Good prophets know when the end is near

SIDDHARTHA BANERJEE, Cornell, School of Operations Research and Information Engineering, USA
DANIEL FREUND, MIT, Sloan School of Management, USA

We consider a class of online decision-making problems with exchangeable actions, where in each period a controller is presented an input type drawn from some stochastic arrival process and must choose an action, and the final objective depends only on the aggregate type-action counts. Such a framework encapsulates many online stochastic variants of common optimization problems with knapsack, bin packing and generalized assignment as canonical examples. In such settings, we study a natural model-predictive control algorithm. We introduce general conditions under which this algorithm obtains uniform additive loss (independent of the horizon) compared to an optimal solution with full knowledge of arrivals. Our condition builds on the compensated coupling technique of (Vera and Banerjee 2020), providing a unified view of how uniform additive loss arises as a consequence of the geometry of the underlying decision-making problem.

Our characterization allows us to derive uniform-loss algorithms for several new settings, including the first such algorithm for online stochastic bin-packing. It also lets us study the effect of other modeling assumptions, including choice of horizon, batched decisions, and limited computation. In particular, we show that our condition is fulfilled by the above-mentioned problems when the end of the time-horizon is known sufficiently long before the end. In contrast, if at a late stage, there is still uncertainty about the end of the time horizon we show that such uniform loss guarantees are impossible to achieve. We demonstrate the performance of our algorithm via large-scale experiments on real and synthetic data.

1 INTRODUCTION

In this work, we focus on a class of online decision-making problems characterized by three features:

(i) *Exogenous Randomness*: The system dynamics are subject to *exogenous* stochastic fluctuations, which we model as a random arrival of *input types*, conditionally independent of the system state, which become known to the controller at the start of each period.

(ii) *Exchangeable Actions*: The system evolves over a (almost sure) finite, but potentially a priori unknown, horizon, with the objective depending only on the aggregate counts of different actions taken over the horizon in response to each arrival type (and not on the order or timing of the actions).

(iii) *Limited Distributional Knowledge*: The controller has limited distributional information (e.g. noisy mean estimates) about arrivals; these are obtained, for example, from historical traces.

These three features are present in many problems of practical interest, across various fields. We formally define our general setting in Section 2; for the moment, however, the following canonical examples serve to illustrate some settings of interest:

- Online resource allocation (online packing): A finite supply of resources is allocated to arriving demand types, with varying requirements and valuations, so as to maximize overall rewards.
- Online cloud scheduling (bin-packing): Cloud-computing jobs with different resource requirements arrive over a finite horizon, and the controller needs to load them onto existing host machines with sufficient free resources, or start-up new host machines with a fixed budget of resources, while minimizing operating costs. Jobs remain till the end of the horizon, and once allocated to a machine, can not be re-allocated.
- Online routing in batch-processing systems (generalized assignment): Partitioning arriving compute jobs into batches in batch-processing systems, shipments among trucks in logistics systems, and online financial transactions among payment-processing firms with different contract structures. More generally, each arriving type is mapped to an action, and the final reward depends on the aggregate vector of counts of types mapped to each action.

In all the above settings, given distributional knowledge of arrivals, including the length of the time horizon, the underlying control problems can be formulated as *Markov Decision Processes* (MDP), and corresponding optimal solutions computed via dynamic programming (Bertsekas 2011). However, due to their combinatorial nature, these problems typically suffer from the 'curse of dimensionality', which makes optimal policies computationally intractable. Furthermore, such intractable optimal policies also do not provide easy benchmarks for evaluating the performance of simpler heuristics.

An alternate control paradigm in such settings is that of *model-predictive control*, wherein the controller makes decisions based on some offline benchmark. One natural class of benchmarks are the so-called *prophet benchmarks* – hindsight optimal policies with full information of arrivals. Controls based on such benchmarks are often simple and easy to interpret; moreover, the benchmark also provides a natural way of computing performance bounds for such policies. However, under *worst-case* input distributions (in particular, when the distributions can be chosen based on the horizon length T), it is known that any policy must incur additive loss compared to the prophet benchmark which grows with T; in particular, for the problems we study, existing algorithms typically incur either constant-factor multiplicative losses (when the hindsight optimal reward is small), or $\Omega(\sqrt{T})$ additive losses.[1] Significantly, several of these performance guarantees coincide with *minimax bounds* for these settings, and thus hold even under adversarial arrivals.

An alternate setting of interest is one in which input distributions are fixed beforehand, and then the problem size (horizon length T and budgets) is made larger. This scaling regime circumvents the pessimistic bounds of worst-case models, and is often more relevant in practice. Fixing the input distributions upfront immediately implies $O(\sqrt{T})$ additive losses, and begs the question if even better performance is achievable. In this context, recent work on resource allocation problems introduced the so-called *Bayesian prophet* framework (Arlotto and Gurvich 2019, Vera and Banerjee 2020, Vera et al. 2019) which achieves *uniform additive loss* (i.e., independent of T) compared to the hindsight optimal policy. In this work we extend the study of this framework to provide a simple, unified and exhaustive characterization of when uniform loss guarantees are obtainable, in terms of assumptions on the input distributions as well as the geometry of the underlying problem. Our characterization then allows us to extend these ideas to more general settings, and study the effect of various modeling constraints, including choice of horizon, batching and computation constraints. In summary, our contributions are as follows.

Beyond resource allocation. Our first contribution is to extend the Bayesian prophet framework beyond the resource allocation problems previously studied. We obtain a simple characterization for when uniform-loss algorithms can be obtained based on the geometry of the underlying decision-making problem. While our condition follows from a natural extension of the ideas introduced in (Vera and Banerjee 2020), taking this abstract view allows us to immediately extend the results to a wide class of online decision-making problems. The most striking example of this is to the online stochastic bin packing problem, for which we achieve the first instance-dependent uniform loss guarantee for instances that are not bounded waste (see related work).

Example 1.1. Consider the following online stochastic bin packing instance from Csirik et al. (2006): over a known time horizon with T periods, items of sizes 1,3,4,5, and 8 arrive in each period, i.i.d., with respective probabilities $p_1 = .25, p_3 = .25, p_4 = .125, p_5 = .25, p_8 = .125$. Bins have size 10, and in each period an irrevocable decision is made to place an item in an existing bin or to

[1] The expected additive loss vis-à-vis the hindsight-optimal (see Eqn. 1) is often referred to as the *regret* of a policy (Gupta and Radovanović 2020, Kleinberg 2005, Vera and Banerjee 2020) As this term is interpreted differently across communities, we instead use additive loss in this work.

place it in a new one. Figure 3 of Gupta and Radovanović (2020) displays state-of-the-art algorithms incurring $\Theta(\sqrt{T})$ loss on this instance, whereas ours (see Figure 4 in Section 5) achieve $O(1)$.

Knowing when the end is near. Going beyond particular problem settings, our condition allows us to study the effect of different modeling choices on algorithmic performance. Our first such exploration is on the impact of the choice of a time horizon, i.e., the number of arrivals. The existing literature usually treats the time horizon as either known upfront[2] or adversarial, i.e., the horizon may end abruptly after any arrival. In contrast, many real-world scenarios exhibit a middle ground, wherein the number of arrivals is a priori unknown but becomes clear significantly before the last arrival. Two real-world settings with such a feature arise in refugee resettlement[3] (Ahani et al. 2021, Bansak and Paulson 2022) and budget pacing (Agarwal et al. 2014, Gaitonde et al. 2022). Our results give a tight characterization of how early the uncertainty of the time horizon needs to be resolved for uniform loss guarantees to be achievable, as we illustrate next.

Example 1.2. Consider a marketing campaign with a budget of $\$K$ per arriving customer. T customers arrive sequentially, being drawn, independently, from an unknown distribution with finite support that dictates the cost of targeting the customer and the expected value thereof; the value of T is also a priori unknown. The campaign aims to maximize its expected value while spending at most $\$K \times T$ (with some penalty for overspending). After $\hat{T} \leq T$ customers, the decision is made that the campaign will be ended after $T - \hat{T}$ additional customers arrive; with $T - \hat{T} \in O(\sqrt{T})$ loss guarantees must be $\Omega(\sqrt{T})$ (Proposition 2) whereas having $T - \hat{T} \in \Omega(T^{\frac{1}{2}+\epsilon})$, for $\epsilon > 0$, ensures that $O(1)$ loss can be achieved (Theorem 2.2).

Superficially, $\hat{T}$ may seem like a purely theoretical construction. We would argue though that it represents a very real practical situation, e.g., in the context of a marketing campaign, a decision being made on a Monday to end the campaign by Friday. Our results characterize how much time is needed between the decision to end the campaign and the actual campaign end (relative to the length of the campaign) for uniform loss guarantees to be attainable.

Infinite horizon approximation. A particularly interesting application of the above perspective arises when comparing finite (fixed) horizon settings to infinite-horizon discounted settings. The latter is commonly used as a simplification, especially when arrivals are stationary, as it greatly reduces the problem complexity (since the resulting problem has a one-step Bellman equation). In Proposition 3 (see also Figure 3) we show that such approximations necessarily involve an additive loss of at least $\Omega(\sqrt{T})$ when compared to the hindsight optimal policy (i.e., the 'value of information'); this contrasts with the $O(1)$ loss our algorithms achieve when the time horizon is known or becomes known sufficiently long before it ends. Thus, our results suggest that there are settings in which infinite-horizon discounted problems should not be used to approximate finite-horizon problems.

We note that there is a parallel line of work on online stopping problems with freezing, pioneered originally by (Samuel-Cahn 1996); recent papers have considered online packing with random horizons (Alijani et al. 2020, Hajiaghayi et al. 2007) in the context of prophet inequalities (constant factor approximations to the optimum). While our results share common features with these, the prophet inequality settings allow for arbitrary distributions and thus always involve linear regret.

Batched Controls. The settings in Example 1.2 also immediately generalize to more common batched-control settings. For example, suppose T arrivals occur in batches over time where T is initially

[2]This includes works with Poisson arrivals occurring over an interval of known length, e.g., (Jasin and Kumar 2012).

[3]In refugee resettlement the number of refugees arriving in a year can vary significantly, especially when policies change, e.g., as (Ahani et al. 2021) describe for the US in 2018; however, such a reduction is likely better modeled through an early signal than through an adversary suddenly ending the horizon.

unknown, but the last batch contains at least $T^{1/2+\epsilon}$, for some $\epsilon > 0$, arrivals, and upon arrival of the last batch, it is revealed that it is the last one. This would be a special case of Example 1.2. We further illustrate the application of our results to batching settings in Section 5.2.

Lazy resolving. Our work also improves upon the existing Bayesian prophet studies with respect to its computational requirements (see Appendix C). Whereas the work of Vera and Banerjee (2020) solves a packing LP in each of the T time periods, we present algorithms that only require infrequent resolving. For i.i.d. arrivals with a known distribution we show that resolving $O(\log \log T)$ times suffices, which for network revenue management matches the result of (Bumpensanti and Wang 2020) for which we provide a simpler and more general proof.

Online payment routing with QoS-penalties. Our work was initially motivated by the following application, which we term *online payment routing problem*. Consider online merchants that are large enough to work with more than one credit card payment processing firm. According to Square (2020) "the average cost of processing payments for U.S. businesses that do between $10,000 and $250,000 in annual payments volume is between 2.87 percent and 4.35 percent per transaction." A part of this transaction cost is due to credit card payment processing firms (Square is such a firm). Oftentimes these firms offer special rates for merchants whose sales transactions fulfill average quality-of-service (QOS) conditions, e.g., Square (2020) suggests a transaction volume of at least $250,000 and an average transaction value of $15. We define the *online payment routing problem* (see Section 2) which identifies, among different payment processing firms, the cost-minimizing firm for each payment, and show that our algorithm has uniform loss guarantees for it. While we use this specific application to highlight the application of our ideas to a particular real-world problem, we note that this form of average-QOS violation penalties is common in many other long-term contracts, and our approach may find use in such settings as well.

Numerical results. We complement our analytic results with a comprehensive numerical study that illustrates the order-wise better performance of our algorithms compared to the existing ones. We also illustrate our impossibility results numerically, showing that uncertainty on the horizon length makes $O(1)$ loss unachievable for any algorithm. Finally, our numerical work includes a data-driven case study of the *online payment routing problem* in which arrivals occur in batches. Whereas most of our numerical results are based on instances with relatively few input types (up to 20), the data-driven case study is based on real data with hundreds of types (see Figure 7). For the performance of our algorithms on less well-posed instances of the bin packing problem we refer the reader to Sinclair et al. (2022) who have studied the performance of a variant of our algorithm in allocating compute resources for real-world Microsoft Azure workloads.

1.1 Related Work

Our work is most similar to, and indeed, builds upon and generalizes the Bayesian prophet framework of Vera and Banerjee (2020), who study multidimensional knapsack problems like network revenue management. Previously, uniform loss guarantees for multisecretary, i.e., online knapsack with unit weights, were obtained by Arlotto and Gurvich (2019) via a related algorithm. A follow-up work by Vera et al. (2019) had generalized earlier results to consider more and different kinds of uncertainty, e.g., when values can be probed. In contrast, our work focuses on providing uniform loss guarantees for problems outside of resource allocation, and on identifying the fundamental assumptions required to achieve uniform loss.

For the multisecretary problem, Arlotto and Xie (2020) and Bray (2019) show that uniform-loss guarantees break down when the arrival distribution is continuous; this, in a sense, shows the necessity of our thickness assumptions on the arrivals (Assumption 1). We show similar impossibility

results to identify further settings where uniform loss guarantees are not attainable when (i) when there is uncertainty about the length of the time horizon (Propositions 2 and 3), (ii) arrivals of a particular type are too rare or *too rare* towards the end of the time-horizon (Propositions 4 and 7).

Our algorithms combine ideas from Bumpensanti and Wang (2020) and Vera and Banerjee (2020) to improve upon both. In contrast to Vera and Banerjee (2020) and Vera et al. (2019) we develop algorithms that only resolve infrequently (Sun et al. (2020) highlight the complexity of resolving in large-scale applications). In contrast to Bumpensanti and Wang (2020) our algorithms resolve at times that are dynamically set (rather than offline) and thereby yield uniform loss guarantees for network revenue management that do not depend on the relative scaling of supply and demand (Vera and Banerjee (2020) show that the guarantee by Bumpensanti and Wang (2020) can fail when supply and demand are scaled differently). We extend the techniques from both to settings beyond resource allocation problems.

Bin Packing. The most significant application of our results is to online stochastic bin packing. In this well-studied problem, we are able to obtain an instance-dependent $O(1)$ suboptimality gap, i.e., the expected difference between the number of bins opened by our algorithm and the offline optimal number of bins is bounded by an instance-dependent constant that is independent of T.

Traditionally, the bin packing literature has measured its objective as *waste*, i.e., the amount of empty bin space across all bins opened by an algorithm. Taking bin sizes as constant sized, a classical result by Courcoubetis and Weber (1986) states that in the case of discrete size distributions occurring i.i.d., the amount of waste of an optimal packing (i.e., the waste the clairvoyant would incur) scales either as $O(T), O(\sqrt{T})$ or $O(1)$ over a long time horizon — these regimes are referred to as linear waste (LW), perfectly packable (PP), and bounded waste (BW). Noticeably, even with LW it is possible to have the additional waste of an algorithm relative to the optimal packing scale as $O(1)$; this is exactly the guarantee we provide, and it holds regardless of regime. In contrast, state-of-the-art algorithms in the literature so far have only guaranteed $O(1)$ loss bounds for the bounded waste regime. In more detail: Csirik et al. (2006) studied the so-called sum-of-squares algorithm for bin packing, and showed that when the arrival distribution is known, the algorithm can be adapted to obtain $O(1)$ loss in the BW regime, and $O(\sqrt{T})$ in the other regimes. For unknown distributions, they found that SS has no strong guarantees in the LW regime; (Gupta and Radovanović 2020) designed a distribution-oblivious algorithm with a $O(\sqrt{T})$ guarantee that holds across all regimes. In contrast, our results provide a single algorithm that obtains $O(1)$ loss across all regimes, though, in contrast to existing work, this hides an instance-dependent constant. Moreover, our framework also provides a $O(1)$ loss guarantee in the multidimensional online stochastic bin packing and minimum cost packing with heterogeneous bin costs (referred to by Gupta and Radovanović (2020) as the configuration model); to the best of our knowledge, the only previous guarantee here was $O(\sqrt{T})$, obtained by Gupta and Radovanović (2020). An important distinction to Gupta and Radovanović (2020) is that their algorithm never requires the distribution of arrival types; in contrast, our algorithm either knows the distribution of arrival items a priori, or learns it as items arrive.

Prior to Csirik et al.'s analysis of the sum-of-squares algorithm, there had been a long line of literature for different heuristics, such as First Fit (e.g., Albers and Mitzenmacher (2000), Shor (1986)), and Best Fit (e.g., Coffman Jr et al. (1991), Shor (1991)), under particular (often continuous) distributions. As pointed out by Gupta and Radovanović (2020), these simple heuristics tend to do badly even under very simple discrete distributions (e.g., bins of size 6 and items of size 2 and 3 with equal probability). Somewhat orthogonal to these is a packing heuristic proposed by Rhee and Talagrand (1993) which uses the item sizes seen so far to form a bin packing LP relaxation; they prove that when the item sizes are i.i.d. from a general distribution (the support of the distribution

can be continuous), their algorithm has a loss of $O(\sqrt{T}\log(T))$. Subsequent to our work, Liu and Li (2021) obtained a $O(\sqrt{T})$ guarantee that holds, when T is known, both for general i.i.d. distributions and under the random permutation model.

Online decision-making paradigms. Before proceeding, we briefly discuss the connections of our approach to three alternate approaches in online decision-making – online convex optimization, competitive analysis, and prophet inequalities. Our results on online LPs and convex programs are closely related to work on online convex optimization (Agrawal and Devanur 2014, Agrawal et al. 2014), which considers similar settings, but with adversarial arrivals, and with a fixed set of online benchmark policies (as opposed to our offline benchmarks). As a consequence, the techniques and results with the two approaches are somewhat incomparable. We note though that a hybrid approach similar to ours is proposed in (Devanur et al. 2019) for a similarly flavored analysis.

In contrast, competitive analysis goes beyond online convex optimization by studying the performance of online policies for combinatorial problems compared to the offline optimal under worst-case arrivals (Buchbinder et al. 2009). This literature is also connected to the work on *prophet inequalities* (Düetting et al. 2017, Kleinberg and Weinberg 2012), which are concerned with solving stochastic packing and pricing problems to obtain worst-case competitive ratio bounds (i.e., maxmin bounds over all distributions). By focusing on the worst case, such approaches can typically only provide constant multiplicative guarantees, which correspond to an additive loss of $\Omega(T)$, in comparison to our uniform loss guarantees. Note though that our loss guarantees depend on parameters of the distribution; understanding the exact dependence and interpolating between these results remains an open question.

2 PRELIMINARIES

We now introduce the class of stochastic online decision-making problems that we study in this work. At a high level, we consider finite-horizon control problems, where, in each period, the controller is presented with a random arriving *type*, and must choose a feasible *action* for that type; crucially, we assume that *the overall objective can be expressed as a function that only depends on the final state-action counts.* Such a setting admits a rich class of problems, including the ones previously discussed as well as many others. Note, however, that we do not consider settings where the order and/or timing of decisions affects the objective.

Throughout we use $[k]$ for the set $\{1, 2, \ldots, k\}$ and write $[t, t'] = \{t, t+1, \ldots, t'\}$. In addition, our arguments often use the number of future periods after a given period t; denoting the *discrete-time finite-horizon*[4] by T, the number of future periods is $(T - t)$, and we always denote it by $\bar{t}$. When the time-horizon is unknown, the value of $\bar{t}$ is unkown, and our algorithms do not rely on $\bar{t}$ then. As is commonly done, we use inequality between vectors to refer to component-wise inequalities, and use $Bin(n, p)$ to denote a binomial distribution with n trials and probability p.

Periods, Arrivals, and Types. In each time period t, the controller observes an *arrival* θ_t drawn from an *underlying finite set of types* Θ, where $|\Theta| = k$, and must then choose an action. Throughout the main body of the paper we assume that the types are drawn i.i.d. from an unknown distribution over Θ, where p_θ denotes the probability for an arrival to be of type θ. We denote the number of arrivals of type θ in $[1, \ldots, t]$ by $N_\theta(t)$, and define N_θ as the number of type θ arrivals over the entire horizon. Furthermore, we let $\bar{N}_\theta(t)$ denote the number of *future* type θ arrivals after period t, i.e., ones in $[t+1, T]$, so $N_\theta(t) + \bar{N}_\theta(t) = N_\theta$. We make the following assumption on the arrival thickness of each arrival type.

Assumption 1 (Arrival Thickness). *For each type $\theta \in \Theta$, we have $p_\theta \geq \beta$ for some $\beta > 0$.*

[4]Vera and Banerjee (2020) explain in their Appendix B.1 how a continuous Poisson process reduces to this setting.

We often rely on a bound obtained from combining a union bound with Hoeffding's inequality.

LEMMA 2.1. *In period t, we have* $\mathbb{P}\left[\max_\theta\{|\bar{N}_\theta(t) - \bar{t}p_\theta|\} > \sqrt{\alpha \bar{t}\log(\bar{t})}\right] \le 2k\bar{t}^{-2\alpha}$.

For completeness, we include the proof in Appendix A.1. In Appendix C we discuss new results for a setting where the distribution is known, but arrivals may be rarer and change over time.

Actions. Each type $\theta \in \Theta$ is associated with a constant number of actions, one of which must be played by the controller for each arrival of that type. For example, in an online knapsack problem, types correspond to (value, size) tuples, and actions correspond to *accept*/*reject*; for online bin packing, types correspond to item sizes, while actions correspond to *maximal bin configurations* in which an arriving item may be placed (see end of this section).

For notational simplicity, we assume that the number of available actions is the same regardless of type; denoting the set of actions associated with type θ as $\mathcal{A}_\theta$ we can then write, with some fixed ℓ, $\mathcal{A}_\theta = \{a_{\theta 1}, a_{\theta 2}, \ldots, a_{\theta \ell}\}$ for each θ; note this is w.l.o.g, as we can add dummy actions. Next, for any $t \in [T]$, we define $x_{\theta j}(t)$ to be the *type-action counts* in periods $[1, t]$, i.e., the number of times action $a_{\theta j}$ is used, upon arrival of a type θ, in periods $\{1, \ldots, t\}$.

Objective Function. Let $\vec{x} = \{x_{\theta j}[1, T]; \theta \in \Theta, j \in [\ell]\}$ denote any vector of type-action counts; observe that for all types θ we must have taken some action for each arrival implying that $\sum_{j\in[\ell]} x_{\theta j} = N_\theta$ and $\sum_\theta \sum_j x_{\theta j} = T$. We assume that the overall objective can be expressed as minimizing a given objective $f(\vec{x})$ that *only depends on the final state-action counts*, whereas the order and timing of decisions do not affect the objective. We remark that, in some cases, $f_{\vec{N}}(\vec{x})$ may be a lower bound on the objective we truly want to measure, e.g., for bin packing it is a relaxation of the number of bins that need to be opened; proving our guarantees with respect to such a lower bound implies them with respect to the true objective. In the main body of the paper we focus on functions $f_{\vec{N}}(\cdot)$, with parameters $\vec{c}_1, \vec{c}_2, A, D$, and B of the following type:

$$\begin{aligned} f_{\vec{N}}(\vec{x}) \quad = \quad & \min_{\vec{z}} \vec{c}_1 \cdot \vec{x} + \vec{c}_2 \cdot \vec{z} \\ & \text{s.t.} \quad \sum_j x_{\theta j} = N_\theta \;\forall\, \theta \\ & A\vec{x} - D\vec{z} \le \vec{B} \sum_\theta N_\theta \\ & z_{\theta j} \ge 0 \;\; \forall\, \theta, j, \end{aligned}$$

where D is a non-negative matrix with at least one positive value in every row. Intuitively, $\vec{x}$ models decisions and $A\vec{x}$, $\vec{B}\sum_\theta N_\theta$ model constraints, which $D\vec{z}$ relaxes at a penalty. At the end of this section we write two of our problems in this format, which we expand on in Appendix B. More broadly, we focus on functions that fulfill two properties, both guaranteed for $f_{\vec{N}}(\cdot)$ as above.

PROPOSITION 1. *Suppose, given $\vec{N}$, a function $f_{\vec{N}}(\cdot)$ can be written as above; then there exist constants λ, δ that depend on A and D but not on $\vec{c}_1, \vec{c}_2, B$ or $T = \sum_\theta N_\theta$ such that*

(i) $f_{\vec{N}}$ is λ-Lipschitz continuous on the set $\{\vec{x} : \sum_j x_{\theta j} = N_\theta\}$;

(ii) for any $\vec{y}$, any $\vec{N}$ and $\vec{N}'$ with $|\vec{N}|_1 = |\vec{N}'|_1$, and any $\vec{x} \in \arg\min_{\vec{x}\ge\vec{y}}\{f_{\vec{N}}\}$, there exists

$$\vec{x}' \in \arg\min_{\vec{x}'\ge\vec{y}}\{f_{\vec{N}'}\} \text{ with } |\vec{x} - \vec{x}'|_\infty \le \delta|\vec{N} - \vec{N}'|_\infty.$$

The proof of the first part of the proposition is straightforward, the second part follows from classical perturbation results in linear programming theory (Mangasarian and Shiau 1987); we defer both to Appendix A.1. We first make a unique identification assumption:

Assumption 2. *For any $\vec{N}, \vec{y}$, we are given a unique tie-breaking rule for selecting a solution $\vec{x}$ that minimizes $f_{\vec{N}}$ subject to $\vec{x} \geq \vec{y}$.*

For the objective functions we consider this is loss of generality, as we can use a lexicographic tie-breaking rule, or alternately, use infinitesimal perturbations of $\vec{c}_1$ to guarantee this without affecting the constants λ and δ from Proposition 1. In particular, we note that this is critically different from a 'non-degeneracy' assumption on the certainty-equivalent problem (i.e., the ex-ante or deterministic linear program, which we refer to as the DLP in Section 3). This assumption is convenient for our analysis as, under an a priori unknown time horizon, it allows our arguments to refer to *the* optimal solution.

Time horizon. Most of our results assume that the time horizon T is a priori unknown (i.e., T refers to an unknown quantity). However, we assume instead there exists an exogenous *heads-up* period $\hat{T}$ at which the value of T is revealed.

Assumption 3 (Heads-up). *The true horizon T is revealed at a heads-up time $\hat{T}$, which is a priori unknown, and independent of past arrivals and actions. For example, $\hat{T}$ may be chosen by an oblivious adversary. However, the heads-up time and true horizon satisfy $T - \hat{T} > T^{\frac{1}{2}+\epsilon}$ for some known $\epsilon > 0$.*

We assume without loss of generality that $\hat{T} \geq T/2$ (for $\hat{T} < T/2$, the problem is strictly easier).

Performance Benchmark. We define the loss of an online algorithm as the expected difference between the objective achieved by the algorithm, and the objective achieved by the optimal actions taken by a *clairvoyant algorithm* on the same sample path — the clairvoyant always knows the sample path, including its length, i.e., the time horizon. To do so, we first define the objective of an optimal clairvoyant algorithm, for fixed arrivals $\vec{N}$, as

$$\texttt{OPT}(\vec{N}) = \min_{\vec{x}} f_{\vec{N}}(\vec{x}).$$

$\texttt{OPT}(\vec{N})$ is a random variable which needs full information of $\vec{N}$. Now, the loss of an algorithm ALG

$$\mathcal{L}_{\texttt{ALG}} := \mathbb{E}\left[f_{\vec{N}}(\vec{x}^{\texttt{ALG}}) - \texttt{OPT}(\vec{N})\right], \tag{1}$$

where $\vec{x}^{\texttt{ALG}}$ denotes the algorithm's actions. Notice that the expectation here is not only over the number of arrivals of each type, $\vec{N}$, but also over the order of the arrivals which may influence $\vec{x}^{\texttt{ALG}}$. Our goal is to design an algorithm ALG for which $\mathcal{L}_{\texttt{ALG}}$ can be bounded independently of T, and to understand the necessary conditions for any algorithm to have this property.

Theorem 2.2. *There exists an algorithm* ALG *and a constant $M(\beta, \lambda, \delta, \epsilon, \ell)$, independent of T, such that $\mathcal{L}_{\texttt{ALG}} \leq M(\beta, \delta, \lambda, \epsilon, \ell)$.*

In Section 4 we show that the assumptions made in this section are a minimal set of assumptions to achieve uniform loss guarantees, i.e., dropping Assumptions 1 or 3 or replacing f by a function that does not fulfill the properties in Proposition 1, uniform loss guarantees are, in general, unachievable.

Examples of our framework in action

Before diving into our algorithmic ideas and analyses, we present how functions $f_{\vec{N}}(\cdot)$ as described before can be used to capture two of the problems we aim to solve: bin packing and online payment routing. In this section we define these problems, and describe how $f_{\vec{N}}(\cdot)$ can model them. In Appendix B we discuss online packing problems like network revenue management or AdWords. The problems in Appendix B contain budgets, which $\vec{B}$ in the formulation of $f_{\vec{N}}$ is used to encode; in contrast, both problems here set $\vec{B} = \vec{0}$.

Bin packing. In online stochastic bin packing types are characterized by their size: an item of type $\theta \in \Theta$ is of size s_θ, bins are of size κ, where $s_\theta \leq \kappa \ \forall \theta$, and κ is assumed to be of constant size. In period t an item of size s_{θ_t} arrives and needs to be irrevocably packed either into an existing bin, that contains items of combined size at most $\kappa - s_{\theta_t}$, or into a new bin. We let $\mathcal{J}$ denote the set of maximal bin configurations. For example, if items have sizes 2, 3, or 5, and bins have capacity 9, then the maximal configurations correspond to $\mathcal{J} = \{\{3,3,3\},\{2,2,5\},\{2,2,2,2\},\{2,2,2,3\},\{3,5\}\}$. Denoting by j a particular configuration, we write $\alpha_{\theta j}$ for the reciprocal of the number of times item θ fits into a bin of configuration j, e.g., with $j = \{2,2,5\}$ we have $\alpha_{2j} = 1/2$; when θ is not part of a bin configuration, e.g., 3 is not in j, we let $\alpha_{\theta j} = 0$. We define the set of actions for type θ as $\{\alpha_{\theta j} : A_{\theta j} > 0\}$, i.e., the set of (maximal) configurations in which an item of type θ can be placed; thus, with $x_{\theta j}$ counting the number of type-θ items in type-j configurations, $\sum_j x_{\theta j} = N_\theta$ ensures that each item is placed into a bin. The objective is to minimize the number of bins used, where we allow opening a bin fractionally, meaning, e.g., that if a bin of configuration $\{2,2,5\}$ has only one 2 allocated to it, we would count it as only half a bin — this fractional relaxation of the objective is within $|\mathcal{J}|$, independent of T, of the integer optimum. To capture the objective as $f_{\vec{N}}$ we let $\vec{c}_1 = \vec{B} = 0$ and $\vec{c}_2$ be a $|\mathcal{J}|$-dimensional vector of all 1s; A and D enforce the constraints $\alpha_{\theta j} x_{\theta j} \leq z_j \ \forall \theta, j$; then, for a given $\vec{x}$, the optimal $z_j := \max_\theta\{\alpha_{\theta j} x_{\theta j}\}$ is the (fractionally relaxed) number of configuration-j bins that fit $x_{\theta j}$ items of type θ (for every θ), and $|\vec{z}|_1 = c_2 \cdot \vec{z}$ is the total number of open bins. Thus, $f_{\vec{N}}(\vec{x})$ evaluates to the (relaxed) bin packing objective.

Online payment routing. In our online payment routing problem, a controller observes arrivals of transactions that need to be processed; each transaction has a value, v_θ, that characterizes its type $\theta \in \Theta$. The controller needs to decide for each transaction which of ℓ different payment processing firms to send it to (Elliott 2020). Payment processing firm j charges the merchant, to process a payment of value v_θ, a base charge b_j *per transaction* sent to j, and a charge $m_j v_\theta$ that is proportional to the value of each transaction sent to j, i.e., it charges $b_j + m_j v_\theta$. In addition, j may have a minimum average charge a_j across payments that enforces that the total paid to j is at least the number of payments routed to j multiplied by a_j.[5] Denoting by $x_{\theta j}$ the number of payments of value v_θ to be routed to firm j, and by z_j the combined charges paid to firm j, the objective can be written as $\sum_{j \in \mathcal{J}} z_j$, where A and D enforce the constraints $\sum_\theta x_{\theta j}(b_j + m_j v_\theta) \leq z_j$ and $\sum_\theta x_{\theta j} a_j \leq z_j \ \forall j$, relying, again, on $\vec{c}_1 = \vec{B} = 0$ and $\vec{c}_2$ being a $|\mathcal{J}|$-dimensional vector of all 1s.

3 ALGORITHMIC FRAMEWORK

In this section we present our main algorithmic ideas, guarantees, and analyses. We present Algorithm 1, which consists of two different for-loops, one for the periods before and one for the periods after $\hat{T}$. In the periods before $\hat{T}$ the main difficulty arises from p_θ and T being unknown; after $\hat{T}$, we know T and we have already learned good estimates of p_θ; the remaining difficulty stems from the inherent stochastic fluctuations in the last periods. Lemma proofs are included in Appendix A.2.

3.1 Periods before $\hat{T}$

Suppose in a given period t, having already taken actions $x_{\theta j}^{\mathsf{ALG}}(t-1)$ in periods $[1, t-1]$, we knew all of the remaining arrivals, i.e., we knew $N_\theta \ \forall\, \theta$. Then, we would solve the hindsight-optimal optimization in period t, denoted $\mathtt{OPT}^{\,t}$, which can be written as

$$\min_{\vec{x}} f_{\vec{N}}(\vec{x}) \qquad \text{s.t.} \qquad \vec{x} \geq \vec{x}^{\mathsf{ALG}}(t-1) \ \ \forall\, \theta, j. \tag{$\mathtt{OPT}^{\,t}$}$$

[5] For firms with a small b_j and a larger m_j, such a_j ensures that they do not receive exclusively small transactions.

In an abuse of notation we will overload $\texttt{OPT}^{\,t}$ to also, depending on context, refer to either the objective or the optimal solution of this optimization. Having it refer to the objective allows us to rewrite the loss in (1) as $\mathcal{L}_{\texttt{ALG}} = \mathbb{E}\left[f_{\vec{N}}(\vec{x}^{\texttt{ALG}}) - \texttt{OPT}(\vec{N})\right] = \mathbb{E}\left[\texttt{OPT}^{\,T} - \texttt{OPT}^{\,1}\right]$ since $\vec{x}^{\texttt{ALG}}(0) = \vec{0}$. Further, if we had access to the solution $\texttt{OPT}^{\,t}$ it would be easy to identify what actions we should take, for each arrival, in the remaining periods. Indeed, in period t, taking any action j that fulfills $\texttt{OPT}^{\,t}_{\theta j} \geq x^{\texttt{ALG}}_{\theta j}(t-1) + 1$ guarantees that the objective value of $\texttt{OPT}^{\,t+1}$ is the same as that of $\texttt{OPT}^{\,t}$. In particular, though taking such an action leads to $\texttt{OPT}^{\,t+1}$ being more constrained, the optimal solution to $\texttt{OPT}^{\,t}$ remains feasible for $\texttt{OPT}^{\,t+1}$. This insight is formalized and generalized in the next lemma.

Lemma 3.1. *Consider any two periods $t_a < t_b$; if, for every action $a_{\theta j}$, we have $x^{\texttt{ALG}}_{\theta j}(t_b - 1) \leq \texttt{OPT}^{\,t_a}_{\theta j}$, then the objective values $\texttt{OPT}^{\,t_a} = \texttt{OPT}^{\,t_b}$.*

Of course, as we do not know N_θ, we cannot solve $\texttt{OPT}^{\,t}$; however, we will see that we can empirically construct reasonable estimates of N_θ that suffice to guarantee that the probability of an action incurring loss is small in most periods. The key difficulty in obtaining such estimates is that, a priori, we know neither T nor p_θ, and even if we did know them we would not know N_θ.

Unknown horizon. Suppose first we did know p_θ but neither the horizon T nor the arrivals N_θ. In that case, we cannot create good estimates of N_θ; however, we can solve the following deterministic linear program (DLP), in which p_θ, which is independent of T, stands in for N_θ.

$$\min_{\vec{x} \geq \vec{0}} f_{\vec{p}}(\vec{x}). \tag{DLP}$$

The optimal solution – for which we again abuse notation by overloading DLP – has a natural interpretation: $\texttt{DLP}_{\theta j}$ denotes the relative frequency that a type θ arrives and an action j is taken by an optimal policy when the number of arrivals of each type exactly matches their expectation. Further, we can combine DLP with Lemma 2.1 and Proposition 1 to obtain a natural high-probability lower bound on $\texttt{OPT}_{\theta j}$, i.e., on the number of times the clairvoyant optimal solution plays each action. Roughly speaking, since (Lemma 2.1) arrivals of each type θ are close to $p_\theta T$ with high probability, it must be the case (Proposition 1, ii) with the same probability that, for every $a_{\theta j}$, the values of $\texttt{OPT}_{\theta j}$ and $\texttt{DLP}_{\theta j}$ are also close to each other. The next lemma formalizes this.

Lemma 3.2. *With probability at least $1 - 2kT^{-\frac{T^\epsilon}{\delta^2}}$ we have for every action $a_{\theta j}$ that*

$$\texttt{OPT}_{\theta j} \geq T \times \texttt{DLP}_{\theta j} - [T\log(T)]^{\frac{1+\epsilon}{2}}.$$

If we knew p_θ, Lemmas 3.1 and 3.2 would suggest a natural policy before $\hat{T}$ which does not require us to know T: upon an arrival of type θ, take action $a_{\theta j}$ with probability $\texttt{DLP}_{\theta j}/p_\theta$. Then, by period $\hat{T}$, each action $a_{\theta j}$ has been taken $Bin(\hat{T}, \texttt{DLP}_{\theta j})$ times which is, with high probability (for sufficiently large T), less than $T \times \texttt{DLP}_{\theta j} - [T\log(T)]^{\frac{1+\epsilon}{2}}$. Thus, with high probability, we would find that $\texttt{OPT}^{\,\hat{T}} = \texttt{OPT}^{\,1}$. Unfortunately, without knowing p_θ, we cannot follow this policy. We next explore how we can form empirical estimates that allow us nonetheless to appropriately imitate it.

Unknown p_θ and empirical DLP. Though p_θ is unknown, in period t we have already observed t i.i.d. samples. Instead of solving DLP, we can then solve the following empirical estimate of the DLP:

$$\min_{\vec{x}} f_{\vec{N}(t)/t}(x). \tag{EDLP(t)}$$

A convenient feature of $\mathsf{EDLP}(t)$ is that, with high probability in t, the optimal solution to EDLP is approximately equal to that of DLP. Formally, we can define a bad event in period t as

$$\mathcal{S}_t = \left\{ \max_{\theta} |N_\theta(t) - t p_\theta| > \frac{[t \log(t)]^{\frac{1+\epsilon}{2}}}{\delta} \right\}.$$

By Proposition 1, a bad event is necessary for the inequality $|\mathsf{DLP}_{\theta j} - \mathsf{EDLP}(t)_{\theta j}| \geq [t \log(t)]^{\frac{1+\epsilon}{2}}$ to hold for some action $a_{\theta j}$. We now bound the probability of bad events.

Lemma 3.3. *The probability of a bad event can be bounded by* $\mathbb{P}[\mathcal{S}_t] < 2kt^{-\frac{2t^\epsilon}{\delta^2}}$; *as a result, with probability at least* $1 - 2kt^{-\frac{2t^\epsilon}{\delta^2}}$ *we have for every action* $a_{\theta j}$ *that* $|\mathsf{DLP}_{\theta j} - \mathsf{EDLP}(t)_{\theta j}| < [t \log(t)]^{\frac{1+\epsilon}{2}}$.

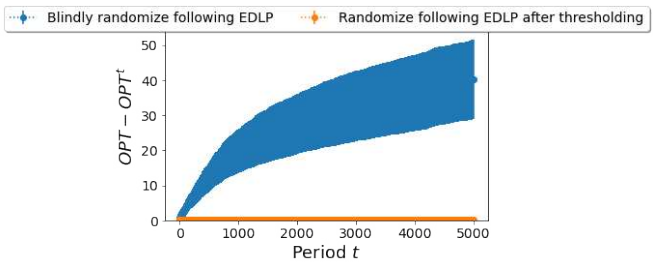


Fig. 1. Plot of 95% CIs of loss incurred in the first $t \leq \frac{T}{2}$ periods of a packing problem instance (arrivals have unit demand and value of either 1 or 2 with probability 1/2, $T = 10^4$, and we may accept at most $T/2$ arrivals); we plot loss for (i) blindly following EDLP, or (ii) following EDLP with thresholds as in Algorithm 1.

ALGORITHM 1:

– Initialize $x^{\mathsf{ALG}}_{\theta j}(0) = 0; F_{\theta j} = 0 \ \ \forall \theta, j$

for $t = 1, \ldots, \hat{T}$ **do**

– Solve optimization problem $\mathsf{EDLP}(t)$ and pick any $j' \in \arg\max_j \{\mathsf{EDLP}_{\theta_t j}(t)\}$;

– For $j \in [\ell], j \neq j'$, set $y_{\theta_t j}(t) = 0$ if $\mathsf{EDLP}_{\theta_t j}(t) < \frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}}$; otherwise $y_{\theta_t j}(t) = \mathsf{EDLP}_{\theta_t j}(t)$;

– Set $y_{\theta_t j'}(t) = 1 - \sum_{j \neq j'} y_{\theta_t j}(t)$;

– In period t, use action $a_{\theta_t j}$ with probability $y_{\theta_t j}(t)$;

– Set $x^{\mathsf{ALG}}(t)$ by incrementing $x^{\mathsf{ALG}}_{\theta_t j}(t-1)$ by 1 and leaving all other values unchanged;

end

for $t = \hat{T} + 1, \ldots, T$ **do**

if $F_{\theta_t j} < 1$ *for every action* j **then**

– Solve optimization problem $\mathsf{TDLP}(t)$;

– Set $F_{\theta j} = \left\lfloor \mathsf{TDLP}_{\theta j}(t) - x^{\mathsf{ALG}}_{\theta j}(t) - 2\bar{t}^{\frac{3}{4}}\sqrt{\log \bar{t}} \right\rfloor \ \ \forall \theta, j$

end

– In period t, use action $a_{\theta_t j}$ for some $j \in \arg\max_j \{F_{\theta_t j}\}$; reduce $F_{\theta_t j}$ by 1;

– Set $x^{\mathsf{ALG}}(t)$ by incrementing $x^{\mathsf{ALG}}_{\theta_t j}(t-1)$ by 1 and leaving all other values unchanged;

end

Given the reasoning after Lemma 3.2, it is tempting to conclude from Lemma 3.3 that, similarly to previously randomizing based on DLP, we could now randomize based on $\mathsf{EDLP}(t)$ to ensure

that $\texttt{OPT} - \texttt{OPT}^{\hat{T}} \in O(1)$ with high probability, i.e., that uniform loss is still attainable at time $\hat{T}$. Figure 1 shows that this is not the case. Indeed, blindly following $\texttt{EDLP}(t)$ can lead to $\Omega(\sqrt{T})$ loss; instead, we need to ensure that the actions we take are ones where $\texttt{DLP}_{\theta j} > 0$; to do so, we threshold based on Lemma 3.3 by setting to 0 all values in EDLP for which we do not know, with high probability in t, that DLP puts positive weight on them. The resulting algorithm is described in the first for-loop of Algorithm 1. Relative to randomizing via DLP, the algorithm may incur loss for two reasons:

- When solving $\texttt{EDLP}(t)$ instead of DLP, we may use an action $a_{\theta j}$ — even after thresholding — that DLP would not use at all. Intuitively, in period t, this requires $\mathcal{S}_t$ to hold true, and the corresponding probability of this event is summable.
- When thresholding the probability of some action $a_{\theta j}$ to 0 we increase the probability of another, thereby risking that the latter is used too often. Intuitively, since the gap $T - \hat{T} \in \Omega(T^{\frac{1}{2}+\epsilon})$ is much larger than the combined thresholded probabilities, roughly $O(T^{\frac{1+\epsilon}{2}})$, there should be a sufficient buffer for these actions to not be taken too often.

We next formalize these intuitions to bound the expected loss they cause by a constant.

Lemma 3.4. *Let $\mathcal{S} = \sum_{t=1}^{\hat{T}} \mathbb{1}_{\{\mathcal{S}_t\}}$ be the count of bad events. There exists a constant T_0 such that, under Algorithm 1, for $T \geq T_0$, we have $\sum_{\theta,j}(x_{\theta j}^{\texttt{ALG}}(\hat{T}) - \texttt{OPT}_{\theta j})^+ \leq \mathcal{S}$ with probability at least $1 - \frac{1}{T}$.*

Lemma 3.4 enables, for $T \geq T_0$, the following bound on $\mathbb{E}\left[\texttt{OPT}^{T} - \texttt{OPT}\right] =$

$$\mathbb{E}\left[\texttt{OPT}^{T} - \texttt{OPT}^{\hat{T}} + \texttt{OPT}^{\hat{T}} - \texttt{OPT}\right] \leq \quad \mathbb{E}\left[\texttt{OPT}^{T} - \texttt{OPT}^{\hat{T}}\right] + 2\lambda\left(\hat{T}\mathbb{P}\left[\left(\sum_{\theta,j}\texttt{OPT}_{\theta j}^{\hat{T}} - \texttt{OPT}_{\theta j}\right)^{+} > \mathcal{S}\right] + \mathbb{E}[\mathcal{S}]\right)$$
$$< \mathbb{E}\left[\texttt{OPT}^{T} - \texttt{OPT}^{\hat{T}}\right] + 2\lambda\left(1 + 2k(\delta^{\frac{2}{\epsilon}} + \pi^2/6)\right).$$

The first inequality uses the fact that the loss incurred in any period is bounded by 2λ since $f(\cdot)$ is λ-Lipschitz continuous; thus, we bound the loss in the first $\hat{T}$ periods by $2\lambda\mathbb{E}[\mathcal{S}]$ plus $2\lambda\hat{T}$ times the small-probability event from the lemma, i.e., in that case we bound by the maximum loss possible over $2\hat{T}$ periods. The second line also uses that

$$\mathbb{E}[\mathcal{S}] \leq \sum_{t=1}^{\infty}\mathbb{P}[\mathcal{S}_t] \leq \sum_{t=1}^{\infty} 2kt^{-\frac{2t^{\epsilon}}{\delta^2}} \leq 2k\left(\sum_{t=1}^{\delta^{\frac{2}{\epsilon}}} 1 + \sum_{t=1}^{\infty} t^{-2}\right).$$

We next explore how the second for-loop in Algorithm 1 ensures that $\mathbb{E}[\texttt{OPT}^{T} - \texttt{OPT}^{\hat{T}}]$ is constant.

3.2 Final Periods after $\hat{T}$

We now describe the second for-loop of Algorithm 1. It is known that randomizing based on DLP incurs $\Omega(\sqrt{T})$ loss over the entire time-horizon; thus, after $\hat{T}$, we cannot continue to just follow DLP. Instead, for $t > \hat{T}$, we solve an optimization that uses the number of remaining periods:

$$\min_{\vec{x}} f_{\frac{T}{t}\vec{N}(t)}(\vec{x}) \qquad \text{s.t.} \qquad \vec{x} \geq \vec{x}^{\texttt{ALG}}(t-1) \;\; \forall\, \theta, j. \tag{TDLP(t)}$$

Comparing $\texttt{TDLP}(t)$ and $\texttt{OPT}^{t}$ we observe similar optimization problems: the only difference stems from $\frac{T}{t}\vec{N}(t) \neq \vec{N}(t)$, or rather, as the two are equal for the first t arrivals, the differences between $\frac{\bar{t}}{t}N_\theta(t)$ and $\bar{N}_\theta(t)$. However, with high probability (Lemma 2.1) both are close to $\bar{t}p_\theta$ for t and $\bar{t}$ sufficiently large. With this in mind, we use $\texttt{TDLP}_{\theta j}(t)$ to construct lower bounds on $\texttt{OPT}_{\theta j}^{t}$.

Lemma 3.5. *Consider a period $t > \frac{T}{2}$ with actions $x^{\mathsf{ALG}}_{\theta j}(t-1)$ already taken in periods $[1, t-1]$; with probability at least $1 - 4k\bar{t}^{-\left(\frac{2\sqrt{\bar{t}}}{\delta^2}\right)}$ we have $\forall \theta, j : \mathtt{OPT}^t_{\theta j} \geq \left\lfloor \mathtt{TDLP}_{\theta j}(t) - 2\bar{t}^{\frac{3}{4}}\sqrt{\log \bar{t}} \right\rfloor$.*

Based on Lemmas 3.1 and 3.5, in any period $t > \hat{T}$ we can construct an explicit lower confidence bound (LCB), $\lfloor \mathtt{TDLP}_{\theta j}(t) - 2\bar{t}^{\frac{3}{4}}\sqrt{\log \bar{t}} \rfloor$, on how often $\mathtt{OPT}^{\,t}$ uses $a_{\theta j}$ in the periods from t through T. Such LCBs provide a budget of actions for each type; Algorithm 1 accordingly sets $F_{\theta j}$ equal to that LCB. Then, upon arrival of some $\theta_{t'}$ in period t' we can take any action $a_{\theta_{t'} j}$ that has a positive $F_{\theta_{t'} j}$, and reduce the action budget $F_{\theta_{t'} j}$ by 1 afterwards. Assuming the LCBs hold true, we are guaranteed that the objective $\mathtt{OPT}^{\,t}$ remains attainable. Thus, in period t' we either (i) use an action $a_{\theta_{t'} j}$ with positive $F_{\theta_{t'} j}$, or (ii) if no such action exists, we solve $\mathtt{TDLP}(t')$ to create new LCBs. We show below that the loss incurred that way can be bounded by summing over the probabilities of LCBs not holding true. In the last periods we may find that, even after resolving, $F_{\theta j} < 1 \,\forall\, j$, i.e., no action can be taken without violating a LCB. In that case the algorithm picks one of the actions *most frequently taken* in $\mathtt{TDLP}(t)$; we prove that this happens only in $O(1)$ periods, and thus does not affect our guarantee.

Proof of Theorem 2.2. Given the derivations after Lemma 3.4, we first bound $\mathbb{E}\left[\mathtt{OPT}^{\,T} - \mathtt{OPT}^{\,\hat{T}}\right]$. Denote by $T_1 = \hat{T}, T_2, \ldots, T_s$ the periods in which Algorithm 1 solves $\mathtt{TDLP}(t)$, and denote by T_η the first period in which Algorithm 1 solves $\mathtt{TDLP}(t)$ and it is the case that for some θ all budgets $F_{\theta j} < 1$; we denote by $\mathcal{T}_i$ the event that the LCB created in period T_i fails to hold true:

$$\mathcal{T}_i = \left\{ \exists \theta, j : \mathtt{OPT}^{\,T_i}_{\theta j} < \left\lfloor \mathtt{TDLP}_{\theta j}(T_i) - 2\bar{T}_i^{\frac{3}{4}}\sqrt{\log \bar{T}_i} \right\rfloor \right\}.$$

Then we can bound

$$\mathbb{E}\left[\mathtt{OPT}^{\,T} - \mathtt{OPT}^{\,\hat{T}}\right] \leq 2\lambda \mathbb{E}\left[\sum_{i=1}^{\eta-1}(T_{i+1} - T_i)\mathbb{P}[\mathcal{T}_i] + \bar{T}_\eta\right] \leq 2\lambda \mathbb{E}\left[\sum_{i=1}^{\eta-1}\bar{T}_i\mathbb{P}[\mathcal{T}_i] + \bar{T}_\eta\right].$$

From Lemma 3.5 we know that $\mathbb{P}[\mathcal{T}_i] \leq 4k\bar{T}_i^{-\frac{2\sqrt{T_i}}{\delta^2}}$; when $\bar{T}_i > \delta^4$ this bound is less than $4k\bar{T}_i^{-2}$; thus, we can bound the above as $2\lambda\left(4k\left(\delta^4 + \sum_{t=\hat{T}}^{T}\bar{t}^{-2}\right) + \mathbb{E}\left[\bar{T}_\eta\right]\right) \leq 2\lambda\left(4k(\delta^4 + \pi^2/6) + \mathbb{E}\left[\bar{T}_\eta\right]\right)$.

For $\mathbb{E}[\bar{T}_\eta]$ we construct a loose upper bound that assumes $\beta > \sqrt{\frac{2\log(T/2)}{T}}$, and let T_2 be such that for $T > T_2$ this inequality holds true. Then, applications of Lemma 2.1 show that, with probability greater $1 - \frac{k}{T}$, every θ and every $t > \hat{T}$ fulfill $N_\theta(t) > p_\theta t/2 \,\forall\, \theta$; we condition on this to be the case, implying that $\mathtt{TDLP}(t)$ is constrained, for every θ, by $\sum_j \mathtt{TDLP}_{\theta j}(t) - x^{\mathsf{ALG}}_{\theta j}(t) \geq p_\theta \bar{t}/2 \geq \beta\bar{t}/2$.

Now, to have $t \leq T_\eta$, it must be the case that when setting $F_{\theta j}$ at time t we find for every action $a_{\theta j}$ that $F_{\theta j} = \left\lfloor \mathtt{TDLP}_{\theta j}(t) - x^{\mathsf{ALG}}_{\theta j}(t) - 2\bar{t}^{\frac{3}{4}}\sqrt{\log \bar{t}} \right\rfloor < 1$, which requires $\ell(2\bar{t}^{\frac{3}{4}}\sqrt{\log \bar{t}} + 1) > \beta\bar{t}/2$; as the right-hand side is linear in $\bar{t}$ and the left-hand side is sublinear, there exists a constant T_1 such that for $\bar{t} > T_1$ the right-hand side is larger, meaning the condition cannot be fulfilled for $\bar{t} > T_1$. Thus, we bound $\mathbb{E}[\bar{T}_\eta] \leq (1 - \frac{k}{T})T_1 + k < T_1 + k$ and find that

$$\mathcal{L}_{\mathsf{ALG}} = \mathbb{E}\left[\mathtt{OPT}^{\,T} - \mathtt{OPT}^{\,\hat{T}} + \mathtt{OPT}^{\,\hat{T}} - \mathtt{OPT}\right] \leq 2\lambda\left(1 + 4k\left(\delta^4 + \pi^2/3\right) + (2\delta^2)^{\frac{1}{\epsilon}} + T_1 + k\right)$$

holds for $T > \max\{T_0, T_2\}$, with T_0 as in Lemma 3.4. In order to drop the condition $T > \max\{T_0, T_2\}$ we set $M(\beta, \delta, \lambda, \epsilon, \ell)$ as the maximum of this expression and $2\lambda \max\{T_0, T_2\}$. □

Extensions. In this section we analyzed the i.i.d. setting with unknown p_θ and a priori unknown horizon length. In Appendix C we show that similar ideas yield $O(1)$ loss guarantees when arrivals are not i.i.d., but (i) T is known, and (ii) the arrival distribution admits some thickness/concentration assumption, and (iii) a single sample of T arrivals is available. In that setting, we can also relax the requirements on the objective function, and handle a slightly more general class of objective functions. Further, in that setting we show that computational requirements are significantly relaxed as resolving optimization problems, like EDLP, DLP, etc., $O(\log(T))$ times suffices to obtain $O(1)$ loss. Indeed, when arrivals are i.i.d. from a known distribution, we show that our algorithm resolves just $O(\log\log(T))$ times.

4 IMPOSSIBILITY RESULTS

Our impossibility results in this section focus on three different aspects of our assumptions: we begin by studying the informational structure around the horizon length (Assumption 3); next, we consider the geometry of the objective function (Property (ii) of Proposition 1); finally, we consider the density of arrivals of each type (Assumption 1). All our results in this section are based on the same lemma that follows from applying the Berry-Esseen Theorem (Berry 1941, Esseen 1956, Shevtsova 2011) to our settings. Effectively, it shows that over a horizon of a given length, with constant probability, we observe at most a given upper bound or at least a given lower bound of arrivals of each type. We state the Lemma here and prove it in Appendix A.3.

Lemma 4.1. *Consider an interval $1, \dots, \tau$ where in each period an arrival is of type θ i.i.d. with probability $p_\theta \in (0,1)$, and let N_θ denote the number of arrivals of type θ in the interval. Then*

$$\forall m \exists \xi, \tau_0 > 0 : \mathbb{P}[N_\theta \geq p_\theta \tau + m\sqrt{\tau}] \geq \xi \; \forall \tau \geq \tau_0.$$

Notice that by replacing p_θ by $1 - p_\theta$ the lemma also implies that

$$\forall m \exists \xi, \tau_0 > 0 : \mathbb{P}[N_\theta \leq p_\theta \tau - m\sqrt{\tau}] \geq \xi \; \forall \tau \geq \tau_0.$$

Late heads-up. In the spirit of Example 1.2 we first show that no algorithm can have a $O(1)$ loss guarantee when Assumption 3 is relaxed to allow for $T - \hat{T} = \sqrt{T}$.

Proposition 2. *There exists a family of instances to the online stochastic bin packing problem, with $T - \hat{T} = \sqrt{T}$, for which no algorithm can obtain loss $o(\sqrt{T})$.*

Proof. Consider an online stochastic bin packing problem instance in which bins have size 3, items are of size either 1 or 2 with probability $\frac{1}{2}$, and the time horizon has length either T or $2T$, with probability $\frac{1}{2}$ each, which is revealed after $T - \hat{T}$ periods. Notice that $\mathcal{J} = \{\{1,1,1\},\{1,2\}\}$. Denote by a_1 the action of placing an item of size 1 into a bin of configuration $\{1,1,1\}$ and by a_2 the action of placing an item of size 1 into a bin of configuration $\{1,2\}$, and let x_{12} denote the number of times an item of size 1 is placed in a bin of configuration $\{1,2\}$, x_{11} denote the number of times an item of size 1 is placed in a bin of configuration $\{1,1,1\}$ (as items of size 2 are always placed in the latter, we focus only on the actions for items of size 1). We define the following random variables:

- A is the number of items of size 1 in the first $\hat{T}$ periods (with $\hat{T} - A$ items of size 2);
- B is the number of items of size 1 in the next $T - \hat{T}$ periods (with $T - \hat{T} - B$ items of size 2);
- C is the number of items of size 1 in periods $T+1, \dots, 2T$ (with $T - B$ items of size 2).

Further, let $\Delta_T = 2A + B - T$, which, if greater 0, denotes the number of items of size 1 in the first $\hat{T}$ periods that cannot be placed into $\{1,2\}$ bins with items of size 2 that arrive in the first T periods (subtract from A 1s in the first $\hat{T}$ periods the $T - (A+B)$ 2s that arrive in the first T periods). We also define $\Delta_{2T} = 2(A + B + C - T)$, where $\Delta_{2T} = 2(A + B + C - T) \leq 0$ implies that there are at

least T 2s among the $2T$ periods. We focus on the following event: $E = \{\Delta_T \geq T - \hat{T}\} \cap \{\Delta_{2T} \leq 0\}$. With $T - \hat{T} = \sqrt{T}$, Lemma 4.1 implies that $\mathbb{P}[E]$ is bounded away from 0. Further, with $T - \hat{T} = \sqrt{T}$, we argue that when E occurs, an algorithm must incur $\Omega(\sqrt{T})$ loss on at least one of the two possible time horizon lengths. Under E, the optimal packing under the horizon of length T puts $T-A-B$ (this is the number of 2s) 1s in bins of configuration $\{1,2\}$, and the remaining items of size 1 in bins of configuration $\{1,1,1\}$ — for an objective of $T-A-B+\frac{2(A+B)-T}{3} = \frac{2T-(A+B)}{3}$. Now, consider the actions of an algorithm ALG which gives an objective of $x_{11}^{\text{ALG}}(T)/3 + (A+B) - x_{11}^{\text{ALG}}(T) = A + B - \frac{2}{3}x_{11}^{\text{ALG}}(T)$. This means the loss of ALG is

$$\left(A + B - \frac{2}{3}x_{11}^{\text{ALG}}(T)\right) - \left(\frac{2T-(A+B)}{3}\right) = \frac{2}{3}\left(2(A+B) - T - x_{11}^{\text{ALG}}(T)\right) \geq \frac{2}{3}\left(\Delta_T - x_{11}^{\text{ALG}}(T)\right).$$

At the same time, over a horizon of length $2T$, $\Delta_{2T} < 0$ implies there are more items of size 2 than of size 1, so the optimal packing places all items, of either type, in $\{1,2\}$ bins, and for any three items of size 1 in a $\{1,1,1\}$ bin, there is one bin created that the optimal packing would not create, so the loss incurred is at least $\frac{x_{11}^{\text{ALG}}(T)}{3}$. The result follows since decisions $x_{11}^{\text{ALG}}(\hat{T})$ at time $\hat{T}$ lead to

$$\mathcal{L}_{\text{ALG}} \geq \frac{\mathbb{P}[E]}{2}\max\left\{\frac{x_{11}^{\text{ALG}}}{3}, \frac{2}{3}\left(\Delta_T - x_{11}^{\text{ALG}}\right)\right\} \geq \frac{\mathbb{P}[E]}{9}\Delta_T \in \Omega(\sqrt{T}). \quad \square$$

Recall from Section 1.1 that known algorithms in the literature achieve anytime loss $O(\sqrt{T})$ on this instance, as it is perfectly packable. Thus, Proposition 2 implies that while there is value in knowing the end of the time horizon $T^{\frac{1}{2}+\epsilon}$ periods before the end (see Theorem 2.2), there is effectively, up to constant factors, no value in knowing it $T^{\frac{1}{2}}$ periods before the end.[6]

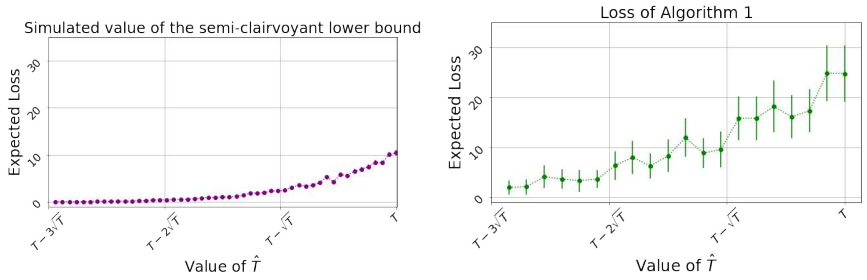
The unavoidable loss is also illustrated in our numerical results in Figure 2. In Figure 2(a) we simulate the *semi-clairvoyant* lower bound on the instance from the proposition over a horizon of length $T = 10^6$. While the lower bound remains small throughout, it starts to grow around $T - \hat{T} \approx 1.5\sqrt{T}$. For comparison, Figure 2(b) shows, for a time horizon of length $T = 10^4$, the loss of Algorithm 1. Here, we naturally see that the loss is larger, but in particular we also find that around $T - \hat{T} \approx 1.5\sqrt{T}$ it grows significantly. We also find that around $\hat{T} \approx T - 2.5\sqrt{T}$ the benefit of an earlier *heads-up* vanishes for both the semi-clairvoyant and the algorithm. This illustrates that knowing the exact value of T when *the end is near* is necessary to achieve small loss, whereas knowing it initially, or knowing the exact arrivals, is not.

Our next result sheds light on the value of Assumption 3 as compared to, e.g., the assumption of a geometrically distributed horizon length. We include the proof in Appendix A.4.

Proposition 3. *Suppose T is a geometrically distributed random variable with mean τ; without additional information, such as a heads-up as in Assumption 3, the optimal policy is based on a (potentially intractable) one-step Bellman equation. This policy incurs an expected loss of $\Omega(\sqrt{\tau})$.*

The lower bound in the proposition does not negate the $O(1)$ expected loss in our results; instead, it highlights that knowing the horizon end sufficiently long, i.e., $\Omega(T^{\frac{1}{2}+\epsilon})$ ahead of time, allows the optimal policy an improvement of $\Omega(\sqrt{T})$. Figure 3 shows, for an online packing instance with geometric horizon length, how the loss of the optimal solution increases with τ.

[6] While Proposition 2 focused on a bin packing instance, one can similarly construct an instance of an online packing problem for the specific setting in Example 1.2: customer types have unit cost and targeting them yields, type-dependent, a value of either 1 or 2. Arrivals are of either type with probability 1/2, the campaign budget, K in the example, is 1/2 per arrival (with additional spend being penalized), and the decision for each arriving customer is whether or not to target them; the campaign length is either T or $2T$ with probability 1/2. Theorem 2.2 implies that $O(1)$ loss is achievable when $T - \hat{T} \in \Omega(T^{\frac{1}{2}+\epsilon})$ whereas a similar argument to the one above implies that $\Omega(\sqrt{T})$ loss is unavoidable when $T - \hat{T} = \sqrt{T}$.



(a) Loss of semi-clairvoyant (b) Loss of algorithm

Fig. 2. The dependence of loss on $\hat{T}$ illustrates the necessity of knowing *when the end is near* for the semi-clairvoyant (Proposition 2) and Algorithm 1.

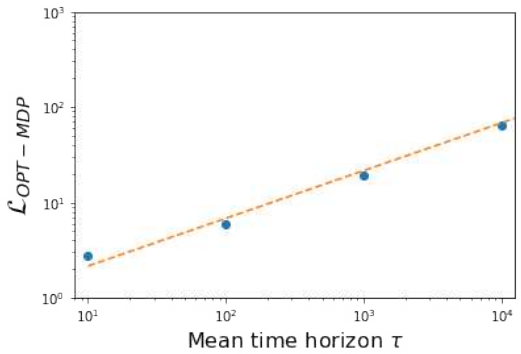


Fig. 3. Plot of the clairvoyant objective and the loss of the optimal policy in an instance with geometric time horizon length with mean τ; as the mean horizon length increasies from 10 to 10000, without a heads-up as in Assumption 3, the loss of the optimal policy grows with it.

Instability of optimal solution. Instead of considering functions $f_{\vec{N}}$ as described in Section 2, consider the following problem over a known time horizon T: in each period, with probability $1/2$, we need to irrevocably accept/reject an arrival type 1 or type 2; we may accept $T/2$ arrivals in total, and our goal is to maximize the number of accepted arrivals of *one type*, i.e.,

$$f_{\vec{N}}(\vec{x}) \quad = \quad \max\{x_1, x_2\} \qquad \text{s.t.} \quad x_1 + x_2 \leq T/2 \qquad \text{and} \quad x_i \leq N_i \text{ for } i \in \{1, 2\}.$$

Though one can always guarantee an objective of $T/2$ when $\vec{N}$ is known, it is easy to see that the optimal policy (accept θ_1, and thereafter only accept arrivals of that type) in this problem incurs $\Omega(\sqrt{T})$ loss: with probability $1/2$ it picks the type of which there are fewer than half the arrivals, and the gap is likely of order $\sqrt{T}$ (see Lemma 4.1). At the same time, the problem fulfills property (i) of Proposition 1 (Lipschitz-continuity) and Assumptions 1 and 3; thus, this problem suggests that a dependence of our guarantee on δ is fundamental.

No thickness. We next show that Assumption 1 is necessary for $O(1)$ loss guarantees. Consider an online packing problem over T time periods with (i) arrivals types $\Theta = \{1, 2, 3\}$, (ii) there is a budget to accept $T/2$ arrivals, (iii) the value of accepting an arrival of type i is i, and the arrival probabilities are (iv) $p_1 = 1/2, p_2 = \frac{1}{\sqrt{T}}, p_3 = 1/2 - \frac{1}{\sqrt{T}}$.

Proposition 4. *In the instance above, no algorithm can obtain $O(1)$ uniform loss guarantees.*

The proof of the proposition is similar to that of Proposition 2, analyzing the number of accepted type-2 arrivals in the first $T/2$ periods; we include it in Appendix A.4.

5 NUMERICAL RESULTS

In this section we numerically verify the uniform loss guarantees of Algorithm 1 for bin-packing and the online payment routing problem. We compare the performance of Algorithm 1, to two natural control policies – static certainty-equivalent control (also referred to as static randomized control, or the *fluid policy*), and randomized control based on adaptive certainty-equivalent (sometimes called fluid-policy with resolving). When considering bin packing, we also compare to the well-known sum-of-squares (SOS) algorithm (Csirik et al. 2006), discussed in Section 1.1. We first describe the static and the adaptive randomized control policies.

Static Randomized Control. The static randomized control policy (SRC) solves the DLP at the beginning of the time horizon to obtain the fraction of times each action shall be used for each type over the entire time horizon, assuming arrivals are as in expectation. Thereafter, it randomizes by using action $a_{\theta j}$ for arrivals of type θ with probability $\frac{\mathsf{DLP}_{\theta j}}{p_\theta}$.

Adaptive Randomized Control. The adaptive randomized control policy (ARC) is a resolving heuristic that solves a variant of $\mathsf{TDLP}(t)$, with access to the exact values of p_θ and T, in each round:

$$\min_{\vec{x}} f_{\vec{N}(t)+\bar{t}\times\vec{p}}(\vec{x}) \qquad \text{s.t.} \qquad \vec{x} \geq \vec{x}^{\mathsf{ALG}}(t-1) \;\; \forall\, \theta, j. \tag{ETDLP(t)}$$

Thereafter, if the arriving item in period t is of type θ it uses action $a_{\theta j}$ with probability $\frac{\mathsf{ETDLP}_{\theta j}(t)}{N_\theta(t)+\bar{t}p_\theta}$.

5.1 Uniform loss of Algorithm 1

Bin packing instance. Our first bin packing instance is described in Example 1.1. In Figure 4 we display for Algorithm 1, ARC, SRC, and SOS the average results over 15 random instances as well as corresponding 95% confidence intervals. Noticeably, only the loss of Algorithm 1 does not grow with T. We remark that Figure 3(b) in (Gupta and Radovanović 2020) shows results for their algorithm and SOS on this same instance, and demonstrates that their algorithm performs significantly worse on it than SOS, while also obtaining a loss that scales as $\Omega(\sqrt{T})$.

In Figure 5 we display the loss of SOS, SRC, and Algorithm 1 for a more complicated bin packing instance with bins of size 20 and items of size $\{1, 2, \ldots, 20\}$, that arrive i.i.d. uniformly at random. The log-log plot shows that the loss of SRC and SS both scale at rate $\Theta(\sqrt{T})$. In contrast, though this instance (with $k = 20$ and $\ell = 490$) is less well-behaved than the one in Example 1.1 our algorithm performs remarkably well, and obtains a uniform loss guarantee that is far below the one our theoretical bounds guarantee.

Online payment routing problem. Payment amounts in our instance are either $v_\theta \in \{2, 5, 8\}$ with i.i.d. probabilities $p_2 = 0.6, p_3 = 0.3, p_8 = 0.1$. Provider A charges $b_A = 0.40\$$, and $m_A = a_A = 0\$$ per payment, i.e., processing each payment costs 0.40\$ regardless of the transaction value, whereas provider B charges a proportional $m_B = 0.10\$$ and $b_B = 0\$$ per dollar processed (e.g., processing a 8\$ payment would cost 0.80\$ with that provider). Further, B also requires the merchant to pay a minimum average amount of $a_B = 0.30\$$ per payment processed. As Figure 6(a) shows, only Algorithm 1 obtains uniform loss. We remark that this example is intentionally designed to guide the ARC into non-uniform loss, and requires only a slight perturbation for ARC to obtain uniform loss. For example, Figure 6(b) shows the example from Figure 6(a), but with 2, 5, and 8\$ appearing with probabilities .5, .4, and .1 rather than .6, .3, and .1; in this case, the ARC does get uniform loss.

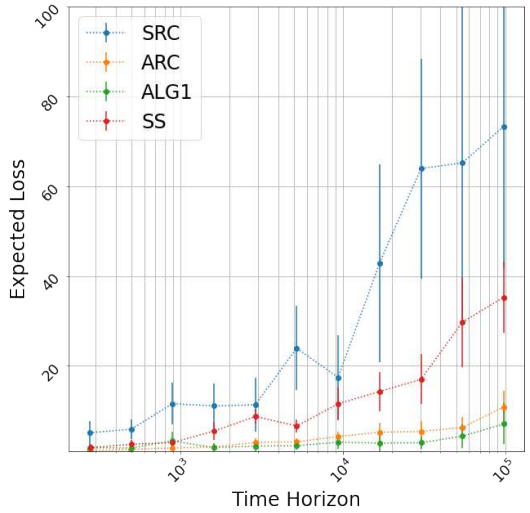


Fig. 4. Expected loss $\mathcal{L}_{\mathsf{ALG}}$ of Algorithm 1, ARC, SRC, and SOS for the bin packing instance from Example 1.1, induced over time horizons of varying length where $\hat{T} = .9 \times T$.

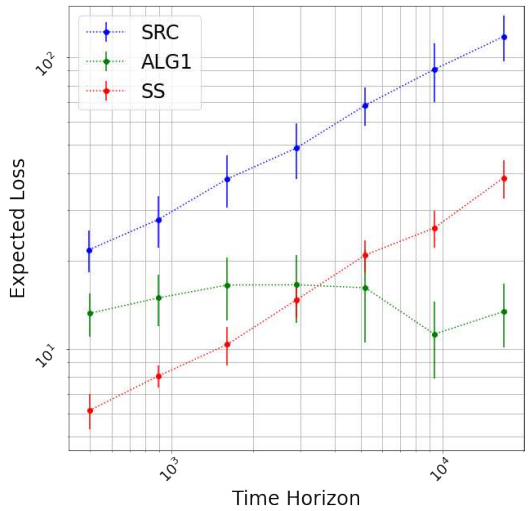
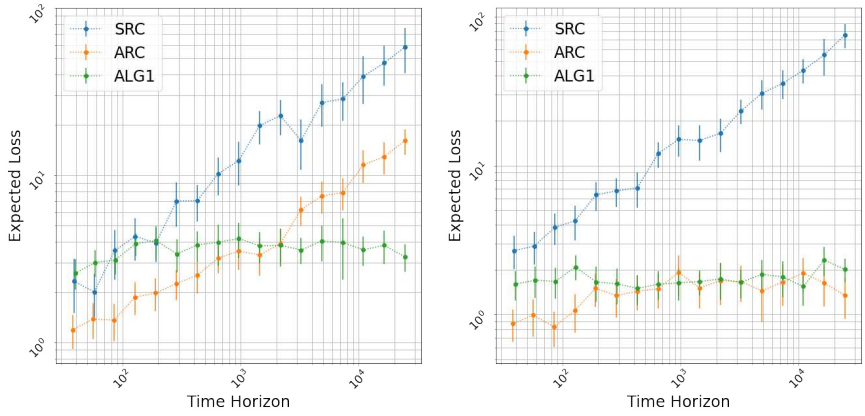


Fig. 5. Expected loss $\mathcal{L}_{\mathsf{ALG}}$ of Algorithm 1, SRC, and SOS for a bin packing instance with bins of size 20 and items of size $\{1, \ldots, 20\}$ with $p_\theta = 1/20$ for every θ, and $\hat{T} = .9 \times T$.

However, in contrast to Algorithm 1, ARC relies on knowledge of p_θ and T; SRC generally does not achieve uniform loss guarantees for the online payment routing problem.

5.2 Case study: batching one day of credit card processing fees

In this last part of our numerical investigation we work with a real-world data set of credit card payments (see Electronic Supplement) over the course of one month. To better illustrate the kind of batch settings we previously alluded to, we study the following policy.



(a) Dual-degenerate instance (b) Non-degenerate instance

Fig. 6. The expected loss of SRC, ARC, and Algorithm 1 is plotted for two instances of the online payment routing problem. The first is dual-degenerate with ARC having loss $\Omega(\sqrt{T})$, whereas in the second (not dual-degenerate) it has loss $O(1)$, like Algorithm 1; SRC has loss $\Theta(\sqrt{T})$ in both.

Static Randomized Control then Batch. The static randomized control then batch policy (SRC-B) imitates the SRC policy for the first $\hat{T}$ periods, and then observes the remaining $T - \hat{T}$ arrivals at once (*in one batch*), i.e., it can make the last $\hat{T}$ decisions with full information of $\vec{N}$. Viewed differently, SRC-B is the hybrid clairvoyant for SRC in period $T - \hat{T}$.

Corollary 5.1. *Under the same assumptions as in Theorem 2.2 SRC-B obtains uniform loss.*

Proof. Based on the same reasoning as in Lemma 3.4 (see also paragraph after Lemma 3.2) under SRC-B, with high probability, after $T - \hat{T}$ periods the optimal objective is still attainable. However, as the remaining decisions can be made in one batch, they can be made without incurring any additional loss. Thus, with high probability, SRC-B incurs no loss at all.

Data set. The data (Dal Pozzolo et al. 2015) for our numerical study is displayed in Figure 7. For each day of a month we observe the number of payments of each amount that need to be processed. We assume there are two different payment processing firms: firm A, which charges an average fee of at least \$.30 per payment, no base fee, and \$.10 per dollar of value processed; and firm B which charges a flat fee of \$.40 per payment. In our study we repeatedly sample T transactions from the data. In Figure 8 we compare 4 different policies: naive benchmarks that route all payments to either firm A or B, SRC, and SRC-B, wherein we batch all payments that occur on the last day of the month. Unsurprisingly, routing all payments to a single firm incurs a huge loss relative to the optimum. This indicates that with providers as heterogeneous as in our example, merchants may find significant value in balancing between providers via some kind of optimized policy like SRC. We also observe the uniform loss guarantee for SRC-B, as well as the increasing loss of SRC. For the number of transactions we consider (up to 10^7), we find that uniform loss algorithms like SRC-B or Algorithm 1, give an improvement of a factor of almost 100.

REFERENCES

Agarwal D, Ghosh S, Wei K, You S (2014) Budget pacing for targeted online advertisements at linkedin. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining,*

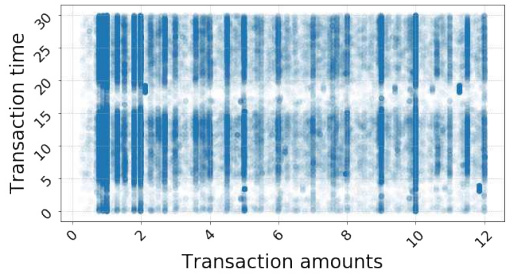


Fig. 7. Scatterplot displaying the data points underlying the case study in Section 5.2.

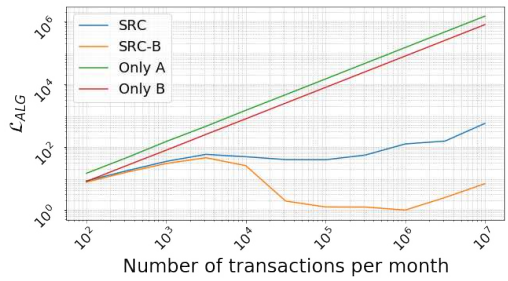


Fig. 8. Performance of 4 policies: SRC, SRC-B, and two naive policies that only route to one provider.

1613–1619.

Agrawal S, Devanur NR (2014) Bandits with concave rewards and convex knapsacks. *Proceedings of the fifteenth ACM conference on Economics and computation*, 989–1006 (ACM).

Agrawal S, Wang Z, Ye Y (2014) A dynamic near-optimal algorithm for online linear programming. *Operations Research* 62(4):876–890.

Ahani N, Gölz P, Procaccia AD, Teytelboym A, Trapp AC (2021) Dynamic placement in refugee resettlement. *arXiv preprint arXiv:2105.14388* .

Alaei S, Hajiaghayi M, Liaghat V (2013) The online stochastic generalized assignment problem. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 11–25 (Springer).

Albers S, Mitzenmacher M (2000) Average-case analyses of first fit and random fit bin packing. *Random Structures & Algorithms* 16(3):240–259.

Alijani R, Banerjee S, Gollapudi S, Munagala K, Wang K (2020) Predict and match: Prophet inequalities with uncertain supply. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 4(1):1–23.

Arlotto A, Gurvich I (2019) Uniformly bounded regret in the multisecretary problem. *Stochastic Systems* .

Arlotto A, Xie X (2020) Logarithmic regret in the dynamic and stochastic knapsack problem with equal rewards. *Stochastic Systems* .

Bansak K, Paulson E (2022) Outcome-driven dynamic refugee assignment with allocation balancing. *Proceedings of the 23rd ACM Conference on Economics and Computation*, 1182–1183.

Berry AC (1941) The accuracy of the gaussian approximation to the sum of independent variates. *Transactions of the american mathematical society* 49(1):122–136.

Bertsekas DP (2011) Dynamic programming and optimal control 3rd edition, volume ii. *Belmont, MA: Athena Scientific* .

Bray RL (2019) Does the multisecretary problem always have bounded regret? *arXiv preprint arXiv:1912.08917* .

Buchbinder N, Naor JS, et al. (2009) The design of competitive online algorithms via a primal–dual approach. *Foundations and Trends® in Theoretical Computer Science* 3(2–3):93–263.

Bumpensanti P, Wang H (2020) A re-solving heuristic with uniformly bounded loss for network revenue management. *Management Science* .

Coffman Jr EG, Courcoubetis C, Garey M, Johnson DS, McGeoch LA, Shor PW, Weber RR, Yannakakis M (1991) Fundamental discrepancies between average-case analyses under discrete and continuous distributions: A bin packing case study. *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, 230–240 (ACM).

Courcoubetis C, Weber R (1986) Necessary and sufficient conditions for stability of a bin-packing system. *Journal of applied probability* 23(4):989–999.

Csirik J, Johnson DS, Kenyon C, Orlin JB, Shor PW, Weber RR (2006) On the sum-of-squares algorithm for bin packing. *Journal of the ACM (JACM)* 53(1):1–65.

Dal Pozzolo A, Caelen O, Johnson RA, Bontempi G (2015) Calibrating probability with undersampling for unbalanced classification. *2015 IEEE Symposium Series on Computational Intelligence*, 159–166 (IEEE).

Devanur NR, Jain K, Sivan B, Wilkens CA (2019) Near optimal online algorithms and fast approximation algorithms for resource allocation problems. *Journal of the ACM (JACM)* 66(1):7.

Düetting P, Feldman M, Kesselheim T, Lucier B (2017) Prophet inequalities made easy: Stochastic optimization by pricing non-stochastic inputs. *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, 540–551 (IEEE).

Elliott J (2020) *The 7 Best Credit Card Processing Companies of 2020*. URL https://www.investopedia.com/best-credit-card-processing-companies-5080522.

Esseen CG (1956) A moment inequality with an application to the central limit theorem. *Scandinavian Actuarial Journal* 1956(2):160–170.

Gaitonde J, Li Y, Light B, Lucier B, Slivkins A (2022) Budget pacing in repeated auctions: Regret and efficiency without convergence. *arXiv preprint arXiv:2205.08674* .

Gupta V, Radovanović A (2020) Interior-point-based online stochastic bin packing. *Operations Research* 68(5):1474–1492.

Hajiaghayi MT, Kleinberg R, Sandholm T (2007) Automated online mechanism design and prophet inequalities. *AAAI*, volume 7, 58–65.

Hoeffding W (1994) Probability inequalities for sums of bounded random variables. *The collected works of Wassily Hoeffding*, 409–426 (Springer).

Jasin S, Kumar S (2012) A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research* 37(2):313–345.

Kleinberg R (2005) A multiple-choice secretary algorithm with applications to online auctions. *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, 630–631 (Society for Industrial and Applied Mathematics).

Kleinberg R, Weinberg SM (2012) Matroid prophet inequalities. *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, 123–136 (ACM).

Liu S, Li X (2021) Online bin packing with known t. *arXiv preprint arXiv:2112.03200* .

Mangasarian OL, Shiau TH (1987) Lipschitz continuity of solutions of linear inequalities, programs and complementarity problems. *SIAM Journal on Control and Optimization* 25(3):583–595.

Rhee WT, Talagrand M (1993) On line bin packing with items of random size. *Mathematics of Operations Research* 18(2):438–445.

Samuel-Cahn E (1996) Optimal stopping with random horizon with application to the full-information best-choice problem with random freeze. *Journal of the American Statistical Association* 91(433):357–364.

Shevtsova I (2011) On the absolute constants in the berry-esseen type inequalities for identically distributed summands. *arXiv preprint arXiv:1111.6554* .

Shor PW (1986) The average-case analysis of some on-line algorithms for bin packing. *Combinatorica* 6(2):179–200.

Shor PW (1991) How to pack better than best fit: tight bounds for average-case online bin packing. *[1991] Proceedings 32nd Annual Symposium of Foundations of Computer Science*, 752–759 (IEEE).

Sinclair SR, Frujeri F, Cheng CA, Swaminathan A (2022) Hindsight learning for mdps with exogenous inputs. *arXiv preprint arXiv:2207.06272* .

Square (2020) *Payments processor's fee*. URL https://squareup.com/us/en/townsquare/credit-card-processing-fees-and-rates.

Sun R, Wang X, Zhou Z (2020) Near-optimal primal-dual algorithms for quantity-based network revenue management. *Available at SSRN 3728397* .

Talluri K, Van Ryzin G (1998) An analysis of bid-price controls for network revenue management. *Management science* 44(11-part-1):1577–1593.

Vera A, Banerjee S (2020) The bayesian prophet: A low-regret framework for online decision making. *Management Science* .

Vera A, Banerjee S, Gurvich I (2019) Online allocation and pricing: Constant regret via bellman inequalities. *arXiv preprint arXiv:1906.06361* .

A OMMITTED PROOFS

A.1 Preliminaries

Proof of Lemma 2.1. The Lemma follows from Hoeffding's Inequality (Hoeffding 1994), which asserts that for independent random variables $X_1, \ldots, X_n$, where $X_i \in \{0,1\} \forall i$, we have

$$\mathbb{P}\left[|\sum_{i=1}^{n} X_i - \mathbb{E}[\sum_{i=1}^{n} X_i]| \geq \psi \right] \leq 2e^{-\frac{2\psi^2}{n}}.$$

With $n = \bar{t}$, X_i denoting the indicator of whether the ith remaining arrival is of type θ, and $\psi = \sqrt{\alpha \bar{t} \log(\bar{t})}$:

$$\mathbb{P}\left[|\bar{N}(t) - \bar{t} p_{\theta|} \geq \sqrt{\alpha \bar{t} \log(\bar{t})} \right] \leq 2e^{-\frac{2\sqrt{\alpha \bar{t} \log(\bar{t})}^2}{\bar{t}}} = 2\bar{t}^{-2\alpha}.$$

The result follows by taking a union bound over all k types θ. □

Proof of Proposition 1. The proof of part (ii) of the proposition heavily relies on the following result from the literature (where the variable names are adapted to our notation):

Proposition 5 (Theorem 2.4 in Mangasarian and Shiau (1987)). *Let the linear program*

$$\min_{\vec{x},\vec{z}} \vec{c}_1 \cdot \vec{x} + \vec{c}_2 \cdot \vec{z} \quad s.t.\ C_1 \begin{bmatrix} \vec{x} \\ \vec{z} \end{bmatrix} \leq \vec{M},\ C_2 \vec{x} = \vec{N}.$$

have non-empty optimal solution sets S^1 and S^2 for right-hand sides $(\vec{M}^1, \vec{N}^1)$ and $(\vec{M}^2, \vec{N}^2)$ respectively. For each $(\vec{x}^1, \vec{z}^1) \in S^1$ there exists $(\vec{x}^2, \vec{z}^2) \in S^2$ such that $|\vec{x}^1 - \vec{x}^2|_\infty \leq v(C_1; C_2) \left\| \begin{bmatrix} N^1 - N^2 \\ M^1 - M^2 \end{bmatrix} \right\|_\infty$ where

$$v(C_1, C_2) = \sup_{\vec{u},\vec{v}} \Big\{ \max\{\|u\|_1, \|v\|_1\} : \|uC_1 + vC_2\|_1 = 1, \textit{rows of } \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} \textit{ corresponding to nonzero elements of } \begin{pmatrix} u \\ v \end{pmatrix} \textit{ are lin. ind.} \Big\}$$

To adopt the result for our purposes, set $\vec{M}^1 = \begin{bmatrix} -\vec{y} \\ \vec{B}|\vec{N}^1|_1 \end{bmatrix}$, $\vec{M}^2 = \begin{bmatrix} -\vec{y} \\ \vec{B}|\vec{N}^2|_1 \end{bmatrix}$, and let C_1 and C_2 be such that $C_1 \begin{bmatrix} \vec{x} \\ \vec{z} \end{bmatrix} \leq \vec{M}^1$ encodes the constraints $-x_{\theta j} \leq -y_{\theta j}$, and $A\vec{x} - Dz \leq \vec{B}|\vec{N}^1|_1$ and C_2 encodes the constraints $\sum_j x_{\theta j} = N^1_\theta$, respectively $A\vec{x} - Dz \leq \vec{B}|\vec{N}^2|_1$ and $\sum_j x_{\theta j} = N^2_\theta$ for $(\vec{M}^2, \vec{N}^2)$. Then the proposition says that there exists δ, dependent only on C_1 and C_2 (which are independent of T or B) such that for any $(\vec{x}^1, \vec{z}^1)$ optimal for $\vec{N}^1$ we have some $(\vec{x}^2, \vec{z}^2)$ optimal for $\vec{N}^2$ where

$$\left\| \begin{bmatrix} \vec{x}^1 - \vec{x}^2 \\ \vec{z}^1 - \vec{z}^2 \end{bmatrix} \right\|_\infty \leq \delta \left| \begin{bmatrix} \vec{N}_1 - \vec{N}_2 \\ \vec{M}_1 - \vec{M}_2 \end{bmatrix} \right|_\infty = \delta \left\| \begin{bmatrix} \vec{N}_1 - \vec{N}_2 \\ 0 \end{bmatrix} \right\|_\infty, \text{ so in particular } |\vec{x}_1 - \vec{x}_2|_\infty \leq \delta |\vec{N}_1 - \vec{N}_2|_\infty.$$

The proof of part (i) follows as the inner objective is Lipschitz in $\vec{x}$ and $\vec{z}$, and the corresponding optimal solutions $\vec{x}^1, \vec{x}^2$ and $\vec{z}^1, \vec{z}^2$ are Lipschitz in $\vec{N}$ (from part ii), so $f_{\vec{N}}(\vec{x})$ is Lipschitz in $\vec{x}$. □

A.2 Proofs of Lemmas in Section 3

Proof of Lemma 3.1. Observe from $\texttt{OPT}^{\,t}$ that the optimization problems $\texttt{OPT}^{t_a}$ and $\texttt{OPT}^{t_b}$ vary only with respect to the constraints, where $\vec{x}^{\texttt{ALG}}(t_a - 1)$ is replaced by $\vec{x}^{\texttt{ALG}}(t_b - 1)$. By the lemma assumption, the solution to $\texttt{OPT}^{t_a}$ fulfills that $x^{\texttt{ALG}}_{\theta j}(t_b - 1) \leq \texttt{OPT}^{t_a}_{\theta j}$. Thus, it is feasible to $\texttt{OPT}^{t_b}$, implying that the objective of $\texttt{OPT}^{t_b} \geq \texttt{OPT}^{t_a}$. Equality follows as $\texttt{OPT}^{t_b}$ is more constrained than $\texttt{OPT}^{t_a}$. □

Proof of Lemma 3.2. Observe first that, by Proposition 1, we have

$$\max_{\theta,j} |\texttt{OPT}_{\theta j} - T \times \texttt{DLP}_{\theta j}| \leq \delta \max_\theta |Tp_\theta - N_\theta|, \qquad \text{implying that } \forall\, \theta, j : \texttt{OPT}_{\theta j} \geq T \times \texttt{DLP}_{\theta j} - \delta \max_\theta |Tp_\theta - N_\theta|.$$

Setting $\alpha = \frac{(T \log(T))^\epsilon}{\delta^2}$ and $t = 0$, i.e., $\bar{t} = T$, in Lemma 2.1 we find with probability greater $1 - 2kT^{-\frac{T^\epsilon}{\delta^2}}$ that the expression $\delta \max_\theta |Tp_\theta - N_\theta|$ is at most $[T \log(T)]^{\frac{1+\epsilon}{2}}$, which implies the result. □

Proof of Lemma 3.3. Observe that the first t arrivals, $N_\theta(t)$ and the last t arrivals, $\bar{N}_\theta(t)$ come from the same distribution; thus, Lemma 2.1 gives $\mathbb{P}\left[\max_\theta\{|N_\theta(t) - tp_\theta|\} > \sqrt{\alpha t \log(t)}\right] \leq 2kt^{-2\alpha}$. With $\alpha = \frac{(t \log(t))^\epsilon}{\delta^2}$, we get

$$\mathbb{P}\left[\mathcal{S}_t\right] \leq \mathbb{P}\left[\max_\theta\{|N_\theta(t) - tp_\theta|\} > \frac{[t \log(t)]^{\frac{1+\epsilon}{2}}}{\delta}\right] \leq 2kt^{-2\frac{(t\log(t))^\epsilon}{\delta^2}} < 2kt^{\frac{-2t^\epsilon}{\delta^2}}. \quad \square$$

Proof of Lemma 3.4. We distinguish between periods t where the bad event $\mathcal{S}_t$ holds true, and ones where it does not. The former we charge to the right-hand side; this leaves us with showing that across the periods in which $\mathcal{S}_t$ does not hold true, with probability at least $1 - \frac{1}{T}$, we do not use any action $a_{\theta j}$ more than $\texttt{OPT}_{\theta j}$ times. Observe that when $\mathcal{S}_t$ does not hold true

- the probability that we take an action $a_{\theta j}$ where $\texttt{DLP}_{\theta j} = 0$ is equal to 0, and
- the probability that we take an action $a_{\theta j}$ where $\texttt{DLP}_{\theta j} > 0$ is at most $\texttt{DLP}_{\theta j} + \ell \frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}}$.

The former is thus guaranteed to not be used more than $\texttt{OPT}_{\theta j}$ times; the latter is stochastically dominated by $\sum_{t=1}^{\hat{T}} Ber\left(\texttt{DLP}_{\theta j} + \ell \frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}}\right)$. We want to show that this is less than $\texttt{OPT}_{\theta j}$, with sufficiently high probability, to prove the lemma. Recall first (Lemma 3.2) that $\texttt{OPT}_{\theta j} \geq T \times \texttt{DLP}_{\theta j} -$

$[T\log(T)]^{\frac{1+\epsilon}{2}}$ with probability at least $1-2kT^{\frac{-T^\epsilon}{\delta^2}}$. On the other hand, we have

$$\mu_{\theta j} := \mathbb{E}\left[\sum_{t=1}^{\hat{T}} Ber(\mathtt{DLP}_{\theta j}+\ell\frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}})\right] \le \hat{T}\times\mathtt{DLP}_{\theta j}+\ell\log(T)\int_1^{\hat{T}}\frac{1}{t^{\frac{1-\epsilon}{2}}}dt = \hat{T}\mathtt{DLP}_{\theta j}+\ell\log(T)\frac{2\hat{T}^{\frac{1+\epsilon}{2}}}{\epsilon+1}.$$

Now, we can bound for an action $a_{\theta j}$ with $\mathtt{DLP}_{\theta j}>0$

$$\mathbb{P}\left[\sum_{t=1}^{\hat{T}} Ber(\mathtt{DLP}_{\theta j}+\ell\frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}}) > \mathtt{OPT}_{\theta j}\right]$$

$$\le 2kT^{\frac{-T^\epsilon}{\delta^2}} + \mathbb{P}\left[\sum_{t=1}^{\hat{T}} Ber(\mathtt{DLP}_{\theta j}+\ell\frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}}) > T\times\mathtt{DLP}_{\theta j}-[T\log(T)]^{\frac{1+\epsilon}{2}}\right].$$

Suppose next that

$$T\times\mathtt{DLP}_{\theta j}-[T\log(T)]^{\frac{1+\epsilon}{2}} > \hat{T}\times\mathtt{DLP}_{\theta j}+\ell\log(T)\frac{2\hat{T}^{\frac{1+\epsilon}{2}}}{\epsilon+1}+[T\log(T)]^{\frac{1+\epsilon}{2}}$$

$$\text{or equivalently } (T-\hat{T})\times\mathtt{DLP}_{\theta j} \ge \ell\log(T)\frac{2\hat{T}^{\frac{1+\epsilon}{2}}}{\epsilon+1}+2\,[T\log(T)]^{\frac{1+\epsilon}{2}}.$$

Observe that the left-hand side grows as $\Omega(T^{\frac{1}{2}+\epsilon})$, whereas the right-hand side grows as $\tilde{O}(T^{\frac{1+\epsilon}{2}})$, so for large enough T this holds true. We set T_0 to be the smallest constant such that this holds for $T\ge T_0$ for every θ and j with $\mathtt{DLP}_{\theta j}>0$. Then, we can bound the above probability as

$$2kT^{\frac{-T^\epsilon}{\delta^2}} + \mathbb{P}\left[\sum_{t=1}^{\hat{T}} Ber(\mathtt{DLP}_{\theta j}+\ell\frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}}) > \hat{T}\times\mathtt{DLP}_{\theta j}+\ell\log(T)\frac{2\hat{T}^{\frac{1+\epsilon}{2}}}{\epsilon+1}+[T\log(T)]^{\frac{1+\epsilon}{2}}\right]$$

$$\le 2kT^{\frac{-T^\epsilon}{\delta^2}} + \mathbb{P}\left[\sum_{t=1}^{\hat{T}} Ber(\mathtt{DLP}_{\theta j}+\ell\frac{[\log(t)]^{\frac{1+\epsilon}{2}}}{t^{\frac{1-\epsilon}{2}}}) - \mu_{ij} > [T\log(T)]^{\frac{1+\epsilon}{2}}\right]$$

$$\le 2kT^{\frac{-T^\epsilon}{\delta^2}} + e^{-2\frac{(T\log(T))^{1+\epsilon}}{\hat{T}}} \le 2kT^{\frac{-T^\epsilon}{\delta^2}} + T^{-2},$$

where the last line follows from Hoeffding's Inequality (see Proof of Lemma 2.1). Taking a union bound over all $a_{\theta j}$ with $\mathtt{DLP}_{\theta j}>0$ — noting that we only need the bound from Lemma 3.2 once — we obtain a bound of $2kT^{\frac{-T^\epsilon}{\delta^2}}+k\ell T^{-2} < \frac{1}{T}$ for $T > \max\{2\delta^{\frac{2}{\epsilon}}, 2k\ell, T_0\}$. The result follows. □

Proof of Lemma 3.5. We argue similarly to the proof of Lemma 3.2: by Proposition 1, in period t,

$$\max_{\theta,j}|\mathtt{OPT}^t_{\theta j}-\mathtt{TDLP}_{\theta j}(t)| \le \delta\max_\theta|\frac{\bar{t}}{t}N_\theta(t)-\bar{N}_\theta(t)|,$$

$$\text{implying } \forall\,\theta,j:\mathtt{OPT}^t_{\theta j} \ge \mathtt{TDLP}_{\theta j}(t)-\delta\max_\theta|\frac{\bar{t}}{t}N_\theta(t)-\bar{N}_\theta(t)|.$$

Then, by triangle inequality, $\delta\max_\theta|\frac{\bar{t}}{t}N_\theta(t)-\bar{N}_\theta(t)| \le \delta\left(\max_\theta|\frac{\bar{t}}{t}N_\theta(t)-p_\theta\bar{t}|+\max_\theta|p_\theta\bar{t}-\bar{N}_\theta(t)|\right)$, and we find, with $\alpha=\frac{\sqrt{\bar{t}}}{\delta^2}$ in Lemma 2.1, that the probability of either term being greater $\frac{\bar{t}^{\frac{3}{4}}\sqrt{\log(\bar{t})}}{\delta}$ is at most $4k\bar{t}^{-2\sqrt{\bar{t}}/\delta^2}$, where we are using that $t>\bar{t}$ which follows from $\hat{T}>T/2$. This implies the result, i.e., we have with probability at least $1-4k\bar{t}^{-2\sqrt{\bar{t}}/\delta^2}$ that $\forall\,\theta,j:\mathtt{OPT}^t_{\theta j}\ge\mathtt{TDLP}_{\theta j}(t)-2\bar{t}^{\frac{3}{4}}\sqrt{\log(\bar{t})}$. □

A.3 Proof of Lemma 4.1

As stated before, the lemma follows from the Berry-Esseen Theorem (Berry 1941), of which we use the following formulation.

Proposition 6 (Corollary 3 in Shevtsova (2011)). *Let $X_1, \dots, X_\tau$ be i.i.d. random variables drawn from a distribution fulfilling $\mathbb{E}[X_i] = 0$, $\sigma^2 = Var[X_i] = E[X_i^2] = 1$, and finite third absolute moment β_3. Let $S_\tau = \sum_i X_i$, $Z \in N(0,1)$, and $\phi(r)$ be the CDF of Z. Then*

$$\Delta_\tau = \sup_r \left|\phi(r) - \mathbb{P}\left[S_\tau < r\sqrt{\tau}\right]\right| \leq 0.4748\beta_3/\sqrt{\tau}$$

holds for all n.

Define the Bernoulli random variables $X_1, \dots, X_\tau$ with

$$X_i = \begin{cases} \sqrt{\frac{1-p_\theta}{p_\theta}} & \text{with probability } p_\theta \\ -\sqrt{\frac{p_\theta}{1-p_\theta}} & \text{with probability } 1-p_\theta \end{cases}$$

Observe that $\mathbb{E}[X_i] = 0$, $\sigma^2 = 1 - p_\theta + p_\theta = 1$, and $\beta_3 = \mathbb{E}[|X^3|] \leq \frac{1}{\sqrt{p_\theta}} + \frac{1}{\sqrt{1-p_\theta}}$. Then, with S_n, Z as in Proposition 6 we have

$$\sup_r |\phi(r) - \mathbb{P}[S_\tau < r\sqrt{\tau}]| \leq \left(\frac{1}{\sqrt{p_\theta}} + \frac{1}{\sqrt{1-p_\theta}}\right)\frac{0.4748}{\sqrt{\tau}}.$$

For fixed $r > 0$, we have $1 > \phi(r) > 0$, so for large enough τ there exists $\bar{\xi}_r \in (0,1)$ with

$$\bar{\xi}_r := \phi(r) - \left(\frac{1}{\sqrt{p_\theta}} + \frac{1}{\sqrt{1-p_\theta}}\right)\frac{0.4748}{\sqrt{\tau}} \leq \mathbb{P}[S_\tau < r\sqrt{\tau}].$$

Setting $\xi_r = 1 - \bar{\xi}_r$ we find that $\mathbb{P}[S_\tau \geq r\sqrt{\tau}] \geq \xi_r > 0$. Now, define $Y_i = X_i\sqrt{p_\theta(1-p_\theta)} + p_\theta$, to obtain a random variable that is 1 with probability p_θ and 0 otherwise. Further, we have

$$N_\theta \stackrel{d}{=} \sum_i Y_i = \sum_i X_i\sqrt{p_\theta(1-p_\theta)} + p_\theta = np_\theta + \sqrt{p_\theta(1-p_\theta)}S_\tau$$

With probability at least $\frac{\xi_r}{\sqrt{p_\theta(1-p_\theta)}}$ we have $S_\tau \geq r\sqrt{\tau}$, and thus, with probability at least $\frac{\xi_m}{\sqrt{p_\theta(1-p_\theta)}}$, we have $N_\theta \geq np_\theta + \sqrt{p_\theta(1-p_\theta)}$. □

A.4 Proofs of further impossibility results

Proof of Proposition 3. Consider a bin packing instance as in Proposition 2 in which bins have size 3 and items are of size either 1 or 2 with probability $\frac{1}{2}$; the time horizon T is a geometric random variable with mean τ. As in Proposition 2, all arriving items of size 2 need to be put into a bin of configuration $\{1,2\}$ whereas items of size 1 can either be in a $\{1,2\}$ configuration or in a $\{1,1,1\}$ configuration. The main observation necessary to derive the result is that the optimal Bellman equation gives a threshold solution that sets a threshold d_τ and puts arriving items of size 1 into bins of configuration $\{1,1,1\}$ when there are d_τ bins with only one 1 in them (such bins can have an arriving 2 added to them), i.e., it acts as follows:

- when an item of size 2 arrives, and there is a bin with only one 1 in it, add the size 2 item to that bin; else, add the arriving size 2 item to a new bin;
- when an item of size 1 arrives, and there is a bin with only one 2 in it, add the arriving size 1 item to that bin; if there is no such bin, but there is a bin that has two size 1 items in it, add the arriving size 1 item to that one; if there is no such bin either, and there are less than d_τ

bins that have just a single item of size 1 in them, add the arriving size 1 item to a new bin; else, add the arriving size 1 item to an existing bin that has a single 1 in it.

We distinguish between the case that $d_\tau \le \sqrt{\tau}$ and $d_\tau > \sqrt{\tau}$. Further, we condition on the time horizon lasting between τ and 2τ periods, which occurs with constant probability (as τ grows large). Then, the following events each occur with constant probability:

(i) over the entire horizon the number of items of size 1 is $2\sqrt{\tau}$ greater than the number of items of size 2;
(ii) over the first $\tau/2$ periods there are at least $2\sqrt{\tau}$ more arrivals of size 1 than of size 2 but over the entire horizon there are more arrivals of size 2 than of size 1.

With $d_\tau \le \sqrt{\tau}$, under the first event, at the end of the horizon there are $\Omega(\sqrt{\tau})$ items of size 1 in bins by themselves; these could be in a third as many bins of configuration $\{1,1,1\}$; with $d_\tau > \sqrt{\tau}$, under the second event, the algorithm creates $\Omega(\sqrt{\tau})$ bins of configuration $\{1,1,1\}$ in the first $\tau/2$ periods, whereas the optimal solution requires none of these; thus, for any threshold d_τ the expected loss is $\Omega(\sqrt{\tau})$. □

Proof of Proposition 4. We consider the following events:

- E_1 is the event that there are at least $\sqrt{T}/2$ items of value 2 in both the first and the last $T/2$ periods;
- E_2 is the event that there are at most $T/4$ items of value 3 over the first $T/2$ periods;
- E_3 is the event that there are at least $T/2$ items of value 3 over the entire horizon;
- E_4 is the event that there are at most $T/2 - \sqrt{T}$ items of value 3 over the entire horizon.

Observe that both $E_1 \cap E_2 \cap E_3$ and $E_1 \cap E_2 \cap E_4$ occur with constant probability by Lemma 4.1; now, consider the number of items of value 2 accepted by an algorithm in the first $T/2$ periods: if it is at least $\sqrt{T}/4$, then the algorithm incurs $\Omega(\sqrt{T})$ loss under $E_1 \cap E_2 \cap E_3$, and if it is at most $\sqrt{T}/4$ then the algorithm incurs $\Omega(\sqrt{T})$ loss under $E_1 \cap E_2 \cap E_4$. Thus, in this instance any algorithm must incur $\Omega(\sqrt{T})$ loss. □

B APPLICATION TO ONLINE PACKING PROBLEMS

In this section we describe how our framework captures online packing problems or variants thereof, including AdWords (Alaei et al. 2013), Network Revenue Management (Talluri and Van Ryzin 1998), or refugee resettlement (Bansak and Paulson 2022). In these problems, for a fixed time horizon T, we have a budget of resources $\vec{B}$. Upon each arrival of type θ we need to make an irrevocable assignment decision, wherein an arrival is assigned to a product j, consumes some resources $A_{\theta j}$ for each resource type j, and yields some value $c^1_{\theta j}$ — note that the latter two are dependent on both the arrival type and the product assignment; to fit the minimization objective description in Section 2 we will minimize $-\vec{c}^1 \cdot \vec{x}$, which maximizes the value. In the canonical quantity-based network revenue management problem, the arrival is a specific request and the decision is just *accept*/*reject*; we can model the *reject* decision by allowing for a product with infinite resources and no value. In refugee resettlement there is no *reject* decision as arriving refugees need to be send to a location within a host country; the value to the optimization problem comes from the probability they find employment, as captured through an exogenous machine learning model. As a relaxation of these problems we allow $f_{\vec{N}}$ to violate the budget constraint at a penalty which is greater than the largest value; when T is known from the beginning our algorithm would never violate the budget/incur that penalty; when T is a priori unknown our algorithm may, with vanishingly small probability under Assumption 3, violate the budget and incur the penalty. We have z_j as a decision variable capture by how much the resource constraint on resource j has been violated, and have a penalty of $c^2_j = \max\{c_{\theta j} + 1\}$ Then, we can write the objective as follows

$$\begin{aligned} f_{\vec{N}}(\vec{x}) \quad = \quad & \min_{\vec{z}} -\vec{c}^{\,1} \cdot \vec{x} + \vec{c}_2 \cdot \vec{z} \\ & \text{s.t.} \quad \sum_j x_{\theta j} = N_\theta \; \forall\, \theta \\ & A\vec{x} - I\vec{z} \le \vec{B} \sum_\theta N_\theta \\ & z_{\theta j} \ge 0 \;\; \forall\, \theta, j, \end{aligned}$$

Observe that for any fixed $\vec{x}$ the first term in the objective captures the *negative of the value of the product assignments* (which we aim to minimize); whereas the optimal $z_j = \min\{0, B_j - \sum_\theta A_{\theta j} x_{\theta j}\}$ captures by how much the resource constraint on resource j is violated, and penalizes it by a number greater than the value of any resource assignment.

C KNOWN HORIZON AND LAZY RESOLVING

In this appendix we consider different assumptions on the arrival process, the information structure on the length of the time-horizon, and the objective function. Specifically, we focus on a setting where the objective function (not necessarily linear) fulfills properties (i) and (ii) in Proposition 1, the time horizon T is known from the beginning, we have one sample path (a *trace*) of T arrivals, and the arrivals come from a distribution that fulfills the following two assumptions.

Assumption 4 (Anytime Concentration). *For each type $\theta \in \Theta$ and any period t, we assume that the number of future arrivals $\bar{N}_\theta(t)$ is concentrated about its mean; formally, $\exists$ constant $\alpha > 0$, independent of t, such that $\forall\, \theta, t$ and every $x > 0$*

$$\mathbb{P}\left[\left|\bar{N}_\theta(t) - \mathbb{E}[\bar{N}_\theta(t)]\right| > x\right] \le 2e^{\frac{-\alpha x^2}{t}}.$$

Assumption 5 (Anytime Thickness). *For each type $\theta \in \Theta$ and any period t, we assume that the expected number of future arrivals $\bar{N}_\theta(t)$ grows faster than $\sqrt{t}$; formally, $\exists$ positive constants β, γ independent of t such that $\forall\, \theta, t$ and every $x > 0$*

$$\mathbb{E}\left[\bar{N}_\theta(t)\right] \ge \beta \bar{t}^{0.5+\gamma}.$$

Observe that Assumption 1, with $\gamma = 1/2$, is a special case of Assumption 5. Proposition 7 shows that Assumption 5 is tight in the sense that one cannot guarantee an expected loss of $o(\sqrt{T})$ when $\gamma = 0$. Also, note that Lemma 2.1 (implied by Assumption 1) is a variant of Assumption 4.

ALGORITHM 2: Lazy resolving

– Initialize $x^{\mathsf{ALG}}_{\theta j}(0) = 0; F_{\theta j} = 0 \;\; \forall\, \theta, j$

for $t = 1, \ldots, T$ **do**

  if $F_{\theta_t j} < 1$ *for every action* j **then**

    – Solve optimization problem $\mathsf{SDLP}(t)$;

    – Set $F_{\theta j} = \left\lfloor \mathsf{SDLP}_{\theta j}(t) - x^{\mathsf{ALG}}_{\theta j}(t) - 2\bar{t}^{\left(\frac{1+\gamma}{2}\right)} \sqrt{\log \bar{t}} \right\rfloor \;\; \forall\, \theta, j$

  end

  – In period t, use action $a_{\theta_t j}$ for some $j \in \arg\max_j \{F_{\theta_t j}\}$; reduce $F_{\theta_t j}$ by 1;

  – Set $x^{\mathsf{ALG}}(t)$ by incrementing $x^{\mathsf{ALG}}_{\theta_t j}(t-1)$ by 1 and leaving all other values unchanged;

end

We denote the sample path we have access to by $\hat{\theta}_1, \dots, \hat{\theta}_T$, write $\widetilde{N}^f_\theta(t)$ for the number of type θ arrivals in periods $t+1, \dots, T$ on that sample path, and $\widetilde{N}^f$ for the vector of those arrivals over the entire sample path; it is worth noting that having such a sample path is a significantly weaker assumption than full distributional information, as the later always allows one to draw one sample. Relative to the i.i.d. setting in the main body, this setting here allows for limited correlations between arrivals in different periods. Similar to Algorithm 1 solving TDLP once the time horizon is known, Algorithm 2 relies on solving the following:

$$\min_{\vec{x}} f_{\widetilde{N}^f}(\vec{x}) \qquad \text{s.t.} \qquad \vec{x} \geq \vec{x}^{\mathsf{ALG}}(t-1) \;\; \forall\, \theta, j. \tag{SDLP(t)}$$

Again, similar to Algorithm 1, Algorithm 2 has the following property.

Theorem C.1. *Suppose the arrivals fulfill Assumption 4 and 5 and the objective fulfills properties (i) and (ii) of Proposition 1; then there exists an algorithm* ALG *and a constant $M_2(\beta, \lambda, \delta, \epsilon, \ell)$, independent of T, such that the loss of Algorithm 2 can be bounded as $\mathcal{L}_{\mathsf{ALG}} \leq M_2(\beta, \delta, \lambda, \epsilon, \ell)$.*

Observe that Theorem C.1 does not rely on Assumption 2. Thus, we do not require a solution $\mathsf{OPT}^{\,t}$ to be unique below.[7] The proof of Theorem C.1 requires us to show the following result, similar to Lemma 3.5.

Lemma C.2. *Consider a period t with actions $x^{\mathsf{ALG}}_{\theta j}(t-1)$ already taken in periods $[1, t-1]$; with probability at least $1-4\bar{t}^{-\frac{\alpha \bar{t}^{\,\gamma}}{\delta^2}}$ there exists some solution $\mathsf{OPT}^{\,t}$ with $\mathsf{OPT}^{\,t}_{\theta j} \geq \left\lfloor \mathsf{SDLP}_{\theta j}(t) - 2\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}} \right\rfloor \forall \theta, j.$*

Proof sketch of Theorem C.1. Denote, as in the proof of Theorem 2.2, by $T_1 = \hat{T}, T_2, \dots, T_s$ the periods in which Algorithm 2 solves $\mathsf{SDLP}(t)$, by T_η the first period in which Algorithm 2 solves $\mathsf{TDLP}(t)$ and it is the case that for some θ all budgets $F_{\theta j} < 1$, and let $\mathcal{T}_i$ be the event that the LCB created in period T_i fails to hold true for any optimal solution $\mathsf{OPT}^{\,T_i}$:

$$\mathcal{T}_i = \left\{ \forall\, \mathsf{OPT}^{\,T_i} \; \exists \theta, j : \mathsf{OPT}^{\,T_i}_{\theta j} < \left\lfloor \mathsf{TDLP}_{\theta j}(T_i) - 2\bar{T}_i^{\frac{3}{4}}\sqrt{\log \bar{T}_i} \right\rfloor \right\}.$$

Then we can bound

$$\mathcal{L}_{\mathsf{ALG}} = \mathbb{E}\left[\mathsf{OPT}^{\,T} - \mathsf{OPT}^{\,1}\right] \leq 2\lambda \mathbb{E}\left[\sum_{i=1}^{\eta-1}(T_{i+1} - T_i)\mathbb{P}[\mathcal{T}_i] + \bar{T}_\eta\right] \leq 2\lambda \mathbb{E}\left[\sum_{i=1}^{\eta-1}\bar{T}_i\mathbb{P}[\mathcal{T}_i] + \bar{T}_\eta\right].$$

We bound the sum similarly to how we did in the proof of Theorem 2.2 (using Lemma C.2 instead of Lemma 3.5, where with $\gamma = 1/2$ it is the same bound).

For $\mathbb{E}\left[\bar{T}_\eta\right]$ we construct an upper bound by noting that in period t, SDLP assigns $\widetilde{N}^f_\theta(t)$ actions for type θ, meaning at least one action is assigned at least $\widetilde{N}^f_\theta(t)/\ell$ actions; with high probability (by Assumption 4) this is at least $\frac{\mathbb{E}[\tilde{N}_\theta(t)]}{2\ell} \geq \beta \bar{t}^{\,.5+\gamma}$ which, for $\bar{t}$ large enough is greater $2\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}}$. □

[7] Recall that for functions $f_{\widetilde{N}}$ as described in Section 2 this was without loss of generality, but for the more general class of functions we discuss here this need not hold true.

Proof of Lemma C.2. Consider the solution $\texttt{OPT}^{\,t}$ to $\texttt{SDLP}(t)$ that minimizes $\max_{\theta,j} |\texttt{OPT}^{t}_{\theta j} - \texttt{SDLP}_{\theta j}(t)|$, where $\texttt{SDLP}(t)$ is the solution Algorithm 2 finds when solving in period t. Then,

$$\begin{aligned}
\mathbb{P}\left[\exists \theta, j : \texttt{OPT}^{t}_{\theta j} < \lfloor \texttt{SDLP}_{\theta j}(t) - 2\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}}\rfloor\right] \leq \\
\mathbb{P}\left[\max_{\theta,j} |\texttt{OPT}^{t}_{\theta j} - \texttt{SDLP}_{\theta j}(t)| \geq 2\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}}\right] \leq \\
\mathbb{P}\left[\max_{\theta} |\bar{N}_\theta(t) - \widetilde{N}^f_\theta(t)| \geq \frac{2\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}}}{\delta}\right] \leq \\
k \max_{\theta} \mathbb{P}\left[|\bar{N}_\theta(t) - \widetilde{N}^f_\theta(t)| \geq \frac{2\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}}}{\delta}\right] \leq \\
k \max_{\theta}\left(\mathbb{P}\left[|\bar{N}_\theta(t) - \mathbb{E}[\bar{N}_\theta(t)]| \geq \frac{\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}}}{\delta}\right] + \mathbb{P}\left[|\mathbb{E}[\bar{N}_\theta(t)] - \widetilde{N}^f_\theta(t)| \geq \frac{\bar{t}^{\left(\frac{1+\gamma}{2}\right)}\sqrt{\log \bar{t}}}{\delta}\right]\right) \leq \\
4e^{-\frac{\alpha \bar{t}^{\,\gamma}\log(\bar{t})}{\delta^2}} = 4\bar{t}^{-\frac{\alpha \bar{t}^{\,\gamma}}{\delta^2}} \qquad \square
\end{aligned}$$

Lazy resolving. In this part we argue that Algorithm 2 has not only strong performance guarantees, but also allows for significant lesser resolving; in particular, we claim that with i.i.d. arrivals it only needs to resolve SDLP, in expectation, $O(\log\log(T))$ many times. To see this, note that after resolving in period t, for large enough $\bar{t}$, we have $\bar{t} - o(\bar{t})$ periods to go before we next need to resolve. Indeed, it next resolves after observing, for some type θ, at least $\widetilde{N}^f_\theta(t) - \ell 2\bar{t}^{\frac{3}{4}}\sqrt{\log \bar{t}}$ arrivals. Now, with i.i.d. arrivals we find that $\widetilde{N}^f_\theta(t)$ is close to, or greater, $\bar{t}p_\theta$ with high probability (in $\bar{t}$, by Assumption 4 or Lemma 2.1); thus, θ requires the algorithm to resolve next after observing about $\bar{t}p_\theta - o(\bar{t})$ more arrivals of type θ; however, before observing $\bar{t}p_\theta - o(\bar{t})$ arrivals of type θ, with high probability, we observe $\bar{t}p_{\theta'} - o(\bar{t})$ of each other type θ'; thus, only $o(\bar{t})$ periods remain when the algorithm next resolves.

Necessity of Assumption 5. In this section we prove that Assumption 5 is tight in the sense that with $\gamma = 0$ one may not be able to obtain the general uniform loss guarantees of Theorem C.1.

Proposition 7. *There exists a family of instances wherein $\mathbb{E}[\bar{N}_\theta(t)] \geq \beta \bar{t}^{0.5}$ holds true $\forall t$, yet no online algorithm can obtain uniform loss to the online knapsack problem.*

Proof. Consider a time horizon of even length T and a budget $T/2$. Items have unit size and value 1, 2, or 3. For the first $T/2$ periods, all arriving items have value 2. Then, there are $T/2 - \sqrt{T}$ arrivals of value either 1 or 3 with probability $\frac{1}{2}$ each. Finally, the last $\sqrt{T}$ arrivals have value 1, 2, or 3 with probability $\frac{1}{3}$ each. Observe that with this family of instances we have $\mathbb{E}[\bar{N}_\theta(t)] \geq \frac{\sqrt{\bar{t}}}{3}$ since $\mathbb{E}[\bar{N}_\theta(t)] = \frac{\bar{t}}{3}$ for $t \geq T - \sqrt{T}$, and $\mathbb{E}[\bar{N}_\theta(t)] \geq \frac{\sqrt{T}}{3} \geq \frac{\sqrt{\bar{t}}}{3}$ for $t < T - \sqrt{T}$, so these instances fulfill the assumption of the proposition with $\beta = 1/3$. Further, by Lemma 4.1 there exists $\xi > 0$ such that

$$\mathbb{P}[\bar{N}_2(T/2) + \bar{N}_3(T/2)] + \leq \frac{T}{4} - \sqrt{T}] > \xi \text{ and } \mathbb{P}[\bar{N}_3(T/2) \geq \frac{T}{4} + \sqrt{T}] > \xi.$$

Now, consider an algorithm ALG that accepts $x_2^{\texttt{ALG}}(\frac{T}{2})$ items of value 2 in the first $\frac{T}{2}$ periods. We distinguish between two cases.

Case $x_2^{\mathsf{ALG}}(\frac{T}{2}) \geq \frac{T}{4}$. With probability at least ξ we have $\bar{N}_3(T/2) \geq \frac{T}{4} + \sqrt{T}$, i.e., there are at least $\frac{T}{4} + \sqrt{T}$ arrivals of value 3 in the second half of the time horizon, of which ALG can only accept $\frac{T}{4}$ yielding an expected loss, relative to the clairvoyant, of at least $\xi\sqrt{T} \in \Omega(\sqrt{T})$.

Case $x_2^{\mathsf{ALG}}(\frac{T}{2}) < \frac{T}{4}$ With probability at least ξ we have $\bar{N}_2(T/2) + \bar{N}_3(T/2) + \leq \frac{T}{4} - \sqrt{T}$, i.e., there are at most $\frac{T}{4} - \sqrt{T}$ arrivals of value either 2 or 3 in the remaining periods. Thus, ALG must accept at least $\sqrt{T}$ items of value 1 or leave $\sqrt{T}$ of its budget unused, yielding an expected loss, relative to the clairvoyant, of at least $\xi\sqrt{T} \in \Omega(\sqrt{T})$. □