is accessed via an unnecessarily monotone \( L \)-approximation \( \tilde{\varphi} \). Applying Proposition 6.4 coupled with approximation of (Proposition 3.2(7)) yields the following result.

**Proposition 6.5.** Let \( K > 1 \) and \( L \geq 1 \) be real numbers and let \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be an increasing function over \([A, \ldots, B]\). Let \( \tilde{\varphi} \) be an (unnecessarily monotone) \( L \)-approximation function of \( \varphi \). Then function \textsc{IndirectCompressInc}(\( \tilde{\varphi}, [A, \ldots, B], K \)) returns in \( O(t_{\varphi} \log_K \varphi_{\max}^{\varphi_{\min}} \log(B - A)) \) time a representation of an increasing step function \( \tilde{\varphi} \) with \( O(\log \log K \varphi_{\max}^{\varphi_{\min}}) \) steps that \( KL \)-approximates \( \varphi \), has query time \( O(\log \log K \varphi_{\max}^{\varphi_{\min}}) \), and for which \( \tilde{W} \cup \tilde{W}^+ \) is a \( L \)-approximation set.

We now show how to build an oracle for the cost-to-go functions. In the case of implicit stochastic monotone DP, we build approximate value oracles for the cost-to-go functions via the following proposition, instead of Proposition 6.1 which is used in explicit stochastic monotone DP. (In order to calculate the approximate expectations \( G_t^l(\cdot), G_t^u(\cdot), G_t^u(I_t, \cdot) \) and \( \tilde{Z}_{f+1}(I_t, \cdot) \) we will soon use Proposition 3.7.)

**Proposition 6.6.** Suppose the DP formulation (3) satisfies Condition 4. Let \( K^{l_x}, K^u, L^x, L^l, L^z, L^u, L^z, t \), and \( I_t \) be fixed values, where \( K^x, L^z \geq 1 \), \( L^x, L^l, K^x L^z, K^u L^u \leq K^z L^z \). Let \( g_t \) be as stated in Condition 4. Let \( \tilde{G}^l_t(I_t) \) be an \( L^l \)-approximation value of \( G^l_t(I_t) := E_D(g^l_t(f^l_t(I_t, D_t))) \). Let \( \tilde{g}^l_t(I_t, \cdot) \) be a monotone \( L^l \)-approximation of \( g^l_t(I_t, \cdot) \), and let \( W^l_t \) be a \( K^l \)-approximation set of \( \tilde{g}^l_t(I_t, \cdot) \). Let \( \tilde{G}^z_t(I_t, \cdot) \) be a monotone \( L^z \)-approximation of \( G^z_t(\cdot) := E_D(g^z_t(f^z_t(\cdot, D_t))) \), and let \( W^z \) be a \( K^z \)-approximation set of \( \tilde{G}^z_t(\cdot) \). Let \( \tilde{G}^u_t(I_t, \cdot) \) be a monotone \( L^u \)-approximation of \( G^u_t(I_t, \cdot) := E_D(g^u_t(f^u_t(I_t, \cdot, D_t))) \), and let \( W^u \) be a \( K^u \)-approximation set of \( \tilde{G}^u_t(I_t, \cdot) \). Let \( \tilde{Z}_{f+1}(I_t, \cdot) \) be a monotone \( L^z \)-approximation of \( Z_{f+1}(I_t, \cdot) := E_{D_t}(z_{t+1}(f(I_t, \cdot, D_t))) \), and \( W^z \) be a \( K^z \)-approximation set of \( \tilde{Z}_{f+1}(I_t, \cdot) \). Let

\[
\tilde{z}_t(I_t) = \tilde{G}_t^l(I_t) + \min_{x_t \in W^l_t \cup W^u \cup W^z} \{ \tilde{g}_t^l(I_t, x_t) + \tilde{G}_t^z(x_t) + \tilde{G}_t^u(I_t, x_t) + \tilde{Z}_{f+1}(I_t, x_t) \}. \tag{7}
\]

Then, \( \tilde{z}_t(I_t) \) is an unnecessarily monotone \( K^z L^z \)-approximation value of \( z_t(I_t) \), and it can be determined in \( O([t_{\tilde{g}_t^l} + t_{\tilde{G}_t^z} + t_{\tilde{G}_t^u} + t_{\tilde{Z}_{f+1}}] W^l_t X W^u \cup W^u \cup W^z + t_{\tilde{G}_t^l}) \) time if the elements of \( W^l_t, W^u, W^z \) are given.

**Proof.** If all the 4 functions \( g^l_t, \tilde{G}_t^l, \tilde{G}_t^u, \tilde{Z}_{f+1} \) are monotone in \( x_t \) in the same direction, solving (7) is trivial – the argmin is either \( \min A_t(I) \) or \( \max A_t(I) \). Otherwise, let \( \Phi \) be the set consisting of the functions that are monotone in \( x_t \) in the same direction as \( \tilde{Z}_{f+1} \) is, and let \( m' := \#(\Phi) \) (so \( 1 \leq m' \leq 3 \)). We apply Proposition 3.6 with parameters set to \( n = 4, m = m', \psi_i(x, y) = y, i = 1, \ldots, 4, \varphi_1 = \tilde{Z}_{f+1}(I_t, \cdot), I_1 = L^z, K_1 = K^z, W_1 = W^z \). For \( i = 2, \ldots, m', \) let \( \varphi_i \) be the functions in \( \Phi \setminus \{\tilde{Z}_{f+1}\} \) and \( L_i, K_i \) and \( W_i \) be their corresponding values. For \( i > m' \) let \( \varphi_i \) be the remaining functions and let \( L_i \) and \( K_i \) be their corresponding values. \( \square \)
Procedure FPTASIncDP($\epsilon$)

1. $K \leftarrow \frac{3\epsilon^2}{\sqrt{T} + \epsilon}$, $K' \leftarrow K$, $z_{T+1} \leftarrow$ COMPRESSINC($g_{T+1}, S_{T+1}, K$)

2. For $t := T$ downto 1 do

3. $\tilde{F}_t \leftarrow$ COMPRESSINC($F_{t-1}, B_t, K$)

4. For $x \in \{I, x, u\}$: $\tilde{g}_t \leftarrow$ COMPRESSMON($g_t^0, S_t, K'$)

5. $\tilde{g}_t^I(I, \cdot) :=$ COMPRESSMON($g_t^I(I, \cdot), S_t, K'K^2$)

6. For $x \in \{I, x, u\}$: $\tilde{g}_t^x \leftarrow$ COMPRESSMON($g_t^x(I, \cdot), S_t, K'$)

7. Let $G_t^I(I, \cdot)$ be a monotone $K'K$-approximation for $G_t^I(I, \cdot) := E_D_t(g_t^I(f^I(\cdot), D_t)))$

8. Let $G_t^x(I, \cdot)$ be a monotone $K'K$-approximation for $G_t^x(I, \cdot) := E_D_t(g_t^x(f^x(\cdot), D_t)))$

9. For $x \in \{I, x, u\}$: $\tilde{G}_t^x \leftarrow$ COMPRESSINC($\tilde{g}_t, S_t, K$)

10. $\bar{S}_t \leftarrow$ INDIRECTCOMPRESSINC($\bar{S}_t, K'$)

11. $\bar{z}_t \leftarrow$ INDIRECTCOMPRESSINC($\bar{z}_t, S_t, K$)

12. $\bar{z}_{t+1} \leftarrow$ INDIRECTCOMPRESSINC($\bar{z}_{t+1}, S_{t+1}, K$)

13. $z_{t+1} \leftarrow$ INDIRECTCOMPRESSINC($z_{t+1}, S_{t+1}, K$)

14. $\tilde{K}' \leftarrow K'K^3$

15. end for

Algorithm 5: An FPTAS for an increasing DP.

6.2.1 FPTAS design and analysis

Algorithm 5 is our FPTAS for increasing DP that satisfies Condition 4. (The FPTAS for decreasing DP is similar.) The core of the algorithm is step 13. In order to evaluate an oracle call to $\bar{z}_t$ as defined in (7) for any fixed $I_t$, one needs to have access to the value of $\tilde{G}_t^I(I_t)$ as well as to oracles of $\tilde{g}_t^I(I_t, \cdot), \tilde{G}_t^I(I_t, \cdot), \tilde{Z}_t^I(I_t, \cdot)$ as defined in Proposition 6.6, as well as to their approximation sets.

These are derived via the calls to COMPRESSINC and COMPRESSMON in steps 6 and 10-12. While COMPRESSMON in step 6 has direct access to $g_t^I$ because $g_t^I$ is a priori given in the problem input, the remaining oracles $\tilde{G}_t^I(\cdot), \tilde{G}_t^I(\cdot), \tilde{G}_t^I(I_t, \cdot), \tilde{Z}_t^I(I_t, \cdot)$ are not, and are therefore constructed in steps 7-9 using Proposition 3.7. In each iteration of the algorithm the multiplicative error of $\bar{z}$ is multiplied by $K^3$ - one factor of $K$ is added in steps 7-9, one in steps 10-12, and one in step 13.

Remark. We note that when the symbol $\leftarrow$ is used, we mean that the action is explicitly made. But in steps 6, 8-9 and 11-12, which are only declarational, the oracles are explicitly built only for the values of $I$ that are queried by INDIRECTCOMPRESSINC in step 13. Of course, there are cases when the computation can be done more efficiently. For example, when the transition function is linear and separable in its variables, e.g. $f^u(I, x, D) = I + x + D$, then it suffices to explicitly compute an oracle only for $G^u(0, \cdot)$ and then the value of $G^u(I, x)$ is $G^u(0, I + x)$ for any $I$.

In Appendix E we describe how Algorithm 5 can be simplified whenever FPTASs to calculate $E_D g_t(I, x, D)$ for any fixed $I, x$ are a-priori given.

Proof of monotone case of Theorem 1.2. We start with bounding the approximation ratios of the various $\tilde{z}_t's$ computed throughout the algorithm by backward induction on the time period $t$. The base case is that $\tilde{z}_{T+1}$ is a $K$-approximation of $z_{T+1} \equiv g_{T+1}$ that is given explicitly together with a 1-approximation set of it, see step 2 of the algorithm coupled with Proposition 4.5. The induction hypothesis is that $\tilde{z}_{t+1}$ is a $K^{3(T-1)-t+1}$-approximation of $z_{t+1}$ that is given explicitly together with a 1-approximation set of it. Note that by its initial value in step 2, and its update rule in step 14, the value of $K'$ in lines 4-13 of iteration $t$ is $K^{3(T-1)-t+1}$. Consider now iteration $t$. From the previous iteration we already have an explicit representation of $\tilde{z}_{t+1}$ which is a $K'$-approximation of $z_{t+1}$, together with a 1-approximation set of it. We need to show that in step 13 we get a $K'K^3$-approximation of $z_t$, together with a 1-approximation set of it. We compute the oracles in steps 7-8 by calling Proposition 3.7 with parameters set to $K_1 = K'$ and $K_2 = K$. Considering step 9, by calling Proposition 3.7 with parameters set to $\xi = \tilde{z}_{t+1}, K_1 = 1$ and $K_2 = K$ we get an increasing $K$-approximation oracle for $E_{D_t}(\tilde{z}_{t+1}, f(I, \cdot, D_t))$. Note that by linearity, summation and composition of approximation (Proposition 3.2(2,3,4)), coupled with the induction hypothesis, we get...
that $E_{D_t}(\tilde{z}_{t+1}(f(I, \cdot, D_t)))$ is a $K'$-approximation of $E_{D_t}(z_{t+1}(f(I, \cdot, D_t)))$. Therefore, by approximation of approximation (Proposition 3.2(7)), we get that the oracle computed in this step is in fact a $K'K'$ increasing oracle for $Z_{f_{t+1}}(I, \cdot)$. By using Proposition 4.5 with parameters set to $L = K'K$ and $K = K$ we get in steps 10-12 monotone $K'K^2$-approximations functions together with their 1-approximation sets. We next apply Proposition 6.6 with parameters set to $K^{I_x} = K^x = K^u = K^z = 1$ and $L^{I_x} = L^x = L^u = L^z = K'K^2$ to get an unnecessarily monotone $K'K^2$-approximation oracle to $z_t$. Last, using Proposition 6.5 with parameters set to $L = K'K^2$ and $K = K$, we get in step 13 an increasing $K'K^3$-approximation function of $z_t$ together with its 1-approximation set, as needed.

We next analyze the running time of the algorithm. Let $M$ be a bound on $\frac{\max}{\min}$ and recall from Section 6.1 that $\gamma$ is a bound on the tail probabilities of the random variables. There are $T$ iterations. Within an iteration, the only steps which are operational (as opposed to declarational) are step 4 – which runs in $O(t_F \log_K (1/\gamma) \log U_D)$ time; step 5 – which runs in $O(t_g \log_K M (\log U_A + \log U_S))$ time; step 10 and step 13. Considering step 10, by Proposition 4.5, its running time is $O((t_{G_f} \log U_S + t_{G_f} \log U_A) \log K \frac{M}{\gamma})$.

where due to Proposition 3.7, $t_{G_f} = O((\log \log_K \frac{1}{\gamma} + t_{g_f}) \log_K M)$ and $t_{G_f} = O((\log \log_K \frac{1}{\gamma} + t_{g_f}) \log_K M)$.

We next consider step 13. By Proposition 6.5 it runs in time

$$O(\log_K M \log U_S t_{\tilde{z}}).$$

Due to Proposition 6.6, calculating the value of $\tilde{z}(I)$ as defined in (7) takes

$$t_{\tilde{z}} = O(\log_K M (t_{\tilde{g}_f} + t_{G_f} + t_{G_u} + t_{\tilde{z}_f}) + \text{construction time of apx. sets and oracles for } \tilde{g}^{I_x}, \tilde{G}^x, \tilde{G}^u, \tilde{Z} f)$$

(9)

Clearly, among the various functions, the running times of calculating approximation sets and oracles for $\tilde{g}^{I_x}, \tilde{G}^x$ are dominated by the ones for $\tilde{G}^u, \tilde{Z} f$. The construction of approximation sets and oracles for $\tilde{G}^u(I, \cdot), \tilde{Z} f(I, \cdot)$ for a fixed value of $I$ is done in steps 11-12. Due to Proposition 4.5 this is done in time

$$O(\log_K M \log U_S (t_{G_f} + t_{\tilde{z}_f})).$$

(10)

Once such a construction is made we have query time

$$t_{G_u} = t_{\tilde{z}_f} = O(\log \log_K M).$$

(11)

Therefore, using (10)-(11), (9) is equivalent to

$$t_{\tilde{z}} = O(\log_K M (\log \log_K M + \log U_S (t_{G_f} + t_{\tilde{z}_f}))) .$$

(12)

We now turn to steps 8-9 which construct the oracles for $\tilde{G}^u, \tilde{Z} f$. By Proposition 3.7 (see also Corollaries 3.8-3.9) we have that

$$t_{G_u} = t_{\tilde{z}_f} = O(\log_K M \log \log_K (1/\gamma)).$$

(13)

(Note that in Proposition 3.7 $n = O(\log_K M)$ and $t_\theta = O(\log \log_K (1/\gamma))$.) Combining (8) with (12)-(13), coupled with the equation $\log_K M = O(T^2 \log M)$ which holds for $0 < \epsilon < 1$, we get that the total running time is

$$O\left(\frac{T^2}{\epsilon} (t_F \log U_D \log 1/\gamma + (t_g + \log \frac{T}{\epsilon} \log 1/\gamma) T \log \frac{\max}{\min} (\log (U_A U_S)) + \frac{T^4}{\epsilon^3} \log^2 \frac{\max}{\min} \log U_S \log \frac{T}{\epsilon} \log 1/\gamma) \right).$$

Remark. In many cases, e.g., when the transition function and the various “$f$’s” functions are linear and separable in their variables, see also the remark in the end of Section 6.2.1, in each iteration of the algorithm it suffices to build only once oracle and approximations sets for the various functions. The running time of the algorithm then reduces to

$$O\left(\frac{T^2}{\epsilon} (t_F \log U_D \log 1/\gamma + (t_g + \log \frac{T}{\epsilon} \log 1/\gamma) T \log \frac{\max}{\min} (\log (U_A U_S)) + \frac{T^3}{\epsilon^2} \log^2 \frac{\max}{\min} \log U_S \log \frac{T}{\epsilon} \log 1/\gamma) \right).$$
Let \( E \) approximation functions, \( D \) belong to \( S \) computed in \( O(\log K) \) time for each \( \xi \) breakpoints \( A \). Then Function CompressConv(\( \phi \), \( [A, \ldots, B] \), \( K \)) returns in \( O((1 + t_\phi)(\log K \frac{\epsilon_{\max}}{\epsilon_{\min}} \log (B - A))) \) time a \( O(\log K \frac{\epsilon_{\max}}{\epsilon_{\min}}) \) piecewise linear convex KL-approximation of \( \phi \).

### 6.3 FPTAS for convex DP

We start this section by showing how to construct succinct representations for the various functions that are computed throughout the execution of the FPTAS. Algorithm 6 efficiently constructs a succinct representation of a convex function \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}_+ \). Applying Propositions 3.5 and 3.2(7) we get

**Proposition 6.8.** Let \( K, L \geq 1 \) be real numbers, and let \( \varphi : [A, \ldots, B] \rightarrow \mathbb{R}_+ \) be a convex function. Let \( \hat{\varphi} \) be an increasing piecewise linear convex function with breakpoints \( a_1 < \ldots < a_n < B \) and slopes \( 0 = \Delta_0 \leq \Delta_1 < \ldots < \Delta_n \) that \( K \)-approximates \( \varphi \). Let \( D \) be a (not necessarily nonnegative) discrete random variable with support \( \mathcal{D} = \{d_1 < \ldots < d_m\} \). Let \( S = \{d_1 = d_{s(1)} < \ldots < d_{s(k)} = d_m\} \) be a \( K \)-approximation set of the CDF \( F(\cdot) \) of \( D \). Let \( S^- = \{d_{s(i)} \leq b_1 \leq \ldots \leq d_{s(k)} \mid s = 2, \ldots, k \} \). Let \( f(x, d) = bx + e - d \) for some given \( b, e \). Let \( m_i = \arg \max_{j=1,\ldots,k} \{d_{s(j)} \mid d_{s(j)} \leq bx + e - a_i\} \). Then

\[
\hat{\xi}(x) = \psi(A) + \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \left( (bx + e - d_{s(m_i)} - a_i)F(d_{s(m_i+1)} - 1) + \sum_{j=1}^{m_i-1} (d_{s(j+1)} - d_{s(j)})F(d_{s(j+1)} - 1) \right)
\]

is a convex piecewise linear \( K \)-approximation of \( E_D(\xi(f(x, D))) \) with \( O(nk) \) pieces that can be computed in \( O(n + kt) \) time for each \( x \in [A, \ldots, B] \). Moreover, all queries to \( F(\cdot) \) are made on points that belong to \( S^- \).

The proof of the above proposition relies on the following two facts. First, by the calculus of \( K \)-approximation functions, \( E_D(\psi(f(x, D))) = \sum_{j=1}^{m} \psi(f(x, d))\text{Prob}(D = d_j) \) is a \( K \)-approximation of \( E_D(\xi(f(x, D))) \). Second, it exploits a representation of \( \psi(\cdot) \) as \( \psi(x) = \psi(A) + \sum_{i=1}^{n} \psi_i(x) \), where \( \psi_i(x) = (\Delta_i - \Delta_{i-1})(x - a_i) \) if \( x \geq a_i \) and \( \psi_i(x) = 0 \) if \( x < a_i \).

**Proof.** Let \( M_i = \arg \max_{j=1,\ldots,m} \{d_j \mid d_j \leq bx + e - a_i\} \) and \( \chi(x) = E_D(\psi(f(x, D))) \) and \( A_i = bx + e - a_i \).
and notice that:

$$\chi(x) = \sum_{j=1}^{m} \psi(f(x, d_j)) \text{Prob}(D = d_j)$$

$$= \psi(A) + \sum_{j=1}^{m} \sum_{i=1}^{n} \psi_i(f(x, d_j)) \text{Prob}(D = d_j)$$

$$= \psi(A) + \sum_{j=1}^{m} \left( \sum_{i=1}^{n} \psi_i(f(x, d_j)) \text{Prob}(D = d_j) \right)$$

$$= \psi(A) + \sum_{j=1}^{m} (\Delta_i - \Delta_{i-1}) \sum_{j=1}^{n} f(x, d_j) \geq a_i (f(x, d_j) - a_i) \text{Prob}(D = d_j)$$

$$= \psi(A) + \sum_{j=1}^{m} (\Delta_i - \Delta_{i-1}) \sum_{j=1}^{n} [(x, d_j) \geq a_i (f(x, d_j) - a_i) \text{Prob}(D = d_j)]$$

$$= \psi(A) + \sum_{j=1}^{m} (\Delta_i - \Delta_{i-1}) A \left( \sum_{j=1}^{n} (D - d_j) F(d_j) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \right)$$

Note that

$$\frac{(A_i - d_{M_i}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) =}{(A_i - d_{M_i}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \leq}{(A_i - d_{M_i}) F(d_{s(m_i)+1}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \leq}{(A_i - d_{M_i}) F(d_{s(m_i)+1}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \leq}{(A_i - d_{s(m_i)}) F(d_{s(m_i)+1}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \leq}{(A_i - d_{s(m_i)}) F(d_{s(m_i)+1}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j),}$$

where the first inequality is due to the fact that $s(m_i)$ and $s(m_i + 1)$ are consecutive integers if and only if $d_{M_i} = d_{s(m_i)}$ which implies that $d_{s(m_i)+1} \geq d_{M_i}$. The next two inequalities are due to telescopic sums coupled with the monotonicity of $F(\cdot)$. Therefore, we have

$$E_D(\xi(f(x, D))) = \frac{\sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \geq}{\sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \leq}{\sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \leq}{\sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \leq}{\sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j),}$$

where the first inequality is due to $\psi$ being a $K_1$-approximation function of $\xi$, the second equality is due to (15) and the last inequality is due to (16). On the other hand, by using similar arguments we have

$$\frac{(A_i - d_{M_i}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \geq}{(A_i - d_{M_i}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \geq}{(A_i - d_{M_i}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \geq}{(A_i - d_{M_i}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \geq}{(A_i - d_{M_i}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j),}$$

where the third inequality is due to $S$ being a $K_2$-approximation set of $F(\cdot)$ coupled with the inequality $d_{s(m_i)} \leq d_{M_i} < d_{s(m_i+1)}$. Using (17) we therefore get

$$\chi(x) \geq \frac{1}{K_2} \left( \psi(A) + \sum_{i=1}^{n} (\Delta_i - \Delta_{i-1}) \left( (A_i - d_{s(m_i)}) F(d_{M_i}) + \sum_{j=1}^{M_i-1} (d_{j+1} - d_j) F(d_j) \right) \right)$$

$$= \frac{\xi}{K_2},$$
We conclude the proof by noting that one can compute where each slope increase belongs to the set mation ratio by combining the above two inequalities.

We also have:

\[ \tilde{\phi}, \tilde{\psi} \]

Then, \( \tilde{\phi} \), \( \tilde{\psi} \) is a decreasing piecewise linear convex function. We therefore get the following corollary.

**Corollary 6.9.** Let \( \xi : [A, \ldots, B] \rightarrow \mathbb{R}^+ \) be a (not necessarily monotone) convex function. Let \( K_1, K_2 \geq 1 \), and let \( C \) satisfy \( A \leq C \leq B \). Let \( \psi^- : [A, \ldots, C] \rightarrow \mathbb{R}^+ \) be a decreasing piecewise linear convex function with \( n^- \) breakpoints that \( K_1 \)-approximates \( \xi \) over \([A, \ldots, C]\). Let \( \psi^+ : [C, \ldots, B] \rightarrow \mathbb{R}^+ \) be an increasing piecewise linear convex function with \( n^+ \) breakpoints that \( K_1 \)-approximates \( \xi \) over \([C, \ldots, B]\). Let \( D \) be a (not necessarily nonnegative) discrete random variable. Let \( S = \{d_{s(1)} < \ldots < d_{s(k)}\} \) be a \( K_2 \)-approximation set of the CDF \( F(\cdot) \) of \( D \). Let \( S^- = \{d_{s(i)-1} \mid i = 2, \ldots, k\} \). Last, let \( f(\cdot, \cdot) \) be a function separable and linear in its variables. Applying Proposition 6.8 twice, we can compute a convex piecewise linear value oracle with \( O((n^- + n^+)k) \) pieces that is a \( K_1K_2 \)-approximation of \( E_D(\xi(f(x,D))) \). Each of its values can be computed in \( O(n^- + n^+ + kt_F) \) time where the queries to \( F(\cdot) \) are only on points that belong to \( S^- \).

### 6.3.2 A convex oracle for the cost-to-go functions

When dealing with convex functions it is possible to perform binary search. For this reason we will use the following simple proposition instead of Proposition 3.6.

**Proposition 6.10** (Minimization of summation of composition, Prop. 5.2 in [HKL+14]). Let \( n \in \mathbb{N} \), let \( K_i \geq 1 \) for \( i = 1, \ldots, n \), let \( D \) be any finite domain, and let \( C(x) \) be any finite domain for every \( x \in D \). Let \( \varphi : D \rightarrow \mathbb{R}^+ \), let \( \tilde{\varphi}_i \) be a \( K_i \)-approximation of \( \varphi_i \), and let \( \psi_i : D \otimes C \rightarrow D \) for \( i = 1, \ldots, n \). Let \( \varphi, \tilde{\varphi} : D \rightarrow \mathbb{R}^+ \) such that

\[
\varphi(x) = \min_{y \in C(x)} \left\{ \sum_{i=1}^{n} \varphi_i(x,y) \right\} \quad \text{and} \quad \tilde{\varphi}(x) = \min_{y \in C(x)} \left\{ \sum_{i=1}^{n} \tilde{\varphi}_i(x,y) \right\}.
\]

Then, \( \tilde{\varphi} \) is a max\{\( K_1, \ldots, K_n \)\}-approximation of \( \varphi \).

In the case of implicit stochastic DPs, the following proposition corresponds to Proposition 6.2. (In order to calculate the approximate expectations for the various \( G_t \)'s and for \( Zf_{t+1}(I_t, \cdot) \) we will soon use Proposition 6.8.)

---

25
1: Procedure FPTASConvexDP(ε)
2: $K \leftarrow \frac{2r}{\sqrt{1 + \epsilon}}, K' \leftarrow K$, $z_{T+1} \leftarrow \text{COMPRESSConv}(g_{T+1}, S_{T+1}, K)$
3: for $t := T$ downto 1 do
4:  $\tilde{F}_t \leftarrow \text{COMPRESSINC}(F_t, [A_1, \ldots, B_t], K)$
5:  For $\hat{\phi} \in \{I, x, u\}$: $\hat{g}_{\hat{\phi}} \leftarrow \text{COMPRESSConv}(g_{\hat{\phi}}, S_t, K')$
6:  For $\hat{\phi} \in \{I, x\}$: Let $\hat{G}_{\hat{\phi}}(\cdot)$ be a convex $K'K$-approximation of $G_{\hat{\phi}}(\cdot) := E_{D_t}(g_{\hat{\phi}}(f(\cdot, D_t)))$
7:  Let $\hat{G}_{\hat{\phi}}(\cdot)$ be a convex $K'K$-approximation of $G_{\hat{\phi}}(\cdot) := E_{D_t}(g_{\hat{\phi}}(\cdot + cD_t))$
8:  Let $Z_{f_t+1}(\cdot)$ be a convex $K'K$-approximation of $Z_{f_t+1}(\cdot) := E_{D_t}(z_{t+1}(\cdot + cD_t))$
9:  $\hat{z}_t \leftarrow \text{COMPRESSConv}(z_t, S_t, K)$ if $\hat{z}_t$ is as defined in (18) /*
10: $K' \leftarrow K'K^2$
11: end for

Algorithm 7: An FPTAS for a convex DP.

**Proposition 6.11.** Suppose the DP formulation (3) satisfies Condition 7(iii). Let $K^I, K^x, K^u, K^z$, $t$, and $I_t$ be fixed values, where $K^z \geq K^I, K^x, K^u \geq 1$, $I_t \in S_t$, and $t \in \{1, \ldots, T\}$. Let $\tilde{G}_t(\cdot)$ be a convex $K^I$-approximation of $G_t(\cdot) := E_{D_t}(g_t(f(\cdot, D_t)))$. Let $\hat{G}_t(\cdot)$ be a convex $K^x$-approximation of $G_t(\cdot) := E_{D_t}(g_t(\cdot + cD_t))$. Let $\tilde{z}_t(I_t)$ be a convex $K^z$-approximation of $z_t(I_t)$. Moreover, each value $\tilde{z}_t(I_t)$ can be determined in $O(t\tilde{G}_t + t\hat{G}_t + t\tilde{z}_{f_t+1} + \log |A_t(I_t)|)$ time.

**Proof.** We apply Proposition 6.10 with the following parameter setting: Let $D = S_t, C(\cdot) = A_t(\cdot), n = 3, x = I_t$ and $y = x_t$. Let $\varphi_1(\cdot) = G_t(\cdot), \tilde{\varphi}_1(\cdot) = \tilde{G}_t(\cdot)$ and $\psi_1(\cdot, x) = x$. Let $\varphi_2(\cdot) = G_t(\cdot), \tilde{\varphi}_2(\cdot) = \tilde{G}_t(\cdot)$ and $\psi_2(\cdot, x) = aI + bx$. Let $\varphi_3(\cdot) = Z_{f_t+1}(\cdot), \tilde{\varphi}_3(\cdot) = \tilde{Z}_{f_t+1}(\cdot)$ and $\psi_3(\cdot, x) = aI + bx$. Let $K_1 = K^x, K_2 = K^u, K_3 = K^z$. Hence, by Proposition 6.10, $\hat{\varphi}$ is a max-$\{K^x, K^u, K^z\}$-approximation of $\sum_{i=1}^3 \varphi_i$. By the property “summation of approximation” of the calculus of approximation (Proposition 3.2(3)) we conclude that $\tilde{G}_t + \hat{\varphi}$ is a $K^z$-approximation of $z_t$.

The convexity of $\tilde{z}_t(I_t)$ is due to the same arguments used in [HKL+14, Prop. 9.1], as well as the fact that both $f^I(I, d)$ and $f^x(x, d)$ are separable and linear in their variables.

As for the computational time, note that a summation of convex functions is a convex function, and therefore its minimum can be obtained in $O(\log |A_t(I_t)|)$ steps by performing binary search over the contiguous interval $A_t(I_t)$. Each of these steps involves queries to $\tilde{G}_t^x, \hat{G}_t^u, \tilde{Z}_{f_t+1}$ and requires $O(t\tilde{G}_t + t\hat{G}_t + t\tilde{z}_{f_t+1})$ time.

### 6.3.3 FPTAS design and analysis

Algorithm 7 is our FPTAS for convex DP. The core of the algorithm is step 9. In order to evaluate an oracle call to $\tilde{z}_t$ as defined in (18), one needs to have access to oracles of $\tilde{G}_t(\cdot), \hat{G}_t(\cdot), \tilde{z}_{f_t+1}(\cdot), Z_{f_t+1}(\cdot)$ as defined in Proposition 6.11. These are computed in steps 6-8 via Corollary 6.9. In each iteration of the algorithm the multiplicative error of $\hat{z}$ is multiplied by $K^2$ – one factor of $K$ is added in steps 6-8 and the other factor of $K$ is added in step 9.

**Remark.** We note that our FPTAS for convex DP is simpler than the one for monotone DP because while in the convex case we calculate $\tilde{z}_t$ in (18) via binary search, in the monotone case we calculate $\tilde{z}_t$ in (7) via exhaustive search on the various approximation sets. We construct these approximation sets in steps 10-12 of Algorithm 5 which results in an additional loss of factor $K$ in the approximation ratio within an iteration.
Proof of convex case of Theorem 1.2. We start with bounding the approximation ratios of the various \( \tilde{z}_t \)'s computed throughout the algorithm by backward induction on the time period \( t \). The base case is that \( \tilde{z}_{T+1} \) is a convex \( K \)-approximation of \( z_{T+1} \equiv g_{T+1} \) that is constructed explicitly, see Proposition 6.7. The induction hypothesis is that \( \tilde{z}_{t+1} \) is a convex \( K^{2(T+1-t)+1} \)-approximation of \( z_{t+1} \) that is given explicitly. Note that by its initial value in step 2, and its update rule in step 10, the value of \( K' \) in lines 4-9 of iteration \( t \) is \( K^{2(T-t)+1} \). Consider now iteration \( t \). From the previous iteration we already have an explicit representation of \( \tilde{z}_{t+1} \) which is a \( K' \)-approximation of \( z_{t+1} \). We need to show that in step 9 we get a convex \( K'K^2 \)-approximation of \( z_t \). We compute the oracles in steps 6-8 by calling Corollary 6.9 with parameters set to \( K_1 = K' \) and \( K_2 = K \). We next apply Proposition 6.11 with parameters set to \( K = K' = K^x = K^u = K^z = K'K \) to get a convex \( K'K \)-approximation oracle to \( z_t \). Last, using Proposition 6.7 with parameters set to \( L = K'K \) and \( K = K \), we get in step 9 an explicit representation of a convex \( K'K^2 \)-approximation function of \( z_t \), as needed.

We next analyze the running time of the algorithm. Let \( M \) be a bound on \( \tilde{z}_{\max} \) and recall from Section 6.1 that \( \gamma \) is a bound on the tail probabilities of the random variables. There are \( T \) iterations. Within an iteration, among the most time consuming steps are step 4 – which runs in time \( O(tF_\log_K(1/\gamma) \log U_D) \), step 5 – which runs in time \( O(tg_\log_K M (\log U_A + \log U_S)) \), and step 9 – which by Proposition 6.7 runs in time

\[
O(log_K M \log U_S t\tilde{z}).
\]

Due to Proposition 6.11, calculating the value of \( \tilde{Z}(I) \) as defined in (18) takes

\[
t\tilde{z} = O(tG_I + \log U_A(tG_u + tG_v + t\tilde{Z}_f))
\]
time, once having oracles for \( \tilde{Z}_f \), \( G^\phi \), \( \phi \in \{I, x, u\} \) at hand. These oracles are given explicitly in (14) and have query time

\[
tG^\phi = t\tilde{Z}_f = O(\log_K M + \log_K(1/\gamma)),
\]

once the breakpoints of \( \tilde{g}^\phi, \tilde{z} \) are given. While the breakpoints of \( \tilde{z} \) are already computed in step 9 of the previous iteration, the ones of \( \tilde{g}^\phi \) are computed in step 5 of the current iteration in \( O(tg_\log_K M (\log U_S + \log U_A)) \) time. Overall, we get that

\[
t\tilde{z} = O(\log U_A(\log_K M + \log_K(1/\gamma))).
\]

By using the equation \( \log_K M = O(T^2 \epsilon / \log M) \) which holds for \( 0 < \epsilon < 1 \) we get that the total running time is

\[
O\left(\frac{T^2}{\epsilon} \left[ tF_\log(1/\gamma) \log U_D + tg_\log \frac{\varphi_{\max}}{\varphi_{\min}} \log(U_A U_S) \right] + \frac{T^3}{\epsilon^2} \log U_S \log U_A \log \frac{\varphi_{\max}}{\varphi_{\min}} (\log \frac{\varphi_{\max}}{\varphi_{\min}} + \log(1/\gamma)) \right).
\]

7 Approximation schemes for sample-based dynamic programs

In this section we relax the requirement in Condition 6 that there is oracle access to the CDFs of the random variables. Instead, we require:

Condition 9. For every \( t = 1, \ldots, T+1 \), functions \( f_t, g_t \) are either given explicitly (i.e., as explicit formulae) or accessed via value oracles. Independent samples of the random variables \( D_t \) are available. Moreover, the values of \( g_t \) are polynomially bounded by the (binary) size of the input.
The input data of sample-based DP consists of the number of time periods $T$, the initial state $I_1$, the endpoints $A_t, B_t$ of the supports of $D_t$, and lower bounds $\gamma_t$ on the CDFs of $D_t$ as defined in the Introduction. We are now ready to give a proof for Theorem 1.3.

**Proof.** We “plug-in” density estimation results into the implicit stochastic DP FPTAS framework stated in Section 6. I.e., we reduce the sample-based DP to an instance of the implicit stochastic DP. In order to implement step 4 in Algorithm 5 and step 4 in Algorithm 7 we first construct functions $\tilde{F}_t (\tilde{F}_t^c)$ that with probability $1 - \delta/T$ are $K$-approximations of the CDF $F_t (1 - F_t), t = 1, \ldots, T,$ respectively. Due to union bound, with probability at least $1 - \delta$, all these functions are $K$-approximations, thus calling the FPTAS of Theorem 1.2 will result in an $(1 + \epsilon)$-approximation.

**Remark.** [LRS07, CSL15] study a certain multi-period inventory control problem and provide for it approximation schemes. They make no assumptions on the demand distributions but assume linear single-period cost functions. Their approximation schemes require at least $\Omega(T^4)$ samples in each time period. This problem, when assuming that the demand distributions have finite supports and bounded CDFs, satisfies Conditions 5, 7(iii) and 9. Therefore, our approximation scheme applies to it (and even if the single-period cost functions are convex and not necessarily linear), and it requires only $N = O(T^2)$ samples in each time period.

### 8 Concluding remarks

In this paper we prove that SNV cannot be approximated to within any constant multiplicative factor in polynomial time. We also provide approximation schemes for SNNV in the case that the demand distribution has bounded support and there is a bound on the CDF. We give the first FPTAS framework for implicit stochastic DPs and the first approximation scheme framework for sample-based DPs.

[LRS07] give approximation schemes for single period and multi-period SNVs, assuming nothing on the demand distribution and exploiting the linearity of the cost functions, where the only non-polynomial running time dependency of their algorithm is on $b + h/\min(b, h)$. [LPU15] give a faster approximation scheme for single period SNV under the assumption that the demand distribution is log-concave. [CSL15] give a faster approximation scheme (w.r.t. $T$) for $T$-period SNV, see Table 1. We give approximation schemes for single period SNNV under the assumption that the demand distribution has bounded support and there is a bound $\gamma$ on the CDF of the demand. In this case we provide approximation schemes for SNNV, where the only non-polynomial running time dependency is on $1/\gamma$. Under the same assumption about the existence of $\gamma$, the number of samples required by our algorithm for $T$-period SNNV is independent of the recursion level $t = 1, \ldots, T$, and its dependency on $T$ is only $O(T^2 \log T)$, i.e., by orders of magnitude smaller than the ones of [LRS07, CSL15] (though their algorithms work for general demand distributions but linear single-period cost functions). This raises the question whether the dependency on $T$ can be reduced below $O(T^3)$ in the models considered by [LRS07, CSL15].

It is interesting to explore other tradeoffs between cost functions structure and demand distribution structure. For instance, have a different assumption on the demand distribution (e.g., belongs to a special family), and assume a special structure on the monotone cost functions, in order to get approximation schemes with other running time and sample size complexities.

### References


Appendix

A More impossibility results

**Proposition A.1.** Given an instance of the newsvendor problem where the CDF of the demand is given as a value oracle, calculating the expected optimal cost, even if the ordering, holding and lost sales costs are all linear, requires exponential number of function evaluations.

The proof technique of this proposition is similar to that of [HOS12], who consider the profit maximization NNV problem. Our impossibility results rely on the following trivial observation:

**Observation A.2.** Finding a minimum of an arbitrary integer-valued function $f : [0, ..., N] \rightarrow \mathbb{Z}$, or even deciding whether such a minimum realizes in either $[1, ..., \lfloor N/2 \rfloor]$ or $[\lceil N/2 \rceil, ..., N]$, requires $N+1$ queries in the worst case.

Of course, if we have additional information about the function, the number of queries needed may be reduced. For example, if the function is monotone, only two queries are needed. If the function is convex, only $O(\log N)$ queries are needed. But if the function is unimodal with a unique minimum (e.g., a function which is zero everywhere except for one point), the number of queries needed is $N$. We are now ready for the proof of Proposition A.3

**Proof. (of Proposition A.1)** Let the per-unit cost be zero and per-unit holding cost be equal to 1. Let the per-unit shortage cost be very high, e.g., $N^2$. The support of the demand $D$ is $\{1, \ldots, N\}$ and its probability distribution function (PDF) is $P(D = i) = \frac{2^i}{N(N+1)}$ for all indices $i = 1, \ldots, N$ except for $i^* - 1, i^*$, for which $P(D = i^* - 1) = \frac{2(2i^*-1)}{N(N+1)}$ and $P(D = i^*) = 0$. So the probability distribution function of $D$ over $1, \ldots, N$ is minimized at $i^*$ (with value 0) and is positive otherwise. Moreover, the CDF is $F_D(i) = \frac{2 \sum_{j=1}^{i} j}{N(N+1)}$ for all $i \neq i^* - 1$. Note that finding $i^*$ via either the PDF or the CDF takes the same number of oracle calls. Note also that the instance input size is $O(\log N)$. In this case an optimal policy is to order $N$ units, and the resulting profit is:

$$E(D) = \frac{2}{N(N+1)} \left( \sum_{j=1}^{N} (N - i) i + i^* \right) = \frac{N - 1}{3} - \frac{2i^*}{N(N+1)}.$$ 

Therefore, computing the expected profit is equivalent to finding $i^*$, which by Observation A.2 requires $O(N)$ queries in the worst case.

The next proposition tells us that NNV is hard.

**Proposition A.3.** Given an instance of the nonlinear newsvendor problem, deciding whether there exists a cost less than a given number $N$, even if the holding and lost sales costs are linear and the demand is fixed, requires exponential number of function evaluations:

**Proof.** Let the fixed demand be $D = N$, the holding and lost sales costs be 1 per item, and the cost of each item purchased be 1 except items $i^*$ and $i^* + 1$. Item $i^*$ costs 0, and item $i^* + 1$ costs 2. The index $i^*$ (which minimizes the function $f(x) = \text{cost of item } x$) is unknown to the newsvendor and must be determined by evaluations of $f(\cdot)$. If $i^* \leq N$ and if one orders exactly $i^*$ items, then the total cost is $N - 1$. Otherwise the optimal cost is $N$ and is achieved for any order quantity between 0 and $N$. Observation A.2 tells us that deciding whether $i^* \leq N$ requires $\Omega(N)$ queries in the worst case.
B Formulation of problems in the implicit stochastic model

[HKL+14, Tab. 2] studied seven stochastic problems and formulated them as either monotone or convex explicit stochastic DPs, hence by using their FPTAS framework, all these problems admit an FPTAS [HKL+14, Appendix A]. In this section, we extend these problems into the implicit stochastic settings, point out what changes in the problem formulation are needed to make them fit either the monotone or convex implicit stochastic DP model, so as by Theorem 1.2 they all admit an FPTAS. In the following, we rely on the problem definitions, notations, and formulations as given in [HKL+14, Appendix A]. Therefore, this section should be read along with Appendix A of [HKL+14].

B.1 Stochastic ordered adaptive knapsack problem

[HKL+14, Appendix A.1] defined $\pi_t$ is a given constant:

$$A_t(I_t) = \{0, 1\}, \quad g_t(I_t, x_t, D_t) = x_t \pi_t \delta_{D_t \leq I_t}, \quad f_t(I_t, x_t, D_t) = (I_t - x_t D_t)^+$$

and formulated the problem as a maximization explicit stochastic DP. We note that $E g_t(I_t, x_t, D_t) = x_t \pi_t \Pr(D_t \leq I_t) = x_t \pi_t F_t(I_t)$. Hence, once the CDF of $D_t$ is (implicitly) given, we can calculate $E g_t(I_t, x_t, D_t)$ exactly. Regarding the transition function, because the action $x_t$ is binary, $\Pr(f_t(I_t, x, D) \geq a)$ is trivially monotone in $x$. We express it as $\Pr(f_t(I_t, x, D) \geq a) = \Pr((I_t - x_t D_t)^+ \geq a) = x F_t(I_t - a) + (1 - x) \delta_{I_t \geq a}$. Hence the problem is formulated as a maximization increasing implicit stochastic DP.

B.2 Single-item stochastic inventory control

[HKL+14, Appendix A.5] defined:

$$g_t(I_t, x_t, D_t) = c_t(x_t) + h_t(I_t + x_t - D_t), \quad f_t(I_t, x_t, D_t) = I_t + x_t - D_t$$

and formulated the problem as a convex explicit stochastic DP. By setting $g^+_t = 0, g^-_t(y) = c_t(y), f_t(x, d) = x$ and $g^{u}_t(y) = h_t(y)$ we get that the problem is a convex implicit DP.

B.3 Single-item stochastic batch dispatch

[HKL+14, Appendix A.6] defined $(K_t, c_t, h_t$ are all given constants):

$$g_t(I_t, x_t, D_t) = K_t \delta_{x_t > 0} + c_t x_t + h_{t-1} I_t, \quad f_t(I_t, x_t, D_t) = I_t - x_t + D_t$$

and formulated the problem as an increasing explicit stochastic DP. We note that $g_t(I_t, x_t, D_t)$ is deterministic and explicit, so also is $E g_t(I_t, x_t, D_t)$. $f_t(I_t, x_t, D_t)$ is linear and separable in its variables. Hence the problem is formulated as an increasing implicit stochastic DP.

B.4 Single-resource revenue management

[HKL+14, Appendix A.7] defined $(r_t$ is a given constant):

$$g_t(I_t, x_t, D_t) = r_t \min\{x_t, D_t\}, \quad f_t(I_t, x_t, D_t) = I_t + \min\{x_t, D_t\}$$

and formulated the problem as a maximization decreasing explicit stochastic DP. We set $g^+_t = 0, g^-_t(y) = r_t y, f_t^+(x, d) = \min\{x, d\}$ and $g^{u}_{-t}(u) = 0$. Note that $\Pr(f_t^+(x, D) \geq a) = \Pr(D \geq a) \delta_{x \geq a} = (1 - F(a - 1)) \delta_{x \geq a}$. Therefore the problem satisfies Condition 4. We also note that $\Pr(f_t(I_t, x, D) \geq a) = (1 - F(a - I - 1)) \delta_{x \geq a - I}$. Hence the problem is formulated as a maximization decreasing implicit stochastic DP.
B.5 Lifetime consumption of risky capital

[HKL+14, Appendix A.8] defined \((a_t)\) is a given constant, \(u_t(\cdot)\) is a given nonnegative monotone increasing function, and the support of \(D_t \geq -1\) consists of rational numbers with a given denominator \(q_t\):

\[
g_t(I_t, x_t, D_t) = u_t(x_t), \quad f_t(I_t, x_t, D_t) = (1 + D_t)(I_t - x_t) + y_t, \quad x_t \leq I_t
\]

and formulated the problem as a maximization increasing explicit stochastic DP. We note that \(g_t(I_t, x_t, D_t)\) is deterministic so \(\mathbb{E}g_t(I_t, x_t, D_t) = u_t(x_t)\). Note that because \(D \geq -1\) and \(x \leq I\) we get that \((1 + D)(I - x)\) is nonnegative so \(\text{Prob}(f_t(I, x, D) \geq a) = 1\) for every \(a \leq y_t\). If \(a > y_t\) then \(\text{Prob}(f_t(I, x, D) \geq a) = \text{Prob}(D \geq \frac{a - y_t}{1 - x} - 1) = 1 - F\left(\frac{a - y_t}{1 - x} - 1 - q_t\right)\), which is monotone increasing in \(x\) (taking \(\frac{a - y_t}{0} = -\infty\)). Hence the problem is formulated as a maximization increasing implicit stochastic DP.

**Remark.** In the original problem as formulated in [HKL+14], the supports of the random variables are finite and consist of rational numbers. By multiplying them with \(q_t\) they become finite integer intervals. Moreover, in the original formulation the state and action spaces are integer intervals that are not necessarily contiguous. Recall that in this paper we assumed for simplicity that the action and state spaces, as well as the random variables supports are all contiguous integer intervals. But the analysis and results carry also for any finite integer intervals.

B.6 Stochastic growth model

[HKL+14, Appendix A.9] defined \((\Delta = \alpha/\beta)\) is a given positive rational constant, \(p_t(\cdot), u_t(\cdot)\) are given nonnegative monotone increasing functions, and the support of \(D_t \geq 0\) consists of rational with a given denominator \(q_t\):

\[
g_t(I_t, x_t, D_t) = u_t(x_t), \quad f_t(I_t, x_t, D_t) = D_t p_t(I_t) + (1 - \Delta) I_t - x_t
\]

and formulated the problem as a maximization increasing explicit stochastic DP. We note that \(g_t(I_t, x_t, D_t)\) is deterministic so \(\mathbb{E}g_t(I_t, x_t, D_t) = u_t(x_t)\). Whenever \(p_t(I) = 0\) we get that \(\text{Prob}(f_t(I, x, D) \geq a) = 1\). Otherwise, we have \(\text{Prob}(f_t(I, x, D) \geq a) = \text{Prob}(D \geq \frac{a - (1 - \Delta)I + x}{p_t(I)} = 1 - F\left(\frac{a - (1 - \Delta)I + x}{p_t(I)} - q_t\right)\), which is monotone decreasing in \(x\). Hence the problem is formulated as a maximization increasing implicit stochastic DP.

**Remark.** In the original problem as formulated in [HKL+14], the supports of the random variables are finite and consist of rational numbers. By multiplying them with \(q_t\) they become finite integer intervals. Moreover, in the original formulation the state and action spaces are integer intervals that are not necessarily contiguous. Recall that in this paper we assumed for simplicity that the action and state spaces, as well as the random variables supports are all contiguous integer intervals. But the analysis and results carry also for any finite integer intervals.

B.7 Cash management problem

[HKL+14, Appendix A.10] defined \((\alpha)\) is a constant:

\[
g_t(I_t, x_t, D_t) = \alpha^{t-1}[c(x_t) + h(I_t - x_t - D_t)], \quad f_t(I_t, x_t, D_t) = I_t - x_t - D_t
\]

and formulated the problem as a convex explicit stochastic DP. By setting \(g^I_t = 0, \quad g^d_t(y) = \alpha^{t-1}c(y), \quad f^I_t(x, d) = x\) and \(g^g_t(y) = \alpha^{t-1}h(y)\) we get that the problem is a convex implicit DP.
\section{Statement of function ApxSet}

\begin{algorithm}[H]
\begin{algorithmic}[1]
\Function{ApxSet}{\(\varphi, D, x^*, K\)} [HKL\textsuperscript{+}14, Alg. 1]
\State \(x \leftarrow D^{\max}\)
\State \(W \leftarrow \{D^{\min}, D^{\max}\}\)
\While{\(x > D^{\min}\)}
\If{\(x > x^*\)} \State \(x \leftarrow \min \{\text{prev}(x, D), \min \{y \in D \mid y \geq x^* \text{ and } K\varphi(y) \geq \varphi(x)\}\}\)
\Else \State \(x \leftarrow \min \{\text{prev}(x, D), \min \{y \in D \mid K\varphi(x) \geq \varphi(y)\}\}\)
\EndIf
\State \(W \leftarrow W \cup \{x\}\)
\EndWhile
\State \Return \(W\)
\EndFunction
\end{algorithmic}
\caption{Constructing a \(K\)-approximation set for a unimodal \(\varphi : D \to \mathbb{R}\) that is minimized at \(x^*\).}
\end{algorithm}

For simplicity, we omit the parameter \(D\) in ApxSet whenever the domain of the function is clear from the context. Moreover, whenever function \(\varphi\) is monotone, we omit the parameter \(x^*\).

\section{Statement of function IndirectApxSet}

\begin{algorithm}[H]
\begin{algorithmic}[1]
\Function{IndirectApxSet}{\(\varphi, D, K\)} [HKL\textsuperscript{+}14, Alg. 2]
\State \(x \leftarrow D^{\max}\) and \(W \leftarrow \{D^{\min}, D^{\max}\}\)
\While{\(x > D^{\min}\) and \(K\bar{\varphi}(D^{\min}) < \varphi(x)\)}
\State \(x \leftarrow x' \mid x' < x \text{ and } K\bar{\varphi}(x') < \varphi(x) \text{ and } K\bar{\varphi}(\text{next}(x', D)) \geq \varphi(x)\)
\State \(W \leftarrow W \cup \{x, \text{next}(x, D)\}\)
\EndWhile
\State \Return \(W\)
\EndFunction
\end{algorithmic}
\caption{Constructing a subset of \(D\) for a function \(\varphi\) that approximates a nondecreasing function \(\varphi\).}
\end{algorithm}

For simplicity, we omit the parameter \(D\) in IndirectApxSet whenever the domain of the function is clear from the context.

\section{FPTAS for implicit stochastic increasing DP when FPTASs to calculate \(E_Dg_t(I, x, D)\) are given}

Instead of using Proposition 6.6 we will use the following variant on Proposition 6.1.

\begin{proposition}
Suppose the \(DP\) formulation (3) satisfy Condition 7(i) or Condition 7(ii). Let \(K', L', L'', t,\) and \(I_t\) be fixed values, where \(K', L' \geq 1, 1 \leq L'' \leq K'L', I_t \in \mathbb{S}_t,\) and \(t \in \{1, \ldots, T\}.\) Let \(g_t\) be as stated in Conditions 7(i) and 7(ii). Let \(\bar{G}_t(I, \cdot)\) be a monotone \(L''\)-approximation of \(E_{D_t}(g_t(I, \cdot), D_t))\). Let \(\bar{Z}f_{t+1}\) be a monotone \(L'\)-approximation of \(E_{D_t}(z_{t+1}(f_{t}(I, \cdot), D_t))\), and \(W\) be a \(K'\)-approximation set of \(\bar{Z}f_{t+1}\). Let
\[
\bar{z}_t(I_t) = \min_{x_t \in W} \{\bar{G}_t(I_t, x_t) + \bar{Z}f_{t+1}(I_t, x_t)\}. \tag{19}
\]
Then, \(\bar{z}_t(I_t)\) is an unnecessarily monotone \(K'L'\)-approximation value of \(z_t(I_t)\), and it can be determined in \(O((t\bar{G}_t + t\bar{Z}f_{t+1})|W|)\) time if the elements of \(W\) are given.

Our FPTAS reads as follows (step 7 is implemented by calling the FPTAS for \(E_Dg_t(I, x, D)\)).
1: Procedure FPTASIncDPgivenEG(\(\epsilon\))
2: \(K \leftarrow 3T\sqrt{1 + \epsilon}\), \(K' \leftarrow K\), \(\tilde{z}_{T+1} \leftarrow \text{COMPRESSINC}(g_{T+1}, s_{T+1}, K)\)
3: for \(t := T\) downto 1 do
4: \(\tilde{F}_t \leftarrow \text{COMPRESSINC}(F_t, [A_1, \ldots, B_t], K)\)
5: Let \(\tilde{Z}_{f_{t+1}}(I, \cdot)\) be an increasing \(K'K\)-approximation for \(Z_{f_{t+1}}(I, \cdot) := E_{D_t}(z_{t+1}(f(I, \cdot, D_t)))\)
6: \(\tilde{Z}_{f_{t+1}}(I, \cdot) := \text{COMPRESSINC}(\tilde{Z}_{f_{t+1}}(I, \cdot), S_t, K)\)
7: Let \(G_t(\cdot)\) be a monotone \(K'K\)-approximation for \(G_t(I, \cdot, D) := E_{D_t}(I, \cdot, D_t)\)
8: \(\tilde{z}_t \leftarrow \text{INDIRECTCOMPRESSINC}(\tilde{z}_t, S_t, K)\) /* \(\tilde{z}_t\) is as defined in (19) */
9: \(K' \leftarrow K'K^3\)
10: end for

**Algorithm 10:** An FPTAS for an increasing DP whenever FPTASs to calculate \(E_{D_t}(I, x, D)\) are given.