Online Fair Allocation of Perishable Resources

Siddhartha Banerjee[1], Chamsi Hssaine[2], and Sean R. Sinclair[3]

[1]School of Operations Research and Information Engineering, Cornell University
[2]Marshall School of Business, University of Southern California
[3]Laboratory for Information and Decision Sciences, Massachusetts Institute of Technology

Abstract

We consider a practically motivated variant of the canonical online fair allocation problem: a decision-maker has a budget of *perishable* resources to allocate over a fixed number of rounds. Each round sees a random number of arrivals, and the decision-maker must commit to an allocation for these individuals before moving on to the next round. The goal is to construct a sequence of allocations that is *envy-free* and *efficient*. Our work makes two important contributions toward this problem: we first derive strong lower bounds on the optimal envy-efficiency trade-off that demonstrate that a decision-maker is fundamentally limited in what she can hope to achieve relative to the no-perishing setting; we then design an algorithm achieving these lower bounds which takes as input (*i*) a prediction of the perishing order, and (*ii*) a desired bound on envy. Given the remaining budget in each period, the algorithm uses forecasts of future demand and perishing to adaptively choose one of two carefully constructed guardrail quantities. We demonstrate our algorithm's strong numerical performance — and state-of-the-art, perishing-agnostic algorithms' inefficacy — on simulations calibrated to a real-world dataset.

*A preliminary version of this paper appeared as an extended abstract in ACM SIGMETRICS 2023.

Contents

1 Introduction

Despite a consistent decline in food insecurity in the United States over the past decade, 2022 saw a marked increase in individuals struggling to access enough food to fulfill basic needs. A recent report by the U.S. Department of Agriculture found that over 44 million individuals faced some form of hunger in 2022 — 45% more than the previous year (Godoy, 2023). Due in part to rising food prices and the rolling back of pandemic-era social security measures, this disturbing statistic has further underscored the important role of local food banks; for instance, the Feeding America network of food banks, food agencies, and local food programs distributed over 5.2 billion meals that same year (Feeding America, 2022).

In distributing food throughout their operating horizon, food banks have two competing objectives: distributing *as much* food as possible to communities in need, and ensuring *equitable* access to donations. This tension has attracted much attention in the operations literature, with recent work characterizing the fundamental trade-offs between fairness and overall utility in sequential allocation problems (Bertsimas et al., 2011; Donahue and Kleinberg, 2020; Lien et al., 2014; Manshadi et al., 2023; Sinclair et al., 2022). Understanding such trade-offs in theory is useful, as they allow a system designer to recognize and choose their desired operating point, balancing the loss in efficiency and equity. Despite the useful insights derived from prior work, to the best of our knowledge an important reality of food bank operations remains overlooked: the existence of *perishable* goods, which constitute a substantial portion of food donations. The Los Angeles Regional Food Bank, for instance, distributed over 26 million pounds of produce in 2022 alone (Los Angeles Regional Food Bank, 2022a). Perishable goods pose significant challenges for these organizations, who frequently find themselves needing to throw out spoiled goods (Los Angeles Regional Food Bank, 2022b). Indeed, the equity-efficiency trade-off is exacerbated in the presence of perishables: while equity requires a decision-maker to allocate *conservatively* across arriving streams of demand (Sinclair et al., 2022), perishability starts a "race against time." As goods perish due to a slow allocation rate, not only is efficiency further harmed, but so may be equity, as spoilage runs the risk of a decision-maker running out of items, with nothing left to give out by the end of the operating horizon.

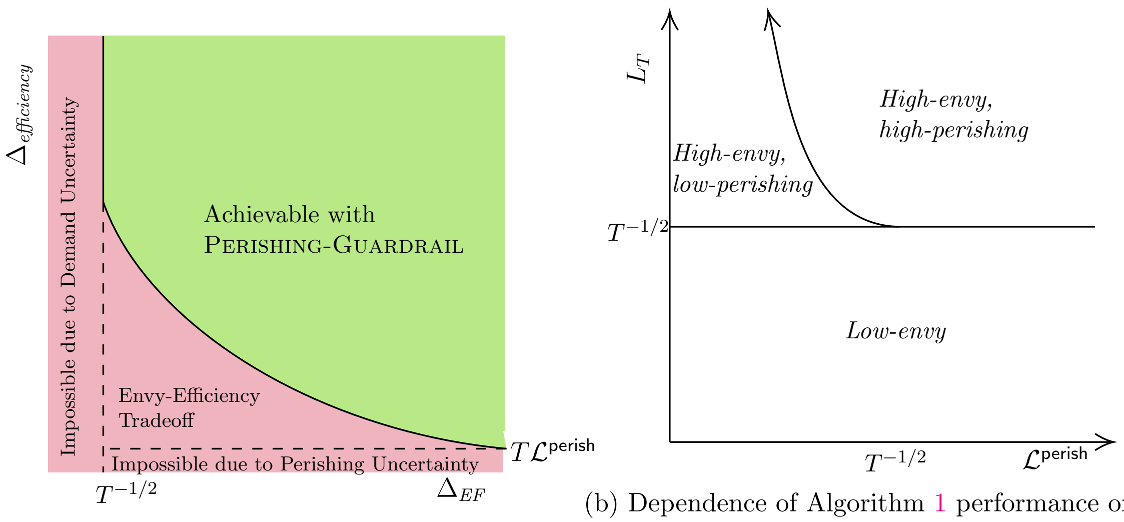
Thus motivated, this paper seeks to answer the following questions:

Do established equity-efficiency trade-offs in dynamic environments persist in the presence of perishable goods? If not, what limits do they impose on fair and efficient allocations? Can we design policies that perform well under these limits?

Before detailing our contributions, we highlight that, though this work is motivated by food bank distribution systems, the interplay between fairness and perishability is an important consideration in several other settings, e.g., vaccine distribution (Manshadi et al., 2023), electric vehicle charging (Gerding et al., 2019), and federated cloud computing (Aristotle Cloud Federation Project, 2022; Ghodsi et al., 2011).

1.1 Our contributions

We consider a model in which a decision-maker has a fixed budget of items (also referred to as *goods*, or *resources*) to be allocated over T discrete rounds. An a priori unknown number of individuals arrives in each period, each seeking a share of goods. Each agent is characterized by an observable type (drawn from a known, potentially time-varying distribution), associated with a linear utility function over the received allocation. Moreover, each unit of good has a stochastic perishing time (similarly drawn from a known distribution, independent of incoming arrivals), after which the



(a) Envy-Efficiency Pareto Frontier

(b) Dependence of Algorithm 1 performance on envy-perishing regime

Figure 1: Graphical representation of Theorems 3.7 and 4.2. Fig. 1a illustrates the envy-efficiency trade-off ($\Delta_{efficiency}$ vs. Δ_{EF}) achieved by PERISHING-GUARDRAIL (Algorithm 1). The dotted lines represent the impossibility results due to either demand or perishing uncertainty. The region below the solid line represents the impossibility due to the envy-efficiency trade-off; the green region is the achievable region for PERISHING-GUARDRAIL. Fig. 1b illustrates the phase transition between the performance of PERISHING-GUARDRAIL depending on the spoilage loss $\mathcal{L}^{\mathsf{perish}}$ (x-axis) and envy parameter L_T (y-axis).

good spoils and can no longer be allocated. The decision-maker allocates goods according to a fixed ordering (also referred to as perishing *prediction*, or *allocation schedule*), e.g., in increasing order of expected perishing time. The goal is to find a policy that trades off between three ex-post metrics:

1. *Hindsight Envy* (ENVY) – Maximum difference in utility obtained by any two agents.

2. *Counterfactual Envy* (Δ_{EF}) – Maximum difference between the utility obtained by any agent, and their utility under the static, proportional allocation (optimal in the no-perishing setting).

3. *Inefficiency* ($\Delta_{efficiency}$) – Amount of unallocated (including spoiled) goods at the end of the horizon.

For this setting, we first characterize the fundamental limits of perishability placed on any online algorithm. We argue that — contrary to the setting without perishable resources — envy ceases to be a meaningful metric for a large class of perishing processes. To see this, consider an extreme scenario in which all items perish at the end of the first round. Clearly, there is no hope of achieving low envy in such a setting since future demand can never be satisfied. Our first main contribution identifies a necessary and sufficient condition on the joint perishing and arrival distribution for low counterfactual envy to be an achievable desideratum (Theorem 3.2). From a managerial perspective, this characterization — which at a high level states that the cumulative perishing must lag behind cumulative arrivals — can be viewed as guidance on the composition of perishables in the initial budget. It moreover underscores the importance of leveraging *joint* information over perishing and demand: if demand is back-loaded, perishing times should be concentrated late in the horizon; if most demand arrives early, however, early perishing times are acceptable.

For this class of processes, which we term *offset-expiring*, zero spoilage occurs if the perishing prediction is perfect. We show, however, that inaccuracies in the allocation schedule pose an insurmountable barrier to any algorithm's performance by characterizing an unavoidable loss in equity and efficiency due to these errors (Theorem 3.7). In contrast to the no-perishing setting, in which the only source of loss is *exogenous* uncertainty in demand, in our setting the loss due to spoilage is *endogenous*: it crucially depends on the rate at which the algorithm allocates items. This endogeneity poses significant challenges in the analysis of any adaptive algorithm; designing a tractable approach to analyzing ex-post spoilage is the main technical contribution of our work. Additionally, the lower bounds we derive give rise to the key insight that, contrary to the "race against time" intuition under which a decision-maker must *increase* the allocation rate to prevent *avoidable* spoilage, achieving low hindsight envy in the presence of *unavoidable* spoilage requires a decision-maker to potentially allocate significantly less than the proportional allocation. Hence, perishability throws a wrench into the well-studied envy-efficiency trade-off: while hindsight and counterfactual envy are aligned in the no-perishing setting, these two may be at odds when goods spoil, since only high counterfactual-envy solutions may yield low hindsight-envy. The tension between efficiency and equity is exacerbated for the same reason, relative to the classical setting.

In our final technical contribution, we leverage these insights to construct an adaptive threshold algorithm (Algorithm 1) that achieves these lower bounds (Theorem 4.2). Our algorithm takes as input (*i*) the fixed allocation schedule, (*ii*) a desired upper bound on hindsight envy L_T, and (*iii*) a high-probability parameter δ. Given these inputs, it computes a high-probability lower bound on a budget-respecting zero-hindsight-envy solution, and an "aggressive" efficiency-improving allocation that is L_T away. In each round the algorithm chooses which of the two quantities to allocate to each individual, cautiously doing so by constructing pessimistic forecasts of future arrivals and perishing. While this algorithm is similar in flavor to state-of-the-art algorithms for the no-perishing setting (Sinclair et al., 2022), the main challenge it contends with is forecasting (endogenous) future spoilage. Here, we leverage the bounding technique used to construct our lower bounds, which relies on the analysis of a knife-edge, "slow" consumption process that tractably decouples past allocations from future perishing. Our algorithm's bounds give rise to three salient regimes (depicted graphically in Figure 1b), as a function of hindsight envy tolerance L_T and the unavoidable loss due to spoilage per period, denoted by $\mathcal{L}^{\mathsf{perish}}$:

1. *Low-envy* ($L_T \lesssim 1/\sqrt{T}$): there are no efficiency or counterfactual envy gains from deviating from the equitable solution $L_T = 0$.

2. *High-envy, high-perishing* ($L_T \gtrsim 1/\sqrt{T}, \mathcal{L}^{\mathsf{perish}} \gtrsim 1/\sqrt{T}$): inefficiency is invariant to L_T; setting $L_T \sim \mathcal{L}^{\mathsf{perish}}$ is optimal with respect to counterfactual envy.

3. *High-envy, low-perishing* ($L_T \gtrsim 1/\sqrt{T}, \mathcal{L}^{\mathsf{perish}} \lesssim 1/\sqrt{T}$): counterfactual envy increases as L_T, and inefficiency decreases as $1/L_T$, until reaching the unavoidable cumulative spoilage loss of $T\mathcal{L}^{\mathsf{perish}}$.

These results further highlight the extent to which a decision-maker is restricted in leveraging inequity to improve efficiency (and vice versa).

We complement our theoretical bounds in Section 5 by testing the practical performance of our algorithm on both synthetic and real-world datasets. Our experiments show that the unfairness required to achieve these efficiency gains is order-wise larger than in settings without perishable resources. Additionally, they underscore the weakness of *perishing-agnostic* online algorithms. We observe that these latter algorithms are incapable of leveraging unfairness to improve efficiency across a variety of perishing regimes. In contrast to these, our algorithm's construction of a *perishing-aware* baseline allocation $\underline{X}$ is necessary to mitigate stockouts across all — rather than simply worst-case — instances. These include instances where offset-expiry fails to hold with high probability, as is the case in the real-world dataset we use to calibrate our experiments (Keskin et al., 2022). Perhaps most surprisingly, despite our baseline allocation being significantly lower than that of algorithms that don't take into account perishability, our algorithm is *more efficient* than these more aggressive algorithms, in addition to being more fair. This observation contradicts the "race against time" intuition that aggressive allocations are necessarily more efficient than cautious ones. Finally, we numerically explore the question of setting practical allocation schedules that perform well along all metrics. Our main managerial insight is that ordering items in increasing order of a high-probability lower bound on their perishing time robustly trades off between the natural ordering that allocates items in increasing order of expected perishing time, and the inherent variability in the perishing process.

Paper organization.

We next survey the related literature. We present the model in Section 2, and formalize the limits of perishability in Section 3. In Section 4 we design and analyze an algorithm that achieves the derived lower bounds; we conclude with numerical experiments in Section 5.

1.2 Related work

Fairness in resource allocation has a long history in the economics and computation literature, beginning with Varian's seminal work (Varian, 1974, 1976). More recently, there has been ongoing work studying the intersection of fairness and operations, including assortment planning (Chen et al., 2022), pricing (Cohen et al., 2022; den Boer et al., 2022), incentive design (Freund and Hssaine, 2021), algorithmic hiring (Salem and Gupta, 2023), and societal systems more generally (Gupta and Kamble, 2021). We highlight the most closely related works below, especially as they relate to *online* fair allocation; see Aleksandrov and Walsh (2019b) for a survey.

Fair allocation without perishable resources. There exists a long line of work in which the *non-perishable* resource becomes available to the decision-maker online, whereas agents are fixed (Benade et al., 2018; Aleksandrov et al., 2015; Mattei et al., 2017, 2018; Aleksandrov and Walsh, 2019a; Banerjee et al., 2020; Bansal et al., 2020; Bogomolnaia et al., 2022; He et al., 2019; Aziz et al., 2016; Zeng and Psomas, 2020). These models lie in contrast to the one we consider, wherein resources are fixed and *individuals* arrive online. Papers that consider this latter setting include Kalinowski et al. (2013), who consider maximizing utilitarian welfare with indivisible goods. Gerding et al. (2019) consider a scheduling setting wherein agents have fixed and known arrival and departure times, as well as demand for the resource, and Hassanzadeh et al. (2023) allows individuals to arrive in multiple timesteps. A series of papers also consider the problem of fair division with minimal disruptions relative to previous allocations, as measured by a *fairness ratio*, a competitive ratio analog of counterfactual envy in our setting (Friedman et al., 2017; Cole et al., 2013; Friedman et al., 2015). Other works design algorithms with attractive competitive ratios with respect to Nash Social Welfare (Azar et al., 2010; Banerjee et al., 2020), or the max-min objective (Lien et al., 2014; Manshadi et al., 2023).

The above papers situate themselves within the adversarial, or worst-case, tradition. A separate line of work considers fair resource allocation in stochastic settings (Donahue and Kleinberg, 2020; Elzayn et al., 2019; Freund and Hssaine, 2021), as we do. The algorithms developed in these papers, however, are *non-adaptive*: they decide on the entire allocation upfront, *before* observing any of the realized demand. In contrast, we consider a model where the decision-maker makes the allocation decision in each round *after* observing the number of arrivals. Freeman et al. (2017) consider a problem in which agents' utilities are realized from an unknown distribution, and the budget resets in each round. They present algorithms for Nash social welfare maximization and discuss some of their properties. Our work is most closely related to (and indeed, builds upon) Sinclair et al. (2022), who first introduced the envy-freeness and efficiency tradeoff we are interested in. Our work considers the more challenging setting of perishable goods, which none of the aforementioned works consider.

Perishable resources. Though dynamic resource allocation of perishable goods has a long history in the operations research literature (see, e.g., Nahmias (2011) for a comprehensive survey of earlier literature), to the best of our knowledge, the question of *fairly* allocating perishable goods has attracted relatively little attention. We highlight the few relevant papers below. Perry (1999) and Hanukov et al. (2020) analyze FIFO-style policies for efficiency maximization in inventory models with Poisson demand and deterministic or Poisson perishing times. Motivated by the problem of electric vehicle charging, Gerding et al. (2019) consider an online scheduling problem where agents arrive and compete for a perishable resource that spoils at the end of every period, and as a result must be allocated at every time step. They consider a range of objectives, including: maximum total resource allocated, maximum number of satisfied agents, as well as envy-freeness. Bateni et al. (2022) similarly consider a setting wherein an arriving stream of goods perish immediately. Recent empirical work by Sengul Orgut and Lodree (2023) considers the problem of a food bank equitably and efficiently allocating perishable goods under complete information. Their case study on data from a partnering food bank numerically validates our theoretical results: in low-budget settings, there is little or no benefit to increasing inequity to counteract the inefficiency due (in part) to spoilage. In contrast to these latter papers, the model we consider locates itself within the smaller category of inventory models in which products have *random* lifetimes. The majority of these assume that items have exponentially distributed or geometric lifetimes (Bakker et al., 2012).

2 Preliminaries

We consider a decision-maker who, over T rounds, must allocate B divisible units (also referred to as *items*) of a single type of resource among a population of individuals. Let $\mathcal{B}$ denote the set of these B units.

2.1 Basic setup

Demand model. At the start of each round $t \in [T]$, a random number of individuals arrives, each requesting a share of units. Each individual is characterized by her *type* $\theta \in \Theta$, with $|\Theta| < \infty$. Each type θ is associated with a known utility function $u_\theta(x) = w_\theta \cdot x$ for a given allocation $x \in \mathbb{R}_+$ of the resource, with $w_\theta > 0$. We let $N_{t,\theta}$ denote the number of type θ arrivals in round t; $N_{t,\theta}$ is drawn independently from a known distribution, with $N_{t,\theta} \geq 1$ almost surely for all $t \in [T], \theta \in \Theta$. This latter assumption is for ease of exposition; our results continue to hold (up to constants) as long as $\mathbb{P}(N_{t,\theta} = 0)$ does not scale with T. For ease of notation we define $N_t = \sum_{\theta\in\Theta} N_{t,\theta}$ and $N = \sum_{t\in[T],\theta\in\Theta} N_{t,\theta}$. We assume $\mathbb{E}[N] = \Theta(T)$, and define $\beta_{avg} = B/\mathbb{E}[N]$ to be the average number of units per individual, with $\beta_{avg} = \Theta(1)$.

Perishing model. Each unit of resource $b \in \mathcal{B}$ is associated with a perishing time $T_b \in \mathbb{N}^+$ drawn from a known distribution. Items' perishing times are independent of one another and of the arrival process, and perishing occurs at the end of each round, after items have been allocated to individuals. For $t \in [T]$, we let $P_t = \sum_{b \in \mathcal{B}} \mathbb{1}\{T_b = t\}$ denote the number of units of resource perishing in period t.

The decision-maker has access to a *predicted ordering* according to which items perish; we will often refer to this ordering as the *allocation schedule*. We use $\sigma : \mathcal{B} \to [B]$ to denote this ordering, i.e., $\sigma(b)$ is rank of b in this ordering. For $b \in [B]$, $\sigma^{-1}(b)$ is used to denote the *identity* of the bth ranked item in σ, with $\sigma^{-1}(1)$ being the item that comes first in the allocation schedule. While our results allow σ to be arbitrary, in Section 5 we investigate natural choices of σ, such as increasing order of $\mathbb{E}[T_b]$.

Remark 2.1. *In this paper we restrict our attention to static (rather than time-varying and sample path-dependent) allocation schedules, given their practical relevance to the motivating real-world applications described in Section 1. We leave the nonstationary extension to future work.*

Additional notation. For any time-dependent quantity Y_t, we define $Y_{\leq t} = \sum_{t' \leq t} Y_{t'}$, $Y_{\geq t} = \sum_{t' \geq t} Y_{t'}$, along with their strict analogs. We let $w_{max} = \max_\theta w_\theta$, $\sigma^2_{t,\theta} = \text{Var}(N_{t,\theta}) < \infty$, and assume $\rho_{t,\theta} = |N_{t,\theta} - \mathbb{E}[N_{t,\theta}]| < \infty$ almost surely. Finally, let $\mu_{\max} = \max_t \mathbb{E}[N_t], \sigma^2_{\min} = \min_{t,\theta} \sigma^2_{t,\theta}, \sigma^2_{\max} = \max_{t,\theta} \sigma^2_{t,\theta}$, and $\rho_{\max} = \max_{t,\theta} \rho_{t,\theta}$. We use $\lesssim$ and $\gtrsim$ to denote the fact that inequalities hold up to polynomial factors of $\beta_{avg}, |\Theta|, w_{max}, \mu_{\max}, \sigma^2_{\min}, \sigma^2_{\max}, \log T$, and $\log(1/\delta)$. We summarize all notation in Table 5.

2.2 Notions of fairness and efficiency

The goal is to design a *fair* and *efficient* online algorithm that determines the amount to allocate to all $N_{t,\theta}$ individuals in each round t, for all $\theta \in \Theta$, given the remaining budget in each round. We assume this amount is allocated uniformly across all $N_{t,\theta}$ individuals of type θ. We use $X^{alg}_{t,\theta} \in \mathbb{R}$ to denote the per-individual amount distributed in period t, with $X^{alg} = (X^{alg}_{t,\theta})_{t \in [T]}$.

Our notions of online fairness and efficiency are motivated by the offline notion of *Varian Fairness* (Varian, 1974), and are the same as those considered in past works (Sinclair et al., 2022).

Definition 2.2 (Counterfactual Envy, Hindsight Envy, and Efficiency)**.** *Given budget B, realized demands $(N_{t,\theta})_{t \in [T], \theta \in \Theta}$, perishing times $(T_b)_{b \in [B]}$, and allocation schedule σ, for any online allocation defined by X^{alg} we define:*

- Counterfactual Envy*:*

$$\Delta_{EF} \triangleq \max_{t \in [T], \theta \in \Theta} \left| w_\theta \left(X^{alg}_{t,\theta} - \frac{B}{N} \right) \right|. \tag{1}$$

- Hindsight Envy*:*

$$\textsc{Envy} \triangleq \max_{t,t' \in [T]^2, \theta, \theta' \in \Theta^2} w_\theta (X^{alg}_{t',\theta'} - X^{alg}_{t,\theta}). \tag{2}$$

- Inefficiency*:*

$$\Delta_{efficiency} \triangleq B - \sum_{t \in [T], \theta \in \Theta} N_{t,\theta} X^{alg}_{t,\theta}. \tag{3}$$

In the offline setting *without* perishable goods, Varian (1974) established that $X_{t,\theta}^{opt} = B/N$ (referred to as the *proportional allocation*) is the optimal fair and efficient per-individual allocation. Hence, counterfactual envy Δ_{EF} can be interpreted as a form of regret with respect to this strong no-perishing benchmark, and can be used to characterize the impact of perishability on our algorithm's performance. Hindsight envy, on the other hand, measures how differently the online algorithm treats any two individuals across time. Finally, the efficiency of the online algorithm, $\Delta_{\mathit{efficiency}}$, measures how wasteful the algorithm was in hindsight. This could happen in two ways: either through spoilage, or because the decision-maker allocated too conservatively throughout the horizon, thus leaving a large number of unspoiled goods unallocated by T.

Even in simple settings without perishability, it is known that these metrics are at odds with each other in online settings. To see this, consider the following two scenarios. On the one hand, an algorithm can trivially achieve a hindsight envy of zero by allocating nothing to individuals in any round; this, however, would result in both high *counterfactual* envy, in addition to maximal inefficiency. On the other hand, a distance to efficiency of zero can trivially be satisfied by exhausting the budget in the first round, at a cost of maximal hindsight envy as individuals arriving at later rounds receive nothing. Sinclair et al. (2022) formalized this tension for the additive utility setting without perishable resources via the following lower bounds.

Theorem 2.3 (Theorems 1 and 2, Sinclair et al. (2022))**.** *Under any arrival distribution satisfying mild regularity conditions, there exists a problem instance without perishing, such that any algorithm must incur* $\Delta_{EF} \gtrsim \frac{1}{\sqrt{T}}$*, where* $\gtrsim$ *drops poly-logarithmic factors of* T*,* $\log(1/\delta)$*,* $o(1)$ *terms, and absolute constants. Moreover, any algorithm that achieves* $\Delta_{EF} = L_T = o(1)$ *or* $\text{Envy} = L_T = o(1)$ *must also incur waste* $\Delta_{\mathit{efficiency}} \gtrsim \min\{\sqrt{T}, 1/L_T\}$.

Since settings without perishable resources are a special case of our setting (e.g., a perishing process with $T_b > T$ a.s., for all $b \in \mathcal{B}$), this lower bound holds in our case; the goal then is to design algorithms that achieve this lower bound with high-probability. However, as we will see in the following section, perishing stochasticity is fundamentally distinct from, and more challenging than, demand stochasticity. This difference is particularly salient in regards to the envy-efficiency trade-off.

3 Limits of perishability

In the presence of perishable resources, a decision-maker must contend with two obstacles: (i) the "aggressiveness" of the perishing process, and (ii) errors in the perishing prediction σ. In this section we formalize the impact of these two challenges. Namely, we identify classes of perishing processes for which there is no hope of achieving the optimal fair allocation, and derive lower bounds on any algorithm's performance, as a function of the quality of the prediction σ.

In the remainder of the section, we say that an online algorithm is *feasible* over a sample if it does not run out of budget. Note that, if an algorithm is infeasible over a sample path, it necessarily achieves $\Delta_{EF} = \Theta(1)$.

3.1 Restricting the aggressiveness of the perishing process

We first argue that the proportional allocation $X_{t,\theta}^{opt} = B/N$ is unachievable unless one places restrictions on the rate at which items perish, even under full information over perishing times and demand.

To see this, consider an instance where all items perish at the end of the first round. There is no hope of achieving low envy in this setting, since there are no items left for arrivals from

$t = 2$ onwards. The following result establishes that the *only* perishing time realizations for which B/N is a meaningful benchmark are ones in which the fraction of perished items "lags behind" the proportion of arrivals in each period. We formalize this via the notion of *offset-expiry*, defined below.

Definition 3.1 (Offset-expiring process)**.** *A perishing process* $(T_b)_{t\in[T]}$ *is offset-expiring if:*

$$\frac{P_{<t}}{B} \leq \frac{N_{<t}}{N} \quad \forall\ t \geq 2.$$

Theorem 3.2 establishes that *offset-expiry* exactly captures the trajectories whereby B/N is a feasible allocation when units are allocated in increasing order of T_b. We refer to this latter ordering as the *hindsight optimal* ordering. We defer its proof to Appendix B.1.

Theorem 3.2. *$X_{t,\theta} = B/N$ for all t, θ is feasible under the hindsight optimal ordering if and only if the perishing process if offset-expiring.*

Thus motivated, for our main algorithmic result we restrict our attention to processes that satisfy the offset-expiry condition with high probability (via a relaxed notion of δ-*offset-expiry*, see Definition 4.1). We moreover provide verifiable necessary and sufficient conditions for this condition to hold in Section 4.2. From a managerial perspective, the (high-probability) offset-expiry condition provides decision-makers with guidance on the selection of perishable goods to stock at the beginning of the horizon. Within the context of food banks, for instance, it highlights that rejecting perishable goods outright is too severe a policy, and that the reasonable rule of thumb that says "don't have more spoilage than what you want to give out" is the *only* correct rule of thumb. It moreover underscores the importance of *jointly* considering demand and perishing processes in this selection.

Finally, we note that though the remainder of our work is focused on high-probability offset-expiring processes, an interesting future research direction is the question of fairly allocating non-offset-expiring goods under more meaningful notions of envy and inefficiency that don't penalize for aggressive perishing.

3.2 Unavoidable loss due to errors in allocation schedule

The previous section established that, even under full information about the perishing process, there exist restrictions on the aggressiveness of the process for the optimal fair allocation to be achievable. We next show that, even under offset-expiry, the quality of the allocation schedule σ is crucial in determining what any online algorithm can achieve. To see this, consider an instance where $B = T$, $|\Theta| = 1$ (with $w_\theta = 1$), and $N_t = 1$ for all t. Suppose moreover that exactly one unit perishes in each period (i.e., the perishing process is offset-expiring), but σ reverses the true perishing order. In this case, allocating $B/N = 1$ to each arrival under σ is infeasible, since after $T/2$ rounds the algorithm will have run out of items.

Motivated by this example, our key insight is that, for any static allocation X, there exists a worst-case loss due to errors in σ, denoted by $\overline{\Delta}(X)$, that the algorithm incurs. As a result, rather than having a budget of B items, the algorithm has an *effective* budget of $B - \overline{\Delta}(X)$ items. Under this effective budget, any feasible stationary allocation must set X such that $\overline{N}X \leq B - \overline{\Delta}(X)$, where $\overline{N}$ is a high-probability upper bound on N. Noting that $X = 0$ is always a feasible solution to this inequality, a natural choice is to set:

$$\underline{X} = \sup\left\{X \ \middle|\ X \leq \frac{B - \overline{\Delta}(X)}{\overline{N}}\right\}, \tag{4}$$

if this supremum is achieved. When $\overline{\Delta}(X) = 0$ for all X (i.e., either items don't perish or predictions are perfect), $\underline{X} = B/\overline{N}$, and we recover the conservative allocation of the no-perishing setting (Sinclair et al., 2022). The notion of *σ-induced loss* plays a central role in our results.

Definition 3.3 (σ-induced loss)**.** *We define $\mathcal{L}^{\mathsf{perish}} = \frac{B}{\overline{N}} - \underline{X}$ to be any algorithm's* σ**-induced loss***. We moreover term $T\mathcal{L}^{\mathsf{perish}}$ to be the cumulative σ-induced loss.*

By (4), given the worst-case perishing loss $\overline{\Delta}(X)$ for any allocation X, one can compute $\underline{X}$ via line search. Obtaining tight bounds on this quantity, however, is the challenging piece. To see this, note that for any algorithm, the quantity of goods that perished by the end of the horizon is:

$$\sum_{t\in[T]} \sum_{b\in\mathcal{B}_t^{alg}} (B_t^{alg}(b) - X_t^{alg}(b))^+ \cdot \mathbb{1}\{T_b = t\}, \tag{5}$$

where $\mathcal{B}_t^{alg}$ denotes the set of remaining items at the beginning of period t, $B_t^{alg}(b)$ is the *quantity* of item b remaining at the beginning of period t, and $X_t^{alg}(b)$ is the quantity of item b given out in period t. Since $X_t^{alg}(b)$ depends on the perishing realizations of previous rounds, computing this quantity requires the ability to simulate sufficiently many replications of the static allocation process under X, for all $X \in [0, B/\overline{N}]$, and for each of these replications to compute the number of unallocated goods that perished by the end of the horizon under this allocation, an approach which fails to scale.

To tackle this difficulty, it will be useful for us to consider a "slow" consumption process, in which $\underline{N}_{\leq t}$ — a high-probability lower bound on $N_{\leq t}$ — individuals arrive before $t+1$, $\underline{N}_{\leq t}X$ items are allocated up to period $t \in [T]$, and no items perish. For $b \in \mathcal{B}$, we let $\tau_b(1 \mid X, \sigma)$ be the period in which b *would* have been entirely allocated under this slow consumption process. Formally,

$$\tau_b(1 \mid X, \sigma) = \inf\bigl\{t \geq 1 : \underline{N}_{\leq t}X \geq \sigma(b)\bigr\}. \tag{6}$$

$\tau_b(1 \mid X,\ \sigma)$ represents an upper bound on the time an algorithm using static allocation X would allocate b, since items ranked higher than b may have perished, thus decreasing the time at which b is allocated. We define $\mu(X) = \sum_{b\in\mathcal{B}} \mathbb{P}(T_b < \min\{T, \tau_b(1 \mid X, \sigma)\})$ and let

$$\overline{\Delta}(X) = \min\bigl\{B, \mu(X) + \textsc{Conf}_1^P(\mu(X))\bigr\}, \tag{7}$$

where $\textsc{Conf}_1^P(\mu(X))$ is an appropriately chosen confidence bound, to be specified later.

Remark 3.4. *We henceforth assume for simplicity that the supremum on the right-hand side of (4) is attained. This is without loss to our results, since our bounds depend on the σ-induced loss $\mathcal{L}^{\mathsf{perish}}$. Hence, if the supremum fails to be attained, one can define $\underline{X} = X^* - \epsilon$, where X^* is the point of discontinuity and $\epsilon = o(1)$ guarantees feasibility of $\underline{X}$.*

Remark 3.5. *Since $N_{\leq t} \geq t$ for all $t \in [T]$ one can similarly define $\tau_b(1 \mid X, \sigma) = \inf\{t \geq 1 : tX \geq \sigma(b)\} = \lceil \frac{\sigma(b)}{X} \rceil$ as an upper bound on the latest possible perishing time. This quantity can then be interpreted as the "effective rank" of item b. This interpretable simplification comes at the cost of our algorithm's practical performance, but does not affect our subsequent theoretical bounds.*

Example 3.6 illustrates the $\underline{X}$ construction for a toy instance.

Example 3.6. *Consider a setting where $B = T = 4$, $|\Theta| = 1$, and $N_t = 1$ for all $t \in [T]$, with the following perishing time distributions for each item:*

$$T_1 = \begin{cases} 1 & w.p.\ 1/2 \\ 2 & w.p.\ 1/2 \end{cases} \qquad T_2 = \begin{cases} 1 & w.p.\ 1/2 \\ 4 & w.p.\ 1/2 \end{cases} \qquad T_3 = \begin{cases} 2 & w.p.\ 1/2 \\ 3 & w.p.\ 1/2 \end{cases} \qquad T_4 = \begin{cases} 3 & w.p.\ 1/2 \\ 4 & w.p.\ 1/2 \end{cases}.$$

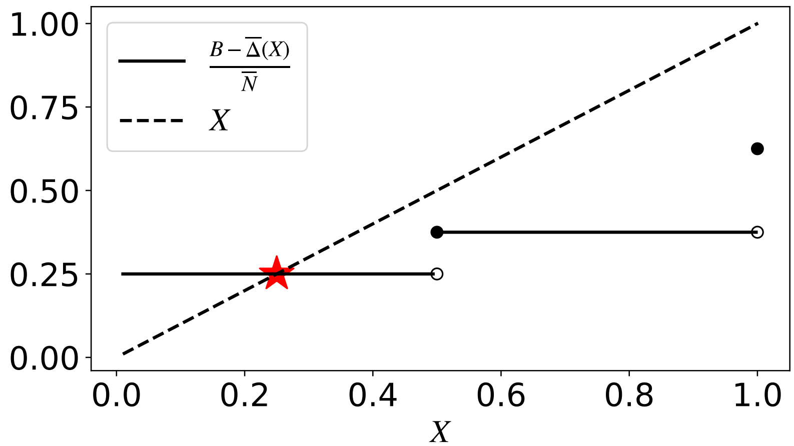


Figure 2: Illustrating the $\underline{X}$ construction (4) for the toy instance in Example 3.6. The dashed line corresponds to the line $Y = X$, and the solid line to $(B - \overline{\Delta}(X))/\overline{N}$. Here, $\underline{X}$ is represented by the red star, the point at which the solid and dashed lines intersect.

Let $\sigma(b) = b$ for all b (i.e., items are allocated in increasing order of expected perishing time, with ties broken in favor of earliest possible perishing time). Fig. 2 illustrates the solution to (4) for this instance, with $\text{CONF}_1^P(\mu(X)) = 0$ for all X. Observe that $\underline{X} = 0.25$, whereas the proportional allocation $B/N = 1$.

Observe that the perishing process described in Example 3.6 is not almost surely offset-expiring, since items 1 and 2 both perish at the end of period 1 with probability 1/4. However, lack of offset-expiry is *not* the reason that any online algorithm incurs an additional perishing-related loss. Theorem 3.7 establishes that any online algorithm's performance necessarily scales with the σ-induced loss, even under offset-expiry.

Theorem 3.7. *There exists an offset-expiring instance such that, for any online algorithm that is feasible with probability at least α, the following holds with probability at least α:*

$$\Delta_{\textit{efficiency}} \geq T\mathcal{L}^{\mathsf{perish}} \qquad \Delta_{EF} \geq \mathcal{L}^{\mathsf{perish}}.$$

Proof. Consider an instance with $B = T$, $|\Theta| = 1$ and $N_t = 1$ for all t. Suppose moreover that resources have deterministic perishing times, with $T_b = b$ for all b, and $\sigma(b) = T + 1 - b$. ($T_b = b$ implies that the process is offset-expiring.) Since the perishing and demand processes are deterministic, we let $\text{CONF}_1^P(\mu(X)) = 0$ for all X, and $\overline{N} = N$. For ease of notation, we omit the dependence of all quantities on θ in the remainder of the proof. The following lemma states that under this flipped ordering, any online algorithm is severely limited in its total allocation.

Lemma 3.8. *Any feasible algorithm must have $\sum_t X_t^{alg} \leq 1$.*

Proof. For any feasible algorithm, there must exist an available unit in period T. Since the only unit that has not perished by $t = T$ is $b = 1$, it must be that this unit is available in period T. Thus, we must have $\sum_t X_t^{alg} \leq 1$ (else, $b = 1$ will have been allocated before T). $\square$

Lemma 3.8 implies that any feasible stationary algorithm, which allocates a fixed amount $X_t^{alg} = X$ for all t, must have $X \leq \frac{1}{N}$. We use this fact to bound $\overline{\Delta}(X)$, for any feasible stationary allocation.

Lemma 3.9. *For any* $0 < X \leq \frac{1}{N}$, $\overline{\Delta}(X) \geq T - 1$.

Proof. By definition:

$$\tau_b(1 \mid X, \sigma) = \inf\{t > 0 : N_{\leq t} X \geq \sigma(b)\} = \inf\{t > 0 : tX \geq \sigma(b)\} = \lceil \frac{\sigma(b)}{X} \rceil = \lceil \frac{T+1-b}{X} \rceil \geq T(T+1-b),$$

where the final inequality follows from the fact that $X \leq \frac{1}{N} = \frac{1}{T}$. Hence,

$$\overline{\Delta}(X) = \sum_b \mathbb{1}\{T_b < \min\{T, \tau_b(1 \mid X, \sigma)\}\} \geq \sum_b \mathbb{1}\{b < \min\{T, T(T+1-b)\}\} = \sum_b \mathbb{1}\{b < T\} = T - 1,$$

where the inequality uses the lower bound on $\tau_b(1 \mid X, \sigma)$ in addition to the assumption that $T_b = b$. □

Putting Lemma 3.8 and Lemma 3.9 together, we have:

$$\underline{X} := \sup\{X : X \leq \frac{B - \overline{\Delta}(X)}{N}\} \leq \sup\{X : X \leq \frac{T - (T-1)}{T}\} = \frac{1}{T} \implies \mathcal{L}^{\mathsf{perish}} \geq 1 - 1/T.$$

We now show the lower bounds on Δ_{EF} and $\Delta_{\mathit{efficiency}}$. By Lemma 3.8, $\sum_t X_t^{alg} \leq 1$, which implies that $\min_t X_t^{alg} \leq \frac{1}{T}$. Hence, $\Delta_{EF} = \max_t |1 - X_t^{alg}| \geq 1 - \frac{1}{T} = \mathcal{L}^{\mathsf{perish}}$. Moreover, $\Delta_{\mathit{efficiency}} = T - \sum_t X_t^{alg} \geq T - 1 = T\mathcal{L}^{\mathsf{perish}}$. □

4 The Algorithm

Our algorithm, Perishing-Guardrail, takes as input (i) a desired bound on envy L_T, and (ii) a high-probability parameter δ. The algorithmic approach is tackled in three steps:

1. Constructing a static allocation (also referred to as *baseline allocation* or *lower guardrail*), $\underline{X}$, under which the algorithm doesn't run out of budget with high probability. Motivated by Section 3.2, we let
$$\underline{X} = \sup\left\{X \mid X \leq \frac{B - \overline{\Delta}(X)}{\overline{N}}\right\},$$
with $\overline{N} = \mathbb{E}[N] + \textsc{Conf}_{0,T}^{N}$, for an appropriately defined high-probability confidence term $\textsc{Conf}_{0,T}^{N}$.

2. Setting an "aggressive" allocation $\overline{X} = \underline{X} + L_T$ to improve efficiency.

3. Determining an appropriate threshold condition that indicates when to allocate $\overline{X}$.

Though the above approach is similar to the guardrail algorithm proposed by Sinclair et al. (2022) for the no-perishing setting, we emphasize that identifying the appropriate static allocation and threshold condition under perishing uncertainty poses significant challenges that do not exist in the classical setting. In this latter setting, the natural static allocation that guarantees budget-feasibility is the proportional allocation in a *high-demand* regime, i.e., $\underline{X} = B/\overline{N}$. Part of the reason this is easily handled is the fact that arrival uncertainty is *exogenous*, i.e. it is invariant to the decisions made by the algorithm. On the other hand, uncertainty around perishing is *endogenous*: as discussed in Section 3, though the distribution around perishing times is fixed, how many — and which — items perish depends heavily on the rate at which items are being allocated, which itself depends on the rate at which items perish. The threshold condition we next describe must contend with this knife-edge effect.

Determining the threshold condition. Recall, the "aggressive" allocation $\overline{X}$ will be used to improve our algorithm's efficiency, at the cost of higher envy. In each period, our algorithm checks whether there is enough budget remaining to accommodate (i) an aggressive allocation in the current period, (ii) a conservative allocation in all future periods, under high demand, and (iii) perishing that may occur under future conservative allocations. The main challenge here lies in estimating (iii): over-optimism runs an increased risk of exhausting the budget early, due to the same phenomenon as that described in Section 3.2, whereas over-pessimism fails to take advantage of efficiency gains from aggressively allocating.

For $t \in [T]$, we let $\overline{P}_t$ denote our algorithm's worst-case perishing prediction. As above, the "slow" consumption process will allow us to obtain a closed-form characterization of $\overline{P}_t$. In particular, for $t \in [T]$, $b \in \mathcal{B}$, we consider the notion of "latest allocation time" after t:

$$\tau_b(t \mid \underline{X}, \sigma) = \inf\Big\{t' \geq t : \underline{N}_{<t}\underline{X} + \underline{N}_{[t,t']}\underline{X} \geq \sigma(b)\Big\}, \tag{8}$$

where $\underline{N}_{[t,t']} = \mathbb{E}\big[N_{[t,t']}\big] - \textsc{Conf}^N_{t,t'}$ for an appropriately defined high-probability confidence term $\textsc{Conf}^N_{t,t'}$. In words, $\underline{N}_{<t}\underline{X} + \underline{N}_{[t,t']}\underline{X}$ corresponds to the least amount the algorithm could have consumed by t' (either by allocating or via perishing). Hence, if b is in the set of remaining items at the beginning of period t, with high probability it will be allocated before $\tau_b(t \mid \underline{X}, \sigma)$. Via similar logic, we let $\overline{\mathcal{B}}_t = \{\sigma^{-1}(\lceil \underline{N}_{<t}\underline{X}\rceil), \ldots, \sigma^{-1}(B)\}$ be the set of items remaining under this slow process, and define the expected number of items that perish from t onwards as:

$$\eta_t = \sum_{b \in \overline{\mathcal{B}}_t} \mathbb{P}(t \leq T_b < \min\{T, \tau_b(t \mid \underline{X}, \sigma)\}). \tag{9}$$

The pessimistic forecast of future spoilage is then defined as:

$$\begin{cases} \overline{P}_t & = \min\big\{\overline{P}_{t-1}, \eta_t + \textsc{Conf}^P_t(\eta_t)\big\} \quad \forall\, t \in [T], \\ \overline{P}_0 & = B \end{cases} \tag{10}$$

for an appropriately defined confidence term $\textsc{Conf}^P_t(\cdot)$. Note that $\overline{P}_1 = \overline{\Delta}(\underline{X})$ (see Eq. (7)).

We present our algorithm, Perishing-Guardrail, in Algorithm 1. For $t \in [T]$, $\mathcal{B}^{alg}_t$ is used to denote the set of remaining resources at the beginning of time t, and B^{alg}_t the *quantity* of remaining resources at the beginning of the period. Moreover, let $\mathcal{A}_t$ be the set of items allocated in round t, and PUA^{alg}_t the quantity of unallocated items that perished at the end of round t.

4.1 Performance guarantee

Definition 4.1 (δ-Offset-Expiry.)**.** *A perishing process is* δ-offset-expiring *if:*

$$\mathbb{P}\left(\frac{P_{<t}}{B} \leq \frac{N_{<t}}{N} \quad \forall\, t \geq 2\right) \geq 1 - \delta$$

Theorem 4.2. *For $t' > t$, define the confidence terms:*

- $\textsc{Conf}^N_{t,t'} = \sqrt{2(t'-t)|\Theta|\rho^2_{\max}\log(2T^2/\delta)}$
- $\textsc{Conf}^P_t(\eta_t) = \frac{1}{2}\left(\log(3t\log T/\delta) + \sqrt{\log^2(3t\log T/\delta) + 8\eta_t\log(3t\log T/\delta)}\right)$

ALGORITHM 1: PERISHING-GUARDRAIL

Input: Budget $B = B_1^{alg}$, allocation schedule σ, envy parameter L_T, arrival confidence terms $(\textsc{Conf}^N_{t,t'})_{t,t' \in \{0,\dots,T\}}$, perishing confidence terms $(\textsc{Conf}^P_t(\cdot))_{t\in\{1,\dots,T\}}$, and perishing inputs $(\eta_t)_{t\in[T]}$ given by (9)

Output: An allocation $X^{alg} \in \mathbb{R}^{T\times|\Theta|}$

Compute $\underline{X} = \sup\left\{X \mid X \leq \frac{B-\overline{\Delta}(X)}{\overline{N}}\right\}$ and set $\overline{X} = \underline{X} + L_T$.

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

  Compute $\overline{P}_t = \min\{\overline{P}_{t-1}, \eta_t + \textsc{Conf}^P_t(\eta_t)\}$ `// Compute ''worst-case'' future perishing`

  if $B_t^{alg} < N_t \underline{X}$ **then** `// insufficient budget to allocate lower guardrail`

    Set $X^{alg}_{t,\theta} = \frac{B_t^{alg}}{N_t}$ for each $\theta \in \Theta$. Allocate items $b \in \mathcal{B}_t^{alg}$ according to σ.

  else if $B_t^{alg} - N_t\overline{X} \geq \underline{X}(\mathbb{E}[N_{>t}] + \textsc{Conf}^N_{t,T}) + \overline{P}_t$ **then**`// use upper guardrail`

    Set $X^{alg}_{t,\theta} = \overline{X}$ for each $\theta \in \Theta$. Allocate items $b \in \mathcal{B}_t^{alg}$ according to σ.

  else `// use lower guardrail`

    Set $X^{alg}_{t,\theta} = \underline{X}$ for each $\theta \in \Theta$. Allocate items $b \in \mathcal{B}_t^{alg}$ according to σ.

  Update $B^{alg}_{t+1} = B^{alg}_t - N_t X^{alg}_t - \mathrm{PUA}^{alg}_t$

end

return X^{alg}

Then, for any δ-offset-expiring perishing process, with probability at least $1-3\delta$, Algorithm 1 achieves:

$$\Delta_{EF} \lesssim \max\{L_T, \mathcal{L}^{\textsf{perish}} + 1/\sqrt{T}\} \qquad \Delta_{\textit{efficiency}} \lesssim \min\left\{\sqrt{T}, L_T^{-1} + \sqrt{TL_T^{-1}\mathcal{L}^{\textsf{perish}}}\right\} + T\mathcal{L}^{\textsf{perish}}$$

$$\textsc{Envy} \lesssim L_T$$

where $\lesssim$ drops poly-logarithmic factors of T, $\log(1/\delta)$, $o(1)$ terms, and absolute constants.

In Appendix B.2, we show that these bounds are indeed tight, relative to the lower bounds in Theorem 2.3 and Theorem 3.7. We dedicate the remainder of this section to further parsing the bounds on counterfactual envy and efficiency — graphically represented in Fig. 1 — given the scalings of L_T and $\mathcal{L}^{\textsf{perish}}$. In Section 4.2 we derive necessary and sufficient conditions on the perishing process for δ-offset-expiry to hold. We finally prove Theorem 4.2 in Section 4.3.

Note that our result is a strict generalization of Sinclair et al. (2022): in the simplest setting where $\mathcal{L}^{\textsf{perish}} = 0$ (i.e., no perishing, or perishing with perfect predictions), we recover the trade-off they identified. The following corollary simplifies our bounds in the "low-envy" setting where $L_T \lesssim 1/\sqrt{T}$.

Corollary 4.3 (Low-Envy)**.** *Suppose $L_T \lesssim 1/\sqrt{T}$. Then,* PERISHING-GUARDRAIL *achieves with probability at least $1-3\delta$:*

$$\Delta_{EF} \lesssim \mathcal{L}^{\textsf{perish}} + 1/\sqrt{T} \qquad \Delta_{\textit{efficiency}} \lesssim \sqrt{T} + T\mathcal{L}^{\textsf{perish}}.$$

Corollary 4.3 implies that there is no efficiency benefit to increasing L_T as long as $L_T \lesssim 1/\sqrt{T}$. When $\mathcal{L}^{\textsf{perish}} \lesssim 1/\sqrt{T}$, our algorithm incurs $\widetilde{O}(\sqrt{T})$ envy and inefficiency. In this case, these quantities are driven by the exogenous uncertainty in demand. When $\mathcal{L}^{\textsf{perish}} \gtrsim 1/\sqrt{T}$, on the other hand, envy and inefficiency are driven by unavoidable perishing due to prediction errors.

Corollary 4.4 next considers the "high-envy, high-perishing" setting, on the other extreme of the spectrum.

Corollary 4.4 (High-Envy, High-Perishing). *Suppose $L_T \gtrsim 1/\sqrt{T}$ and $\mathcal{L}^{\textsf{perish}} \gtrsim 1/\sqrt{T}$. Then,* Perishing-Guardrail *achieves with probability $1-3\delta$:*

$$\Delta_{EF} \lesssim \max\{L_T, \mathcal{L}^{\textsf{perish}}\} \qquad \Delta_{\mathit{efficiency}} \lesssim T\mathcal{L}^{\textsf{perish}}.$$

Similarly in this regime, increasing L_T doesn't guarantee arbitrary gains in efficiency; thus, setting $L_T \sim \mathcal{L}^{\textsf{perish}}$ is optimal. We conclude by exploring the more nuanced "high-envy, low-perishing" regime. In this setting, our algorithm's guarantees depend on whether the efficiency gain from increasing envy, L_T^{-1}, exceeds $T\mathcal{L}^{\textsf{perish}}$. We defer its proof to Appendix B.3.

Corollary 4.5 (High-Envy, Low-Perishing). *Suppose $L_T \gtrsim 1/\sqrt{T}$, $\mathcal{L}^{\textsf{perish}} \lesssim 1/\sqrt{T}$, and $T\mathcal{L}^{\textsf{perish}} \lesssim L_T^{-1}$. Then,* Perishing-Guardrail *achieves with probability at least $1-3\delta$:*

$$\Delta_{EF} \lesssim L_T \qquad \Delta_{\mathit{efficiency}} \lesssim L_T^{-1}$$

On the other hand, if $T\mathcal{L}^{\textsf{perish}} \gtrsim L_T^{-1}$, Perishing-Guardrail *achieves with probability at least $1-3\delta$:*

$$\Delta_{EF} \lesssim L_T \qquad \Delta_{\mathit{efficiency}} \lesssim T\mathcal{L}^{\textsf{perish}}$$

Thus, in "high-envy, low-perishing" regimes, if the cumulative σ-induced loss is order-wise dominated by the efficiency gains from envy (otherwise phrased, our allocation schedule σ is high-enough quality that perishing is low), increasing L_T allows Perishing-Guardrail to achieve inversely proportional gains in efficiency. One can do this until moving into the regime where $L_T^{-1} \lesssim T\mathcal{L}^{\textsf{perish}}$ (i.e., the cumulative σ-induced loss dominates efficiency gains from envy). At this point, further increasing L_T hurts envy, and has no order-wise impact on efficiency.

Section 4.2 next provides conditions on the perishing distribution to satisfy δ-offset expiry.

4.2 On δ-offset expiry

In order to highlight the salient parameters of the perishing process that dictate offset-expiry, in this section we assume $B = N = T$ almost surely, with $N_t = 1$ for all t. At the cost of cumbersome algebra, one can relax this assumption and derive entirely analogous results. In this case, δ-offset expiry (Definition 4.1) reduces to $\mathbb{P}(P_{<t} \le t-1 \ \forall\, t \ge 2) \ge 1-\delta$.

For $t \in [T]$, let $\mathcal{B}_{<t}^{rand} = \{b : \mathbb{P}(T_b < t) \in (0,1)\}$, and $\mathcal{B}_{<t}^{det} = \{b : \mathbb{P}(T_b < t) = 1\}$. Proposition 4.6 states that, in expectation, no more that $t-1$ items can perish for δ-offset expiry to hold for non-trivial values of δ. We defer all proofs in this section to Appendix B.4.

Proposition 4.6. *Suppose there exists $t \ge 2$ such that $\mathbb{E}[P_{<t}] > t-1$. If $\mathcal{B}_{<t}^{rand} = \emptyset$, δ-offset-expiry cannot be satisfied for any value of $\delta \in (0,1)$. Else, δ-offset-expiry cannot be satisfied for $\delta < \frac{1}{2} - \mathrm{Std}[P_{<t}]^{-3} \cdot T$.*

Note that this necessary condition fails to hold for one of the most standard models of perishing: geometrically distributed perishing with parameter $1/T$ (that is, a constant fraction of items perish in each period). This highlights that one of the most popular models in the literature is, in a sense, far too pessimistic; for this setting, there is no hope of achieving low envy and efficiency with high probability. Proposition 4.7 next establishes a sufficient condition for δ-offset expiry to hold.

Proposition 4.7. *Suppose that $\mathbb{E}[P_{<t}] \le t-1$ for all $t \ge 2$. Then, the perishing process is δ-offset-expiring for any $\delta \ge \sum_{t=2}^{T} \min\left\{\left(\frac{\mathrm{Std}[P_{<t}]}{t-\mathbb{E}[P_{<t}]}\right)^2, \exp\left(-\frac{2(t-\mathbb{E}[P_{<t}])^2}{|\mathcal{B}_{<t}^{rand}|}\right)\right\}\mathbb{1}\{|\mathcal{B}_{<t}^{rand}| > 0\}$.*

Proposition 4.7 states that *either* the expected lag $t - \mathbb{E}[P_{<t}]$ must be large, *or* the coefficient of variation with respect to the random lag process $t - P_{<t}$, must be small. This then reduces to a bound on the variability of $P_{<t}$ in settings where perishing closely tracks demand. In our numerical experiments (Section 5) we instantiate the above bounds for common distributions.

4.3 Analysis

The proof of Theorem 4.2 is based on three main building blocks:

- **Defining and bounding the "good event":** (Section 4.3.1): We first show that, with probability at least $1 - \delta$, the realizations of future arrivals and perishing are no worse than our algorithm's pessimistic predictions. As a result, it suffices to condition over such "good" sample paths.
- **Establishing feasibility of $\underline{X}$** (Section 4.3.2): For this good event, we show that the static allocation $\underline{X}$ computed at the start of the horizon will never exhaust the budget, despite incurring σ-induced loss.
- **Improving efficiency via $\overline{X}$** (Section 4.3.3): We next show that the threshold condition guarantees that the algorithm allocates aggressively enough throughout the horizon to ensure high efficiency.

We use these building blocks for our final bounds on envy and efficiency in Section 4.3.4.

4.3.1 Defining and bounding the "good event"

We analyze the performance of our algorithm under a so-called "good event" $\mathcal{E}$, the intersection of the following three events:

1. $\mathcal{E}_N = \Big\{ |N_{(t,t')} - \mathbb{E}\big[N_{(t,t')}\big]| \le \textsc{Conf}^N_{t,t'} \ \forall\, t, t' > t \ \Big\}$,
2. $\mathcal{E}_{\overline{P}} = \Big\{ \overline{P}_t \ge \mathrm{PUA}^{alg}_{\ge t} \,\forall\, t \in \{0, 1, \dots, T\} \Big\}$, where $\mathrm{PUA}^{alg}_{\ge t}$ denotes the quantity of unallocated items that perished between the end of round t and the end of round $T - 1$,
3. $\mathcal{E}_{oe} = \Big\{ \frac{P_{<t}}{B} \le \frac{N_{<t}}{N} \quad \forall\, t \ge 2 \Big\}$.

$\mathcal{E}$ represents the event that the arrival process falls close to its mean, that $\overline{P}_t$ is indeed a pessimistic estimate of the unallocated goods that perish in the future, and that the process is offset-expiring.

Since the process is δ-offset expiring, we have that $\mathbb{P}(\mathcal{E}_{oe}) \ge 1 - \delta$ by assumption. The following lemma provides the high-probability bound on $\mathbb{P}(\mathcal{E}_N)$. We defer its proof — which follows from a standard application of Hoeffding's inequality — to Appendix B.5.

Lemma 4.8. *$\mathcal{E}_N$ holds with probability at least $1 - \delta$.*

The main challenge in analyzing $\mathcal{E}$ lies in showing that $\overline{P}_t$ is indeed a pessimistic estimate of our algorithm's future perishing. To upper bound the amount of unallocated resources that perished between t and $T - 1$, we must account for both the uncertainty in arrivals and the realized order in which resources perished, and relate these two sources of uncertainty to the time at which the algorithm intended to allocate these resources. Establishing this upper bound hinges upon the careful construction of the "slow" consumption process, which decouples future perishing from future allocations to compute $\overline{P}_t$. We formalize these ideas in Lemma 4.9.

Lemma 4.9. *Given $\mathcal{E}_N$, $\mathcal{E}_{\overline{P}}$ holds with probability at least $1-\delta$.*

Proof of Lemma 4.9. We prove the claim by induction.
Base case: $t=0$. Since $\overline{P}_0 = B$ by definition, we have $\overline{P}_0 \geq \mathrm{PUA}^{alg}_{\geq 0}$ trivially.
Inductive step: $t-1 \to t$. Since $\mathrm{PUA}^{alg}_{\geq t}$ represents the amount of unallocated goods that perished between t and $T-1$, we have:

$$\mathrm{PUA}^{alg}_{\geq t} = \sum_{\tau=t}^{T-1} \sum_{b \in \mathcal{B}^{alg}_\tau} \mathbb{1}\{T_b = \tau, b \notin \mathcal{A}_\tau\} = \sum_{b \in \mathcal{B}^{alg}_t} \mathbb{1}\{T_b \geq t, b \text{ not allocated before } T_b\} \tag{11}$$

Recall, $\tau_b(t \mid \underline{X}, \sigma)$ (8) is an upper bound on the *latest* possible time the algorithm would have allocated b. This follows from the fact that the *least* the algorithm could have allocated before t under $\mathcal{E}_N$ is $\underline{N}_{<t}\underline{X}$. Similarly, the least amount of goods that the algorithm can allocate between t and $t' \geq t$ is $\underline{N}_{[t,t']}\underline{X}$. Hence, if $\underline{N}_{<t}\underline{X} + \underline{N}_{[t,t']}\underline{X} \geq \sigma(b)$ and b did not perish before t', it must be that the algorithm allocated b. Applying this logic to (11), we have:

$$\mathrm{PUA}^{alg}_{\geq t} \leq \sum_{b \in \mathcal{B}^{alg}_t} \mathbb{1}\{t \leq T_b < \min\{T, \tau_b(t \mid \underline{X}, \sigma)\}\}, \tag{12}$$

since it could have been that an item b' such that $\sigma(b') < \sigma(b)$ perished early, resulting in an earlier allocation time for b. A similar argument gives us that $\mathcal{B}^{alg}_t \subseteq \overline{\mathcal{B}}_t$. Plugging this into (12):

$$\mathrm{PUA}^{alg}_{\geq t} \leq \sum_{b \in \overline{\mathcal{B}}_t} \mathbb{1}\{t \leq T_b < \min\{T, \tau_b(t \mid \underline{X}, \sigma)\}\}. \tag{13}$$

Recall, $\eta_t = \sum_{b \in \overline{\mathcal{B}}_t} \mathbb{P}(t \leq T_b < \min\{T, \tau_b(t \mid \underline{X}, \sigma)\})$. Applying a Chernoff bound (see Corollary C.3) to the right-hand side of (13), we obtain that, with probability at least $1 - \delta/(3t\log(T))$:

$$\begin{aligned}\mathrm{PUA}^{alg}_{\geq t} &\leq \eta_t + \frac{1}{2}\left(\log(3t\log(T)/\delta) + \sqrt{\log^2(3t\log(T)/\delta) + 8\eta_t \log(3t\log(T)/\delta)}\right) \\ &= \eta_t + \textsc{Conf}^P_t(\eta_t).\end{aligned}$$

Moreover, $\mathrm{PUA}^{alg}_{\geq t} \leq \mathrm{PUA}^{alg}_{\geq (t-1)} \leq \overline{P}_{t-1}$, where the second inequality follows from the inductive hypothesis. Putting these two facts together, we obtain:

$$\mathrm{PUA}^{alg}_{\geq t} \leq \min\left\{\overline{P}_{t-1}, \eta_t + \textsc{Conf}^P_t(\eta_t)\right\} = \overline{P}_t$$

with probability at least $1 - \delta/(3t\log(T))$. A union bound over t completes the proof of the result. $\square$

Lemma 4.10 follows from these high-probability bounds. We defer its proof to Appendix B.5.

Lemma 4.10. *Let $\mathcal{E} = \mathcal{E}_N \cap \mathcal{E}_{\overline{P}} \cap \mathcal{E}_{oe}$. Then, $\mathbb{P}(\mathcal{E}) \geq 1 - 3\delta$.*

In the remainder of the proof, it suffices to restrict our attention to $\mathcal{E}$.

4.3.2 Feasibility of $\underline{X}$

We now show that, given $\mathcal{E}$, our algorithm never runs out of budget, and as a result always allocates $X_{t,\theta}^{alg} \in \{\underline{X}, \overline{X}\}$. Since $X_{t,\theta}^{alg} = X_{t,\theta'}^{alg}$ for all θ, θ', for ease of notation in the remainder of the proof we omit the dependence of $X_{t,\theta}^{alg}$ on θ.

Lemma 4.11. *Under event $\mathcal{E}$, $B_t^{alg} \geq N_{\geq t}\underline{X}$ for all $t \in [T]$.*

Proof of Lemma 4.11. By induction on t.
Base Case: $t = 1$. By definition:

$$\underline{X} \leq \frac{B - \overline{\Delta}(\underline{X})}{\overline{N}} \implies B \geq \overline{N}\underline{X} + \overline{\Delta}(\underline{X}) \geq N\underline{X},$$

where the final inequality follows from $\overline{N} \geq N$ under $\mathcal{E}$, and $\overline{\Delta}(\underline{X}) \geq 0$.
Step Case: $t-1 \to t$. We condition our analysis on $(X_\tau^{alg})_{\tau < t}$, the algorithm's previous allocations.
Case 1: $X_\tau^{alg} = \underline{X}$ for all $\tau < t$. By the recursive budget update, $B_t^{alg} = B - N_{<t}\underline{X} - \text{PUA}_{<t}^{alg}$, where $\text{PUA}_{<t}^{alg}$ denotes the quantity of unallocated goods that perished before the end of round t. To show that $B_t^{alg} \geq N_{\geq t}\underline{X}$, it then suffices to show that $B - \text{PUA}_{<t}^{alg} \geq N\underline{X}$. We have:

$$\text{PUA}_{<t}^{alg} \leq \text{PUA}_{\geq 1}^{alg} \leq \overline{P}_1 = \overline{\Delta}(\underline{X}),$$

where the final inequality follows from Lemma 4.9. Under $\mathcal{E}$, then, as in the base case:

$$B - \text{PUA}_{<t}^{alg} \geq B - \overline{\Delta}(\underline{X}) \geq N\underline{X}.$$

Case 2: There exists $\tau < t$ such that $X_\tau^{alg} = \overline{X}$. Let $t^* = \sup\{\tau < t : X_\tau^{alg} = \overline{X}\}$ be the most recent time the algorithm allocated $\overline{X}$. Again, by the recursive budget update:

$$B_t^{alg} = B_{t^*}^{alg} - N_{t^*}\overline{X} - N_{(t^*,t)}\underline{X} - \text{PUA}_{[t^*,t)}^{alg}.$$

Since $\overline{X}$ was allocated at t^*, it must have been that $B_{t^*}^{alg} \geq N_{t^*}\overline{X} + \overline{N}_{>t^*}\underline{X} + \overline{P}_{t^*}$. Plugging this into the above and simplifying:

$$\begin{aligned}
B_t^{alg} &\geq N_{t^*}\overline{X} + \overline{N}_{>t^*}\underline{X} + \overline{P}_{t^*} - N_{t^*}\overline{X} - N_{(t^*,t)}\underline{X} - \text{PUA}_{[t^*,t)}^{alg} \\
&= \overline{N}_{>t^*}\underline{X} - N_{(t^*,t)}\underline{X} + \overline{P}_{t^*} - \text{PUA}_{[t^*,t)}^{alg} \\
&\geq N_{\geq t}\underline{X} + \overline{P}_{t^*} - \text{PUA}_{[t^*,t)}^{alg},
\end{aligned}$$

where the second inequality follows from the fact that $\overline{N}_{>t^*} \geq N_{>t^*}$ under $\mathcal{E}$. Thus, it suffices to show that $\overline{P}_{t^*} \geq \text{PUA}_{[t^*,t)}^{alg}$. This holds since $\text{PUA}_{[t^*,t)}^{alg} \leq \text{PUA}_{\geq t^*}^{alg} \leq \overline{P}_{t^*}$ by Lemma 4.9. □

4.3.3 Improving efficiency via $\overline{X}$

Having established that the algorithm never runs out of budget, it remains to investigate the gains from allocating $\overline{X}$. By the threshold condition, whenever the algorithm allocates $\overline{X}$ it must be that there is enough budget remaining to allocate $\overline{X}$ in the current period, and $\underline{X}$ in all future periods, under high demand *and* high perishing. Thus, at a high level, $\overline{X}$ being allocated is an indication that the algorithm has been inefficient up until round t. The following lemma provides a lower bound on the last time the algorithm allocates $\underline{X}$. This lower bound will later on allow us to establish that, for most of the time horizon, the remaining budget is low relative to future demand, ensuring high efficiency.

Lemma 4.12. *Given $\mathcal{E}$, let $t_0 = \sup\{t : X_t^{alg} = \underline{X}\}$ be the last time that $X_t^{alg} = \underline{X}$ (or else 0 if the algorithm always allocates according to $\overline{X}$). Then, for some $\tilde{c} = \widetilde{\Theta}(1)$,*

$$t_0 > T - \tilde{c}\left(\frac{1}{L_T} + \sqrt{\frac{T\mathcal{L}^{\mathsf{perish}}}{L_T}}\right)^2.$$

We defer the proof of Lemma 4.12 to Appendix B.5. Observe that, as $\mathcal{L}^{\mathsf{perish}}$ increases, our algorithm stops allocating $\underline{X}$ earlier on. We will next see that this loss propagates to the our final efficiency bound.

4.3.4 Putting it all together

With these building blocks in hand, we prove our main result.

Proof of Theorem 4.2. By Lemma 4.11, the algorithm never runs out of budget under event $\mathcal{E}$, which occurs with probability at least $1 - 3\delta$. As a result $X_{t,\theta}^{alg} \in \{\underline{X}, \overline{X}\}$ for all $t \in [T]$, $\theta \in \Theta$. We use this to bound envy and efficiency.

Counterfactual Envy: Recall, $\Delta_{EF} = \max_{t,\theta} |w_\theta(X_{t,\theta}^{alg} - \frac{B}{N})| \leq w_{\max} \cdot |X_{t,\theta} - \frac{B}{N}|$. We consider two cases.

Case 1: $\underline{X} \leq \overline{X} \leq \frac{B}{N}$. By definition:

$$\frac{B}{N} - \underline{X} = \frac{B}{N} - \frac{B}{\overline{N}} + \frac{B}{\overline{N}} - \underline{X} \leq \frac{B}{\underline{N}} - \frac{B}{\overline{N}} + \mathcal{L}^{\mathsf{perish}},$$

where the inequality follows from the fact that $\underline{N} \leq N$ under $\mathcal{E}$, and $\mathcal{L}^{\mathsf{perish}} = B/\overline{N} - \underline{X}$ by definition. We turn our attention to the first two terms:

$$\begin{aligned}\frac{B}{\underline{N}} - \frac{B}{\overline{N}} &\leq \frac{B}{\mathbb{E}[N] - \textsc{Conf}_{0,T}^N} - \frac{B}{\mathbb{E}[N] + \textsc{Conf}_{0,T}^N} \\ &= \frac{B}{\mathbb{E}[N]}\left(\frac{1}{1 - \frac{\textsc{Conf}_{0,T}^N}{\mathbb{E}[N]}} - \frac{1}{1 + \frac{\textsc{Conf}_{0,T}^N}{\mathbb{E}[N]}}\right) = \beta_{avg}\left(\frac{1}{1 - \frac{\textsc{Conf}_{0,T}^N}{\mathbb{E}[N]}} - \frac{1}{1 + \frac{\textsc{Conf}_{0,T}^N}{\mathbb{E}[N]}}\right).\end{aligned}$$

Using the fact that $\textsc{Conf}_{0,T}^N = \sqrt{2T|\Theta|\rho_{\max}^2 \log(2T^2/\delta)}$ and $\mathbb{E}[N] = \Theta(T)$, there exists $c_1, c_2 = \widetilde{\Theta}(1)$ such that, for large enough T, $\left(1 - \frac{\textsc{Conf}_{0,T}^N}{\mathbb{E}[N]}\right)^{-1} \leq \left(1 - c_1/\sqrt{T}\right)^{-1} \leq 1 + 2c_1/\sqrt{T}$ and $\left(1 + \frac{\textsc{Conf}_{0,T}^N}{\mathbb{E}[N]}\right)^{-1} \geq \left(1 + c_2/\sqrt{T}\right)^{-1} \geq 1 - c_2/\sqrt{T}$. Plugging this into the above:

$$\frac{B}{\underline{N}} - \frac{B}{\overline{N}} \leq \beta_{avg}\left(1 + 2c_1/\sqrt{T} - (1 - c_2/\sqrt{T})\right) \leq \beta_{avg}(2c_1 + c_2)/\sqrt{T} \lesssim 1/\sqrt{T}.$$

Thus, we obtain $|X_{t,\theta}^{alg} - B/N| \lesssim 1/\sqrt{T} + \mathcal{L}^{\mathsf{perish}}$.

Case 2: $\underline{X} \leq \frac{B}{N} \leq \overline{X}$. We have: $|X_{t,\theta}^{alg} - B/N| = \max\{\frac{B}{N} - \underline{X}, \overline{X} - \frac{B}{N}\} \leq \overline{X} - \underline{X} = L_T$. Combining these two cases, we obtain $\Delta_{EF} \lesssim \max\{1/\sqrt{T} + \mathcal{L}^{\mathsf{perish}}, L_T\}$.

Hindsight Envy: $\textsc{Envy}$ is trivially bounded above by $w_{max} \cdot L_T \lesssim L_T$ since, for any t, t':

$$w_\theta(X_{t',\theta} - X_{t,\theta}) \leq w_{max}(\overline{X} - \underline{X}) = w_{max} L_T.$$

Efficiency: Let $t_0 = \sup\{t : X_t^{alg} = \underline{X}\}$. Then:

$$\begin{aligned}
\Delta_{efficiency} &= B - \sum_{t,\theta} N_{t,\theta} X_{t,\theta} = B - \sum_t N_t X_t^{alg} \\
&= B_{t_0} + \sum_{t<t_0} N_t X_t^{alg} + \mathrm{PUA}_{<t_0}^{alg} - \sum_t N_t X_t^{alg} \\
&= B_{t_0} - \sum_{t\geq t_0} N_t X_t^{alg} + \mathrm{PUA}_{<t_0}^{alg} \\
&< N_{t_0}\overline{X} + \overline{N}_{>t_0}\underline{X} + \overline{P}_{t_0} - N_{t_0}\underline{X} - N_{>t_0}\overline{X} + \mathrm{PUA}_{<t_0}^{alg} \\
&= \underline{X}(\overline{N}_{>t_0} - N_{>t_0}) - (\overline{X} - \underline{X})(N_{>t_0} - N_{t_0}) + \overline{P}_{t_0} + \mathrm{PUA}_{<t_0}^{alg},
\end{aligned}$$

where the inequality follows from $X_{t_0}^{alg} = \underline{X}$, and the threshold condition for allocating $\overline{X}$.

Noting that $\underline{X} \leq \beta_{avg}$ and $\overline{N}_{>t_0} - N_{>t_0} \leq 2\textsc{Conf}_{t_0,T}^N$, we have:

$$\begin{aligned}
\underline{X}(\overline{N}_{>t_0} - N_{>t_0}) &\leq \beta_{avg} \cdot 2\sqrt{2(T-t_0)|\Theta|\rho_{max}^2 \log(2T^2/\delta)} \\
&\leq 2\beta_{avg}\sqrt{2\tilde{c}|\Theta|\rho_{\max}^2 \log(2T^2/\delta)} \min\left\{\sqrt{T}, L_T^{-1} + \sqrt{TL_T^{-1}\mathcal{L}^{\mathsf{perish}}}\right\}, \qquad (14)
\end{aligned}$$

where the second inequality follows from Lemma 4.12.

We loosely upper bound the second term by:

$$-(\overline{X} - \underline{X})(N_{>t_0} - N_{t_0}) \leq (\overline{X} - \underline{X})N_{t_0} \leq L_T|\Theta|(\mu_{max} + \rho_{\max}). \qquad (15)$$

Finally, consider $\overline{P}_{t_0} + \mathrm{PUA}_{<t_0}^{alg}$. By construction, $\overline{P}_{t_0} \leq \overline{P}_1 = \overline{\Delta}(\underline{X})$. To upper bound $\mathrm{PUA}_{<t_0}^{alg}$, we consider the process that allocates $\underline{X}$ in each period to all arrivals. Let $B_t(\underline{X})$ denote the quantity of remaining items under this process, and $\mathcal{B}_t(\underline{X})$ the set of remaining items. We use $\mathrm{PUA}_t(\underline{X})$ to denote the quantity of unallocated items that perish at the end of period t under this process, and $\mathrm{PUA}_{<t}(\underline{X})$ those that perished before the end of period t. The following lemma allows us to tractably bound $\mathrm{PUA}_{<t_0}^{alg}$ via this process. We defer its proof to Appendix B.5.4.

Lemma 4.13. *For all $t \in [T]$,*

1. $\mathcal{B}_t^{alg} \subseteq \mathcal{B}_t(\underline{X})$
2. $\mathrm{PUA}_t^{alg} \leq \mathrm{PUA}_t(\underline{X})$.

Using these two facts, we have: $\mathrm{PUA}_{<t_0}^{alg} \leq \mathrm{PUA}_{<t_0}(\underline{X}) \leq \mathrm{PUA}_{\geq 1}(\underline{X}) \leq \overline{P}_1 = \overline{\Delta}(\underline{X})$. Hence,

$$\overline{P}_{t_0} + \mathrm{PUA}_{<t_0} \leq 2\overline{\Delta}(\underline{X}) \leq 2\overline{N}\mathcal{L}^{\mathsf{perish}} \leq 2\mu_{max}(1 + \sqrt{2|\Theta|\rho_{\max}^2 \log(2T^2/\delta)})T\mathcal{L}^{\mathsf{perish}}, \qquad (16)$$

where the second inequality follows from $\overline{\Delta}(\underline{X}) \leq B - \overline{N}\underline{X} = B - \overline{N}\left(\frac{B}{\overline{X}} - \mathcal{L}^{\mathsf{perish}}\right) = \overline{N}\mathcal{L}^{\mathsf{perish}}$, and the last inequality uses the definition of $\overline{N}$. Putting bounds (14), (15) and (16) together, we obtain:

$$\begin{aligned}
\Delta_{efficiency} \leq\ & 2\beta_{avg}\sqrt{2\tilde{c}|\Theta|\rho_{\max}^2 \log(2T^2/\delta)} \min\left\{\sqrt{T}, L_T^{-1} + \sqrt{TL_T^{-1}\mathcal{L}^{\mathsf{perish}}}\right\} + L_T|\Theta|(\mu_{max} + \rho_{\max}) \\
&+ 2\mu_{max}(1 + \sqrt{2|\Theta|\rho_{\max}^2 \log(2T^2/\delta)})T\mathcal{L}^{\mathsf{perish}}.
\end{aligned}$$

Using the fact that $L_T = o(1)$, we obtain the final bound on efficiency. $\square$

5 Numerical experiments

In this section we study the practical performance of PERISHING-GUARDRAIL via an extensive set of numerical experiments. We first consider one of the most popular (and aggressive) models of perishing: geometrically distributed perishing times. For this tractable perishing process, we establish distribution-dependent bounds on the σ-induced loss $\mathcal{L}^{\mathsf{perish}}$, and empirically explore the dependence of the envy-efficiency trade-off on the perishing rate. We leverage these empirical trade-off curves to provide guidance on how to select the envy parameter L_T, and compare the performance of PERISHING-GUARDRAIL to its perishing-agnostic counterpart (Sinclair et al., 2022). We moreover demonstrate the robustness of our algorithm on a real-world dataset on ginger perishability (Keskin et al., 2022). We conclude our numerical study by considering the non-i.i.d. perishing setting to gain insights into the choice of allocation schedule.

5.1 Geometric perishing

Consider the setting in which each available unit perishes independently with probability p in each period, i.e., $T_b \sim \text{Geometric}(p)$, for all $b \in \mathcal{B}$. Throughout the section, we assume $|\Theta| = 1$, as our insights are invariant to the number of types. Since perishing times are identically distributed, the allocation order σ does not have any impact on the performance of the algorithm; hence, in the remainder of this section we assume σ is the identity ordering.

5.1.1 Quantifying the unavoidable perishing loss

We first investigate the impact of perishability by characterizing the lower guardrail $\underline{X}$ as a function of the perishing rate p. As in Section 4.2, we instantiate our theoretical bounds assuming $B = T, N_t = 1$ for all $t \in [T]$. In this case, Proposition 4.6 implies that $p \leq \frac{1}{T}$ is necessary to guarantee δ-offset-expiry, for nontrivial values of δ. Proposition 5.1 below provides a lower bound on $\underline{X}$ for this setting. We defer its proof to Appendix B.6.

Proposition 5.1. *Suppose $T_b \sim \text{Geometric}(p)$ for all $b \in \mathcal{B}$, with $p \leq 1/T$. Then, the perishing process is δ-offset-expiring for any $\delta \geq 2\log T \cdot \frac{Tp}{(1-Tp)^2}$. Moreover, $\underline{X} \geq 1 - 3Tp - \frac{\log(3\log(T)/\delta)}{T}$ for any ordering σ.*

Proposition 5.1 establishes that, in the worst case, $\underline{X}$ decays *linearly* in the rate p at which goods spoil. This highlights the extent to which the commonly used (and practically pessimistic) geometric model limits both the kinds of perishable goods selected before allocation, as well as the rate at which a decision-maker can allocate goods. Letting $p = T^{-(1+\alpha)}$, $\alpha \in (0,1)$, Proposition 5.1 implies that $\mathcal{L}^{\mathsf{perish}}$ is on the order of $T^{-\alpha}$. Alternatively, if a decision-maker wants no more than $T^{-\alpha}$ loss relative to the proportional allocation, Proposition 5.1 provides an upper bound of $T^{-(1+\alpha)}$ on the (exogenous) rate at which goods spoil.

We validate the scaling of $\underline{X}$ numerically in Fig. 3, for an instance where the decision-maker also faces demand uncertainty. We observe that $\underline{X}$ is concave increasing in α, and that our lower bound on $\underline{X}$ in the idealized setting provides a good fit for $\underline{X}$, even under demand uncertainty. For α close to 0 (i.e., $p \sim 1/T$), $\underline{X} \approx 0.4$, less than half of the "naive" no-perishing allocation, $B/\overline{N}$. For $\alpha = 1$, $\underline{X} \approx B/\overline{N}$. Note that, for $\alpha > 1$, $\underline{X}$ is limited by the confidence bound $\log(3\log T/\delta)/T$ in Proposition 5.1. Plugging the lower bound on δ into this term, this implies that, even under no demand uncertainty, $\underline{X}$ incurs a loss on the order of $1/T$ relative to the proportional allocation.

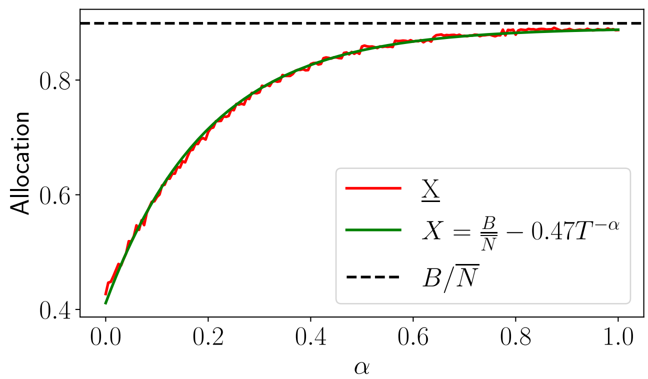
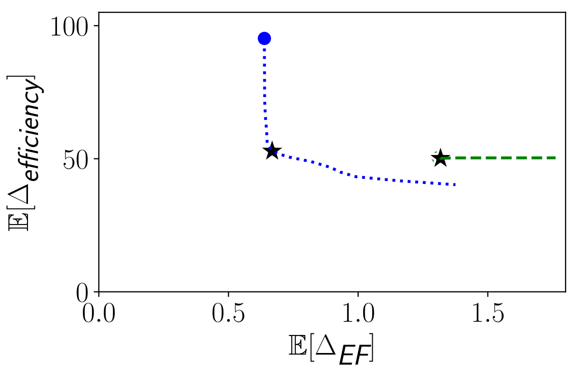
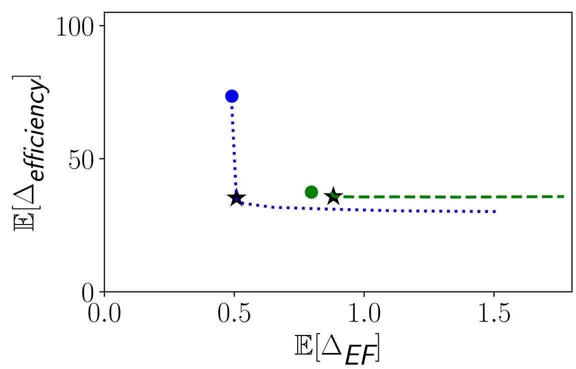


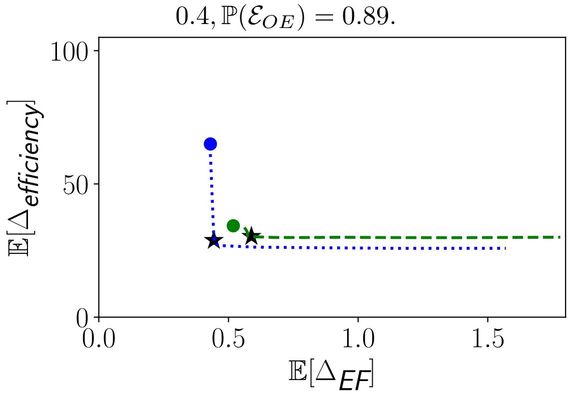
Figure 3: Maximum feasible allocation $\underline{X}$ vs. α, for $T_b \sim Geometric(T^{-(1+\alpha)})$, $B = 200$, $T = 150$, N_t drawn from a truncated normal distribution $\mathcal{N}(2, 0.25)$, and $\delta = 1/T$. Here, $\underline{X}$ was calculated via line search, with Monte Carlo simulation used to estimate $\overline{\Delta}(X)$ for each value of X. The dashed line represents the "naive" allocation $B/\overline{N} = 0.89$ which ignores possible perishing, and the green line is the curve of best fit to $\underline{X}$.



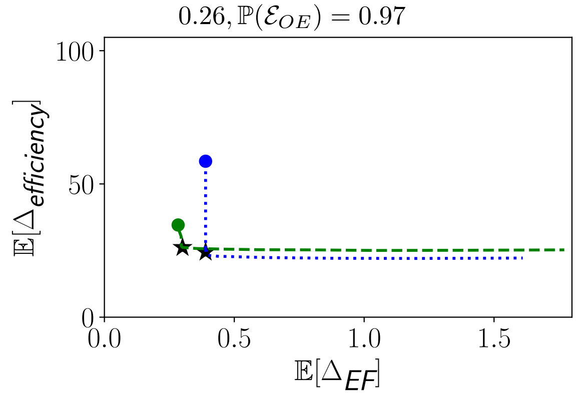
(a) $T_b \sim Geometric(T^{-1.1}) : \underline{X} = 0.70, \mathcal{L}^{\text{perish}} = 0.4, \mathbb{P}(\mathcal{E}_{OE}) = 0.89$.



(b) $T_b \sim Geometric(T^{-1.2}) : \underline{X} = 0.84, \mathcal{L}^{\text{perish}} = 0.26, \mathbb{P}(\mathcal{E}_{OE}) = 0.97$



(c) $T_b \sim Geometric(T^{-1.25}) : \underline{X} = 0.89, \mathcal{L}^{\text{perish}} = 0.21, \mathbb{P}(\mathcal{E}_{OE}) = 0.99$



(d) $T_b \sim Geometric(T^{-1.3}) : \underline{X} = 0.93, \mathcal{L}^{\text{perish}} = 0.17, \mathbb{P}(\mathcal{E}_{OE}) = 0.99$

Figure 4: Empirical trade-off between $\mathbb{E}[\Delta_{efficiency}]$ and $\mathbb{E}[\Delta_{EF}]$. The points on the trade-off curve correspond to increasing values of L_T, from left to right. Static-$\frac{B}{N}$ and Static-$\underline{X}$ respectively correspond to Vanilla-Guardrail and Perishing-Guardrail for $L_T = 0$.

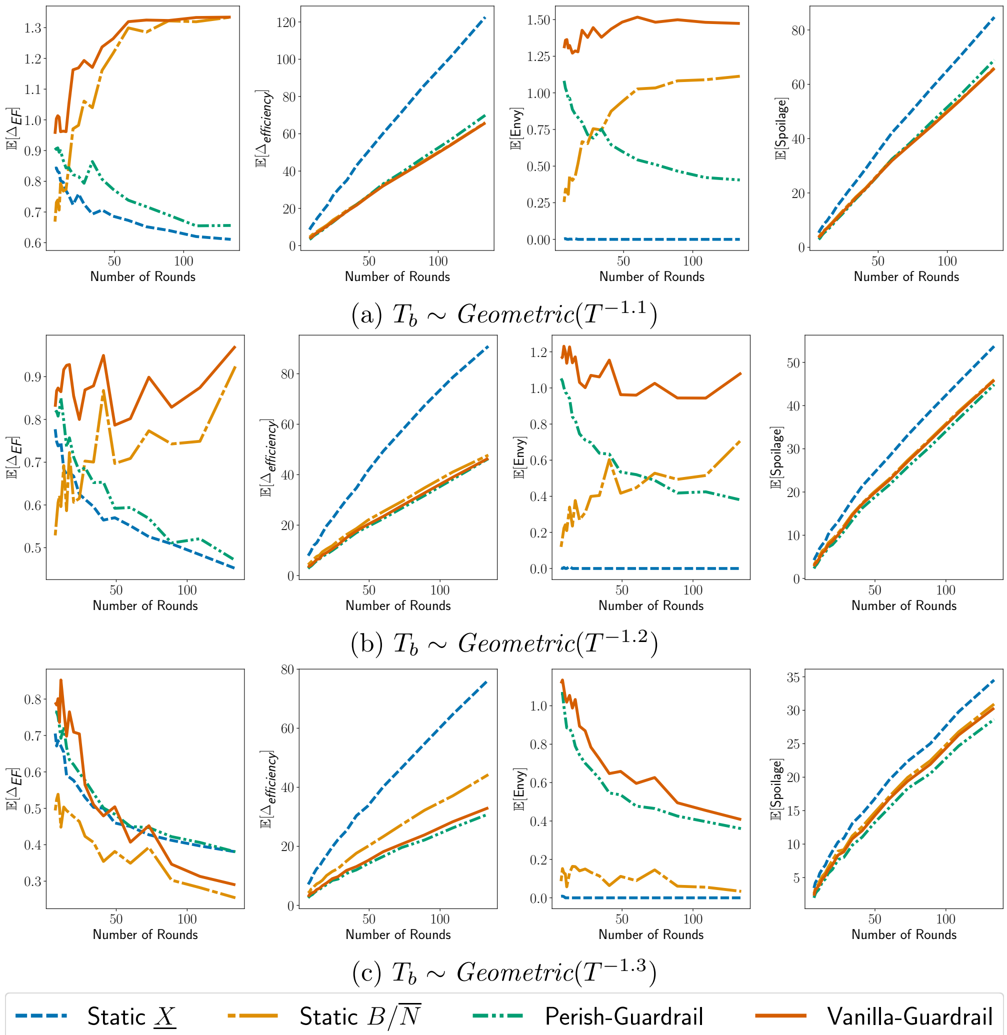
5.1.2 Numerical performance of Perishing-Guardrail

Empirical Trade-off Curves. We numerically investigate the impact of the perishing rate on the envy-efficiency frontier, and use the empirical trade-off curve to provide guidance on how decision-makers should select L_T in the setting of geometric i.i.d. perishing. The demands N_t are drawn from a truncated normal distribution $\mathcal{N}(2, 0.25)$. We let $T = 100$, $B = 200$, and vary $\alpha \in \{0.1, 0.2, 0.25, 0.3\}$ (see Appendix D, Fig. 7 for additional values of α). For these instances, we compare Perishing-Guardrail and Vanilla-Guardrail (Sinclair et al., 2022), a guardrail-based algorithm designed for settings without perishable resources, with $L_T = T^{-\beta}$, $\beta \in \{0, 0.05, 0.1, \ldots, 1\}$. All results are averaged over 150 replications.

The empirical trade-off curves can be found in Fig. 4. For $\alpha \in \{0.1, 0.2\}$, Vanilla-Guardrail makes close to no gains in efficiency for any value of L_T. In these high-perishing settings, then, setting $L_T = 0$ is optimal (there is no trade-off), in stark contrast to the classic setting without perishability. As α increases, Vanilla-Guardrail attains small gains in efficiency, but plateaus very quickly. This yields the important insight that **perishing-agnostic algorithms are *not* able to leverage unfairness to improve efficiency in the presence of perishable resources.** Perishing-Guardrail, on the other hand, sees extremely large gains in efficiency for a very small increase in envy, across all values of α. Even for this algorithm, however, larger values of L_T do not provide marginal gains in efficiency; the horizontal asymptote observed across all values of α is precisely the cumulative σ-induced loss, $T\mathcal{L}^{\mathsf{perish}}$. Moreover, the vertical asymptote across all plots corresponds to the unavoidable loss due to demand uncertainty (Theorem 3.7).

Note that, for small values of α, the Perishing-Guardrail empirical trade-off curve lies to the left of that of Vanilla-Guardrail, i.e., it achieves lower counterfactual envy across all values of L_T. As we will see below, this is due to the fact that Vanilla-Guardrail achieves extremely high stockout rates. This effect is diminished as α increases (i.e., the perishing rate decreases). As this happens, both curves move down and to the left (and closer) as they achieve lower counterfactual envy and inefficiency due to spoilage. When the perishing rate is negligible (largest value of α), the empirical trade-off curve of Vanilla-Guardrail is slightly to the left of that of Perishing-Guardrail; this is due to the loss incurred by our modified $\underline{X}$ construction, which always allocates less than $B/\overline{N}$ as a baseline. However, even when perishing is negligible Perishing-Guardrail is slightly more efficient that Vanilla-Guardrail, despite its baseline allocation being lower. This runs counter to the intuition that Vanilla-Guardrail should be more efficient since it has a higher baseline allocation. The reason for this is the difference in the two algorithms' threshold allocation decisions. Our algorithm, Perishing-Guardrail, allocates $\overline{X} = \underline{X} + L_T$ if it forecasts that it has enough budget remaining to allocate $\overline{X}$ in period t, and $\underline{X}$ onwards. On the other hand, Vanilla-Guardrail allocates $B/\overline{N} + L_T$ if it has enough budget remaining to allocate this high amount in period t, and $B/\overline{N}$ in all future periods. Since $B/\overline{N} > \underline{X}$, Vanilla-Guardrail depletes its budget faster than Perishing-Guardrail whenever they both allocate the lower guardrail. Hence, Perishing-Guardrail is able to allocate aggressively more frequently than Vanilla-Guardrail, which results in improved efficiency and decreased spoilage.

These empirical trade-off curves help to provide guidance on the choice of L_T. In particular, across all experiments, the cusp of the trade-off curve lies at $L_T \sim T^{-0.35}$. This value of L_T is *larger* than no-perishing cusp of $L_T \sim T^{-1/2}$ (Sinclair et al., 2022). This is due to the fact that our baseline allocation is significantly lower to avoid perishing-induced stockouts; hence, in order to recover from this inefficiency, L_T must be higher. We use this observation in the following experiments, comparing the performance of Perishing-Guardrail and Vanilla-Guardrail for $L_T \sim T^{-0.35}$.



(a) $T_b \sim \textit{Geometric}(T^{-1.1})$

(b) $T_b \sim \textit{Geometric}(T^{-1.2})$

(c) $T_b \sim \textit{Geometric}(T^{-1.3})$

Figure 5: Algorithm comparison across $\mathbb{E}[\Delta_{EF}], \mathbb{E}[\Delta_{\textit{efficiency}}], \mathbb{E}[\textsc{Envy}]$, and $\mathbb{E}[\text{Spoilage}]$, for $\alpha \in \{0.1, 0.2, 0.25, 0.3\}$.

Perishing-Guardrail performance: synthetic experiments. We compare the performance of Perishing-Guardrail to three benchmark algorithms:

- Vanilla-Guardrail (Sinclair et al. (2022));
- Static-$\underline{X}$, the algorithm which allocates $X_{t,\theta} = \underline{X}$ for all t, θ, until it runs out of resources;
- Static-$\frac{B}{N}$, the algorithm which allocates $X_{t,\theta} = \frac{B}{N}$ for all t, θ, until it runs out of resources.

We are interested in five metrics: (i) expected counterfactual envy $\mathbb{E}[\Delta_{EF}]$, (ii) expected hindsight envy $\mathbb{E}[\textsc{Envy}]$, (iii) expected inefficiency $\mathbb{E}[\Delta_{\textit{efficiency}}]$, (iv) expected spoilage $\mathbb{E}[\text{PUA}_{\geq 1}]$, and (v) the stockout probability $\mathbb{E}[\textsc{Stockout}]$, i.e. the proportion of replications for which the algorithm runs out of resources before the end of the time horizon. We use the same simulation setup as above, with $B = 2T$ (see Appendix D, Fig. 8 for additional values of α). Fig. 5 illustrates the performance of each algorithm across the first four metrics of interest. We identify three regimes:

- **High Perishing** ($\alpha = 0.1$): While unfairness (as measured by $\mathbb{E}[\Delta_{EF}]$ and $\mathbb{E}[\textsc{Envy}]$) is decreasing in T under Perishing-Guardrail and Static-$\underline{X}$, Vanilla-Guardrail and

α	STATIC-$\frac{B}{\overline{N}}$	STATIC-$\underline{X}$	VANILLA-GUARDRAIL	PERISHING-GUARDRAIL
.1	0.99 ± 0.020	0.00 ± 0.0	1.00 ± 0.0	0.11 ± 0.06
.2	0.63 ± 0.095	0.00 ± 0.0	0.68 ± 0.091	0.03 ± 0.03
.3	0.03 ± 0.037	0.00 ± 0.0	0.06 ± 0.046	0.00 ± 0.0

Table 1: Comparison of stockout probabilities, for $T = 150$, $T_b \sim \text{Geometric}(T^{-(1+\alpha)})$. The second number in each cell corresponds to 95% confidence intervals.

Algorithm	$\mathbb{E}[\Delta_{EF}]$	$\mathbb{E}[\text{ENVY}]$	$\mathbb{E}[\text{SPOILAGE}]$	$\mathbb{E}[\text{STOCKOUT}]$	$\mathbb{E}[\Delta_{efficiency}]$
STATIC-$\frac{B}{\overline{N}}$	1.18 ± 0.01	1.03 ± 0.0	346.4 ± 2.6	1.0 ± 0.0	444.6 ± 2.7
STATIC-$\underline{X}$	0.60 ± 0.01	0.0 ± 0.0	475.9 ± 2.7	0 ± 0.0	605.5 ± 3.0
VANILLA-GUARDRAIL	1.17 ± 0.01	1.44 ± 0.0	341.4 ± 3.0	1.0 ± 0.0	343.5 ± 2.9
PERISHING-GUARDRAIL	0.78 ± 0.04	0.42 ± 0.05	372.2 ± 3.2	0.39 ± 0.09	372.7 ± 3.2

Table 2: Performance of the different algorithms (for $L_T = T^{-0.35}$) on the "ginger" dataset in Keskin et al. (2022). The second number in each cell corresponds to 95% confidence intervals.

STATIC-$\frac{B}{\overline{N}}$ perform remarkably poorly along these two metrics. This is due to the fact that these latter algorithms fail to account for the unavoidable perishing loss, resulting in an extremely high stockout probability, as illustrated in Table 1, for $T = 150$. In contrast, the two perishing-aware algorithms rarely run out of resources. This underscores the importance of modifying the baseline guardrail $\underline{X}$, which was specifically constructed to avoid stockouts due to unavoidable perishing.

Comparing STATIC-$\underline{X}$ to PERISHING-GUARDRAIL, our results also demonstrate that, in this high-perishing regime, the strategy of cautiously over-allocating by L_T comes at a significant reduction in inefficiency $\mathbb{E}[\Delta_{efficiency}]$, at close to no increase in counterfactual envy $\mathbb{E}[\Delta_{EF}]$.

- **Medium Perishing** ($\alpha = 0.2$): Though we observe similar trends as when $\alpha = 0.1$, all algorithms perform better across the board. Still, in Table 1 we see that the perishing-agnostic algorithms run out of resources in over 50% of replications. As observed in Fig. 3, PERISHING-GUARDRAIL exhibits both higher efficiency and lower spoilage than its perishing-agnostic counterpart in this regime, since it satisfies the threshold condition more frequently, as described above.

- **Low Perishing** ($\alpha = 0.3$): For this smaller perishing rate, VANILLA-GUARDRAIL stocks out significantly less frequently. Putting this together with the fact that $B/\overline{N} > \underline{X}$, this explains the fact that it has lower counterfactual envy than PERISHING-GUARDRAIL. However, along all other metrics PERISHING-GUARDRAIL improves upon VANILLA-GUARDRAIL. The improvements in efficiency and spoilage are due to the same effects as described above; moreover, our algorithm improves upon VANILLA-GUARDRAIL on $\mathbb{E}[\text{ENVY}]$ since it never stocks out.

Overall, our results highlight the robustness of PERISHING-GUARDRAIL to perishability, as it is able to achieve similar if not improved performance as VANILLA-GUARDRAIL in settings where there is limited perishing, with vastly superior performance in high-perishing settings.

Real-World Instance. We next investigate the performance of our algorithm using the "ginger" dataset provided by Keskin et al. (2022), which tracks demand, replenishments, and perishing of ginger across $T = 365$ days. We treat the time between each replenishment as an independent sample (102 samples in total), and fit a geometric distribution to the quantity of goods that perish in each sample, obtaining $p = 0.00224$. We similarly fit a truncated normal distribution to the dataset, with $\mathcal{N}(3.2, 1.85)$. Finally, we let $B = 365 \cdot 3.2 = 1168$. For these inputs, $B/\overline{N} = 0.89$ and $\underline{X} = 0.46$. Under these parameters the offset-expiry condition is only satisfied 65.2% of the time; given this aggressive perishing, perishing-agnostic algorithms are expected to perform particularly poorly.

In Table 2 we compare the performance of the different algorithms. We observe the following:

- As conjectured, Static-$\frac{B}{\overline{N}}$ and Vanilla-Guardrail stock out on 100% of replications since they fail to account for endogenous perishing. The high stockout probabilities of Static-$\frac{B}{\overline{N}}$ and Vanilla-Guardrail lead to high unfairness (vis-à-vis hindsight and counterfactual envy), since later arrivals receive allocations of zero. In contrast, Static-$\underline{X}$ never stocks out. Perishing-Guardrail achieves a higher stockout rate of 40%, likely due to the fact that threshold condition does not account for non-offset-expiring trajectories. Still, our algorithm's counterfactual envy and hindsight envy are over 30% and 70% lower, respectively, than that of Vanilla-Guardrail.

- Perishing-Guardrail allocates approximately 10% fewer goods than Vanilla-Guardrail. It is notable, however, that it is more efficient than Static-$\frac{B}{\overline{N}}$; this highlights that naively allocating more aggressively need not always generate gains.

Overall, we see that even when offset-expiry holds with much lower probability, for small losses in efficiency our algorithm makes major gains in fairness relative to perishing-agnostic algorithms.

5.2 Non-i.i.d. perishing

As seen in Section 3, the performance of any algorithm is a function of $\mathcal{L}^{\mathsf{perish}}$, which depends on the allocation order σ in non-i.i.d. settings. A natural question that arises, then, is how a decision-maker should choose σ. In this section, we investigate the impact of three practical allocation orders on the numerical performance of Perishing-Guardrail.

Given Theorem 3.7, a reasonable allocation order to choose would be $\sigma^* \in \arg\min_\sigma \mathcal{L}^{\mathsf{perish}}(\sigma)$, where we emphasize the dependence of $\mathcal{L}^{\mathsf{perish}}$ on the order σ. Computing such a σ^*, however, is infeasible given the space of $N!$ orderings. In lieu of this, we identify sufficient conditions on the perishing process and allocation order that guarantee that $\mathcal{L}^{\mathsf{perish}}$ remain low (equivalently, that $\underline{X}$ remain high), and use these conditions to identify practical and easily interpretable allocation schedules. Proposition 5.2 below provides these sufficient conditions, for $B = T, N_t = 1$ for all $t \in [T]$. We defer the proof to Appendix B.7.

Proposition 5.2. *Suppose there exists $\alpha \in (0,1)$ such that:*

1. $\mathbb{E}[T_b] > \min\left\{T, \lceil \frac{\sigma(b)}{1-T^{-\alpha}} \rceil\right\}$
2. $\sum_b \frac{\mathrm{Var}[T_b]}{\left(\mathbb{E}[T_b] - \min\{T, \lceil \frac{\sigma(b)}{1-T^{-\alpha}} \rceil\}\right)^2} \le \frac{1}{2}T^{1-\alpha}$

Then, for any $\delta \ge 3\log(T)e^{-\frac{1}{8}T^{1-\alpha}}$, the process is δ-offset-expiring, and $\underline{X} \ge 1 - T^{-\alpha}$.

Order	$\mathbb{E}\left[\mathcal{L}^{\mathsf{perish}}\right]$	$\mathbb{E}[\Delta_{EF}]$	$\mathbb{E}[\text{Envy}]$	$\mathbb{E}[\text{Spoilage}]$	$\mathbb{E}[\text{Stockout}]$	$\mathbb{E}[\Delta_{\mathit{efficiency}}]$
Increasing Mean	0.10 ± 0.004	0.36 ± 0.02	0.51 ± 0.02	5.78 ± 0.4	0.04 ± 0.04	6.28 ± 0.4
Decreasing CV / Increasing LCB	0.0 ± 0.0	0.44 ± 0.02	0.48 ± 0.02	1.24 ± 0.1	0.06 ± 0.04	1.79 ± 0.1

Table 3: Performance of Perishing-Guardrail for $L_T = T^{-0.35}$ on the distributions given in Eq. (17).

Proposition 5.2 highlights the two key components that determine the baseline allocation: the distance between the expected perishing time $\mathbb{E}[T_b]$ and the expected allocation time $\min\{T, \lceil \frac{\sigma(b)}{1-T^{-\alpha}} \rceil\}$ (which we colloquially refer to as "room to breathe"), and the variance of the perishing time. Specifically, Condition 1 implies that, if $\mathbb{E}[T_b]$ is low, it must be that the item is allocated early on in the horizon (i.e., $\sigma(b)$ is low). This encodes the "race against time" intuition that is typically held around perishing. Condition 2 can be viewed as an upper bound on the cumulative adjusted *coefficient of variation* (CV) of the perishing process. High-variance perishing times and smaller "room to breathe" push α down, resulting in a lower allocation rate. Hence, to guarantee a high allocation rate, the perishing process needs to satisfy one of two conditions: (1) low variability, or (2) high room to breathe. Having identified these driving factors, we compare the following candidate allocation orders in experiments:

- **Increasing Mean**: Increasing order of $\mathbb{E}[T_b]$;
- **Decreasing Coefficient of Variation** (CV): Decreasing order of $\mathrm{Std}[T_b]/\mathbb{E}[T_b]$. For fixed expected perishing time, this schedule allocates high-variance units earlier on. Conversely, for fixed variance, it allocates items according to the **Increasing Mean** schedule;
- **Increasing Lower Confidence Bound** (LCB): Increasing order of $\mathbb{E}[T_b] - 1.96\mathrm{Std}[T_b]$. This ordering allocates items according to the lower bound of the 95% confidence interval of the normal approximation to its perishing time. This lower bound is expected to be small if either the expected perishing time is small or the variance is large.

We break ties randomly in all cases.

As in Section 5.1, we draw the demands N_t from a truncated normal distribution, $\mathcal{N}(2, 0.25)$; we moreover let $T = 50$, $B = 100$, $\delta = \frac{1}{T}$, and $L_T = T^{-0.35}$. Finally, we consider two sets of perishing distributions:

- **Instance 1: Front-loaded variability**

$$T_b = \begin{cases} \text{Uniform}(T/2 - b/2, T/2 + b/2) & b \leq T \\ T & b > T \end{cases} \tag{17}$$

- **Instance 2: Back-loaded variability**

$$T_b = \begin{cases} b + 1 & b \leq T \\ \text{Uniform}(b+1, T) & b > T \end{cases} \tag{18}$$

It can easily be verified that both instances are δ-offset-expiring, for $\delta = 0.05$. Tables 3 and 4 show the performance of our algorithm over these instances.

Order	$\mathbb{E}\left[\mathcal{L}^{\mathsf{perish}}\right]$	$\mathbb{E}[\Delta_{EF}]$	$\mathbb{E}[\textsc{Envy}]$	$\mathbb{E}[\textsc{Spoilage}]$	$\mathbb{E}[\textsc{Stockout}]$	$\mathbb{E}[\Delta_{\textit{efficiency}}]$
Increasing Mean / LCB	0.0 ± 0.0	0.41 ± 0.02	0.46 ± 0.02	0.0 ± 0.0	0.1 ± 0.05	0.56 ± 0.07
Decreasing CV	0.47 ± 0.0	0.51 ± 0.05	0.48 ± 0.02	48.3 ± 0.08	0.01 ± 0.0	48.7 ± 0.07

Table 4: Performance of Perishing-Guardrail for $L_T = T^{-0.35}$ on the distributions given in Eq. (18).

For the first instance, the **Increasing Mean** schedule allocates the first T items uniformly at random, ignoring the fact that, for $b \leq T$, as b increases the item is more likely to perish earlier on in the horizon. The **Decreasing CV / Increasing LCB** schedules, on the other hand, are identical: they allocate the first T resources in decreasing order of b, and allocate the remaining uniformly at random. Notably, the **Decreasing CV / Increasing LCB** order achieves $\mathbb{E}\left[\mathcal{L}^{\mathsf{perish}}\right] = 0$, i.e., $\underline{X} = B/\overline{N}$, as in the no-perishing setting. (Note that $\mathcal{L}^{\mathsf{perish}} = 0$ implies that this is an optimal ordering.) Since its baseline allocation is higher it results in 78% less spoilage than the **Increasing Mean** order, and a 71% decrease in inefficiency. However, this order performs slightly worse with respect to counterfactual envy and stockouts: this is again due to the more aggressive allocations.

For the second instance, the **Increasing Mean** and **Increasing LCB** schedules are identical: they allocate items lexicographically. The **Decreasing CV** schedule, on the other hand, allocates the last T items (in increasing order of b) before the first T resources, since $\mathrm{Std}[T_b] = 0$ for all $b \leq T$. In this setting, the first schedule is optimal with respect to σ-induced loss, with $\mathbb{E}\left[\mathcal{L}^{\mathsf{perish}}\right] = 0$. This more aggressive allocation results in a 10% stockout rate (versus 1% for the **Decreasing CV** schedule), but outperforms the **Decreasing CV** order across all other metrics. This is intuitive as the number of errors in this latter, clearly bad order results in $\mathbb{E}\left[\mathcal{L}^{\mathsf{perish}}\right] = 0.47$, approximately 50% of the baseline allocation $B/\overline{N}$. The algorithm then incurs both high inefficiency and spoilage.

These results indicate that the **Increasing LCB** schedule is both a practical and robust candidate allocation order as it hedges against the inherent variability of the perishing process.

6 Conclusion

This paper considers a practically motivated variant of the canonical problem of online fair allocation wherein a decision-maker has a budget of *perishable* resources to allocate *fairly* and *efficiently* over a fixed time horizon. Our main insight is that perishability fundamentally impacts the envy-efficiency trade-off derived for the no-perishing setting: while a decision-maker can arbitrarily sacrifice on envy in favor of efficiency in this latter setting, this is no longer the case when there is uncertainty around items' perishing times. We derive strong lower bounds to formalize this insight, which are a function of both the quality of the decision-maker's prediction over perishing times, as well as the inherent aggressiveness of the perishing process. We moreover design an algorithm that achieves these lower bounds; this algorithm relies on the construction of a baseline allocation that accounts for the unavoidable spoilage incurred by any online algorithm. From a technical perspective, the main challenge that the perishing setting presents is that the uncertainty around the quantity of resources that spoil in the future is *endogenous*, in contrast to the *exogenous* uncertainty on the number of arrivals in the classical setting. Deriving tight bounds on spoilage (both for our lower bounds as well as in the design of our algorithm) relied on the "slow allocation" construction, which rendered the highly coupled process amenable to tractable analysis. Finally, our numerical experiments demonstrate our algorithm's strong performance against state-of-the-art *perishing-agnostic* benchmarks.

In terms of future directions, our work identified offset-expiry as a necessary condition for which the classical notion of envy-freeness is even meaningful. While our algorithm performs well numerically in "low-probability" offset-expiring settings, relaxing this assumption in theory remains an interesting open question. Though we conjecture that the slow allocation construction will remain a useful tool in more aggressive perishing settings, the philosophical question of how to define more appropriate notions of envy is likely the more important one. In addition to this, our model assumes that the decision-maker allocates items according to a fixed allocation schedule. Though our results do not require that the perishing distribution be memoryless, allowing for time-varying / adaptive allocation schedules, though less practical, would improve our algorithm's performance in non-memoryless settings. This relates back to the question of deriving theoretical insights into the structure of optimal allocation schedules. Finally, though this paper considered exogenous *depletion* of the budget, a natural practical extension is one wherein B evolves stochastically, accounting for external donations independent of the allocations made by the algorithm.

Acknowledgments

The authors would like to thank Connor Lawless for insightful conversations about this work. Part of this work was done while Sean Sinclair and Sid Banerjee were visiting the Simons Institute for the Theory of Computing for the semester on Data-Driven Decision Processes. We gratefully acknowledge funding from the National Science Foundation under grants ECCS-1847393, DMS-1839346, CCF-1948256, CNS-195599, and CNS-1955997, the Air Force Office of Scientific Research under grant FA9550-23-1-0068, and the Army Research Laboratory under grants W911NF-19-1-0217 and W911NF-17-1-0094.

References

Aleksandrov, M., Aziz, H., Gaspers, S., and Walsh, T. (2015). Online fair division: Analysing a food bank problem. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 2540–2546. AAAI Press.

Aleksandrov, M. and Walsh, T. (2019a). Monotone and online fair division. In *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*, pages 60–75. Springer.

Aleksandrov, M. and Walsh, T. (2019b). Online fair division: A survey. *arXiv preprint arXiv:1911.09488*.

Aristotle Cloud Federation Project (2022). https://www.hpcwire.com/off-the-wire/aristotle-project-advances-campus-cloud-technologies/.

Azar, Y., Buchbinder, N., and Jain, K. (2010). How to allocate goods in an online market? In *European Symposium on Algorithms*, pages 51–62. Springer.

Aziz, H., Schlotter, I. A., and Walsh, T. (2016). Control of fair division. IJCAI.

Bakker, M., Riezebos, J., and Teunter, R. H. (2012). Review of inventory systems with deterioration since 2001. *European Journal of Operational Research*, 221(2):275–284.

Banerjee, S., Gkatzelis, V., Gorokh, A., and Jin, B. (2020). Online nash social welfare maximization via promised utilities. *arXiv preprint arXiv:2008.03564*.

Bansal, N., Jiang, H., Singla, S., and Sinha, M. (2020). *Online Vector Balancing and Geometric Discrepancy*, page 1139–1152. Association for Computing Machinery, New York, NY, USA.

Bateni, M., Chen, Y., Ciocan, D. F., and Mirrokni, V. (2022). Fair resource allocation in a volatile marketplace. *Operations Research*, 70(1):288–308.

Benade, G., Kazachkov, A. M., Procaccia, A. D., and Psomas, C.-A. (2018). How to make envy vanish over time. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 593–610.

Bertsimas, D., Farias, V. F., and Trichakis, N. (2011). The price of fairness. *Operations Research*, 59(1):17–31.

Bogomolnaia, A., Moulin, H., and Sandomirskiy, F. (2022). On the fair division of a random object. *Management Science*, 68(2):1174–1194.

Chen, Q., Golrezaei, N., and Susan, F. (2022). Fair assortment planning. *arXiv preprint arXiv:2208.07341*.

Cohen, M. C., Elmachtoub, A. N., and Lei, X. (2022). Price discrimination with fairness constraints. *Management Science*, 68(12):8536–8552.

Cole, R., Gkatzelis, V., and Goel, G. (2013). Mechanism design for fair division: Allocating divisible items without payments. In *EC 2013*.

den Boer, A. V., Jansen, H. M., and Zhao, J. (2022). Waste reduction of perishable products through markdowns at expiry dates. *Available at SSRN*.

Donahue, K. and Kleinberg, J. (2020). Fairness and utilization in allocating resources with uncertain demand. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 658–668.

Elzayn, H., Jabbari, S., Jung, C., Kearns, M., Neel, S., Roth, A., and Schutzman, Z. (2019). Fair algorithms for learning in allocation problems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 170–179.

Feeding America (2022). Feeding America 2022 Annual Impact Report.

Freeman, R., Zahedi, S. M., and Conitzer, V. (2017). Fair social choice in dynamic settings. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4580–4587. International Joint Conferences on Artificial Intelligence Marina del Rey, CA.

Freund, D. and Hssaine, C. (2021). Fair incentives for repeated engagement. *arXiv preprint arXiv:2111.00002*.

Friedman, E., Psomas, C.-A., and Vardi, S. (2015). Dynamic fair division with minimal disruptions. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, EC '15, page 697–713, New York, NY, USA. Association for Computing Machinery.

Friedman, E., Psomas, C.-A., and Vardi, S. (2017). Controlled dynamic fair division. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, EC '17, page 461–478, New York, NY, USA. Association for Computing Machinery.

Gerding, E., Perez-Diaz, A., Aziz, H., Gaspers, S., Marcu, A., Mattei, N., and Walsh, T. (2019). Fair online allocation of perishable goods and its application to electric vehicle charging.

Ghodsi, A., Zaharia, M., Hindman, B., Konwinski, A., Shenker, S., and Stoica, I. (2011). Dominant resource fairness: Fair allocation of multiple resource types. In *Nsdi*, volume 11, pages 24–24.

Godoy, M. (2023). Millions of American families struggle to get food on the table, report finds.

Gupta, S. and Kamble, V. (2021). Individual fairness in hindsight. *The Journal of Machine Learning Research*, 22(1):6386–6420.

Hanukov, G., Avinadav, T., Chernonog, T., and Yechiali, U. (2020). A service system with perishable products where customers are either fastidious or strategic. *International Journal of Production Economics*, 228:107696.

Hassanzadeh, P., Kreacic, E., Zeng, S., Xiao, Y., and Ganesh, S. (2023). Sequential fair resource allocation under a markov decision process framework. *arXiv preprint arXiv:2301.03758*.

He, J., Procaccia, A. D., Psomas, A., and Zeng, D. (2019). Achieving a fairer future by changing the past. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 343–349. International Joint Conferences on Artificial Intelligence Organization.

Kalinowski, T., Narodytska, N., and Walsh, T. (2013). A social welfare optimal sequential allocation procedure. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

Keskin, N. B., Li, Y., and Song, J.-S. (2022). Data-driven dynamic pricing and ordering with perishable inventory in a changing environment. *Management Science*, 68(3):1938–1958.

Lien, R. W., Iravani, S. M., and Smilowitz, K. R. (2014). Sequential resource allocation for nonprofit operations. *Operations Research*, 62(2):301–317.

Los Angeles Regional Food Bank (2022a). Los Angeles Regional Food Bank 2022 Annual Impact Report.

Los Angeles Regional Food Bank (2022b). Why is Los Angeles Tossing Food for its Homeless Population in Dumpsters.

Manshadi, V., Niazadeh, R., and Rodilitz, S. (2023). Fair dynamic rationing. *Management Science*.

Mattei, N., Saffidine, A., and Walsh, T. (2017). Mechanisms for online organ matching. In *IJCAI*, pages 345–351.

Mattei, N., Saffidine, A., and Walsh, T. (2018). Fairness in deceased organ matching. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 236–242.

Mitzenmacher, M. and Upfal, E. (2017). *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press.

Nahmias, S. (2011). *Perishable inventory systems*, volume 160. Springer Science & Business Media.

Perry, D. (1999). Analysis of a sampling control scheme for a perishable inventory system. *Operations Research*, 47(6):966–973.

Salem, J. and Gupta, S. (2023). Secretary problems with biased evaluations using partial ordinal information. *Management Science*.

Sengul Orgut, I. and Lodree, E. J. (2023). Equitable distribution of perishable items in a food bank supply chain. *Production and Operations Management*, 32(10):3002–3021.

Sinclair, S. R., Jain, G., Banerjee, S., and Yu, C. L. (2022). Sequential fair allocation: Achieving the optimal envy-efficiency trade-off curve. *Operations Research*.

Varian, H. R. (1974). Equity, envy, and efficiency. *Journal of Economic Theory*, 9(1):63–91.

Varian, H. R. (1976). Two problems in the theory of fairness. *Journal of Public Economics*, 5(3-4):249–260.

Vershynin, R. (2018). *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press.

Zeng, D. and Psomas, A. (2020). Fairness-efficiency tradeoffs in dynamic fair division. In *Proceedings of the 21st ACM Conference on Economics and Computation*, EC '20, page 911–912, New York, NY, USA. Association for Computing Machinery.

A Table of notation

Symbol	Definition
	Problem setting specifications
T	Total number of rounds
$B, \mathcal{B}$	Number of resources and set of resources available
Θ, θ	Set of types for individuals, and specification for individual's type
w_θ, w_{max}	Preference for the resource of individuals of type θ and $w_{max} = \max_\theta w_\theta$
$N_{t,\theta}$	Number of individuals of type θ in round t
N_t	$\sum_{\theta \in \Theta} N_{t,\theta}$
$N_{\geq t}$	$\sum_{t' \geq t} N_{t'}$
$\sigma^2_{t,\theta}, \rho_{t,\theta}, \mu_{t,\theta}$	$\mathrm{Var}[N_{t,\theta}]$, bound on $\lvert N_{t,\theta} - \mathbb{E}[N_{t,\theta}]\rvert$, and $\mathbb{E}[N_{t,\theta}]$
$\sigma^2_{min}, \sigma^2_{max}$	The respective maximum and minimum value of each quantity
T_b, P_t	Perishing time for resource $b \in [B]$ and $P_t = \sum_b \mathbb{1}\{T_b = t\}$
β_{avg}	$B / \sum_{\theta \in \Theta} \mathbb{E}[N_\theta]$
X^{opt}, X^{alg}	Optimal fair allocation in hindsight $X_t^{opt} = B/N$ and allocation by algorithm
Δ_{EF}	$\max_{t \in [T], \theta \in \Theta} \lvert w_\theta X_{t,\theta} - w_\theta \frac{B}{N} \rvert$
$\textsc{Envy}$	$\max_{t,t' \in [T]^2, \theta, \theta' \in \Theta^2} w_\theta X^{alg}_{t',\theta'} - w_\theta X^{alg}_{t,\theta}$
$\Delta_{\mathit{efficiency}}$	$B - \sum_{t,\theta} N_{t,\theta} X^{alg}_{t,\theta}$
$\overline{Y}, \underline{Y}$	High probability upper bound and lower bound of a random variable Y
σ	$\sigma : \mathcal{B} \to [B]$ the allocation schedule
$\underline{X}$	Maximum feasible allocation subject to endogenous perishing
$\mathcal{L}^{\mathsf{perish}}$	$\frac{B}{\underline{N}} - \underline{X}$
	Algorithm specification
L_T	Desired bound on $\Delta_{EF}, \textsc{Envy}$
δ	High probability constant
$\textsc{Conf}_t$	Confidence bound on $N_{\geq t}$ and $\mathrm{PUA}_{\geq} t$, indicated by superscript
B_t^{alg}	Budget available to the algorithm at start of round t
$\tau_b(t \mid X, \sigma)$	$\inf\{t' \geq t \mid \underline{N}_{<t}\underline{X} + \underline{N}_{[t,t']}\underline{X} \geq \sigma(b)\}$
$\overline{\Delta}(X)$	$\sum_b \mathbb{P}(T_b < \min\{T, \tau_b(1 \mid X, \sigma)\}) + \textsc{Conf}_1^P$
$\mathcal{A}_t$	Set of resources allocated by algorithm at time t
$\mathrm{PUA}^{alg}_{\geq t}$	$\sum_{\tau=t}^{T-1} \sum_{B \in \mathcal{B}_\tau^{alg}} \mathbb{1}\{T_b = \tau, b \notin \mathcal{A}_\tau\}$, perished and unallocated resources after t
$\overline{P}_t$	Upper bound on $\mathrm{PUA}_{\geq t}$
	Additional notation
$\Phi(\cdot)$	Standard normal CDF

Table 5: Common notation

B Omitted proofs

B.1 Section 3 omitted proofs

B.1.1 Proof of Theorem 3.2

Proof. We first argue that offset-expiry implies feasibility of B/N. Consider the allocation schedule which allocates goods in increasing order of perishing time (breaking ties arbitrarily), and is such that $X_{t,\theta} = B/N$ for all t, θ, as long as there are resources remaining. Noting that $(B/N)N_{<t}$ is precisely the cumulative allocation at the beginning of round t, this implies that we allocate (weakly) more than the number of goods with perishing time before round t (i.e. $P_{<t}$). Since we allocate goods in increasing order of perishing time, this also implies that no unit ever perishes under this sequence of allocations. Thus, the total allocation by the end of the horizon is $\frac{B}{N} \cdot N = B$, implying that B/N is feasible.

We now argue that offset-expiry is necessary for B/N to be feasible. To see this, consider the first period $t \geq 2$ for which $P_{<t}/B > N_{<t}/N$ (i.e., by the end of period $t-1$, there existed some unallocated goods that had perished). Then, the remaining budget at the start of period t for any algorithm, denoted by B_t^{alg}, is:

$$B_t^{alg} \leq B - P_{<t} < B - N_{<t} \cdot \frac{B}{N} = N_{\geq t} \cdot \frac{B}{N},$$

which implies that the remaining budget does not suffice to allocate B/N to all arrivals from t onwards. Hence, B/N is not feasible. $\square$

B.2 Tightness of bounds

Consider the random problem instance which achieves the lower bounds of Theorem 2.3 with probability 1/2, and the lower bounds of Theorem 3.7 with probability 1/2. Putting these two bounds together, we have:

$$\mathbb{E}[\Delta_{EF}] \gtrsim \mathcal{L}^{\mathsf{perish}} + 1/\sqrt{T}.$$

By Theorem 4.2, our algorithm achieves $\mathbb{E}[\Delta_{EF}] \lesssim \max\{L_T, \mathcal{L}^{\mathsf{perish}} + 1/\sqrt{T}\}$. Letting $L_T \lesssim \mathcal{L}^{\mathsf{perish}} + 1/\sqrt{T}$ then, our algorithm achieves this lower bound. We now argue that our algorithm is tight with respect to efficiency in this regime. Suppose $L_T = 0$. By Theorem 2.3 and Theorem 3.7, any online algorithm incurs:

$$\mathbb{E}[\Delta_{\mathit{efficiency}}] \gtrsim T\mathcal{L}^{\mathsf{perish}} + \sqrt{T},$$

which is achieved by our algorithm.

Consider now the regime in which $\Delta_{EF} = L_T$, i.e., $L_T \gtrsim \mathcal{L}^{\mathsf{perish}} + 1/\sqrt{T}$. Again, randomizing between the two lower bounds, we have:

$$\mathbb{E}[\Delta_{\mathit{efficiency}}] \gtrsim T\mathcal{L}^{\mathsf{perish}} + \min\{\sqrt{T}, L_T^{-1}\}. \tag{19}$$

Case 1: $L_T^{-1} + \sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}} \gtrsim \sqrt{T}$. Here, our algorithm achieves $\mathbb{E}[\Delta_{\mathit{efficiency}}] \lesssim \sqrt{T} + T\mathcal{L}^{\mathsf{perish}}$.

If $L_T^{-1} \gtrsim \sqrt{T}$, we achieve the bound in (19). Suppose now that $L_T^{-1} = o(\sqrt{T})$. Then, (19) implies that $\mathbb{E}[\Delta_{\mathit{efficiency}}] \gtrsim T\mathcal{L}^{\mathsf{perish}} + L_T^{-1}$. We argue that, if $L_T^{-1} = o(\sqrt{T})$, then in this case $T\mathcal{L}^{\mathsf{perish}} \gtrsim \sqrt{T}$. $T\mathcal{L}^{\mathsf{perish}}$ then dominates both the lower bound in (19), as well as our upper bound, which gives us tightness.

Case 2: $L_T^{-1} + \sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}} \lesssim \sqrt{T}$. Here, our algorithm achieves $\mathbb{E}[\Delta_{\mathit{efficiency}}] \lesssim L_T^{-1} + \sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}} + T\mathcal{L}^{\mathsf{perish}}$. Since $L_T^{-1} \lesssim \sqrt{T}$, (19) reduces to $\mathbb{E}[\Delta_{\mathit{efficiency}}] \gtrsim T\mathcal{L}^{\mathsf{perish}} + L_T^{-1}$. It is easy to check that $\sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}} \lesssim \max\{L_T^{-1}, T\mathcal{L}^{\mathsf{perish}}\}$, which completes the tightness argument.

B.3 Section 4.1 omitted proofs

B.3.1 Proof of Corollary 4.5

Proof. Consider first the case where $T\mathcal{L}^{\mathsf{perish}} \lesssim L_T^{-1}$. Then:

$$L_T^{-1} + \sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}} \lesssim L_T^{-1} \lesssim \sqrt{T},$$

since $L_T \gtrsim 1/\sqrt{T}$ by assumption. Thus,

$$\Delta_{\mathit{efficiency}} \lesssim \min\left\{\sqrt{T}, L_T^{-1} + \sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}}\right\} + T\mathcal{L}^{\mathsf{perish}} \lesssim L_T^{-1},$$

where again we've used the assumption that $T\mathcal{L}^{\mathsf{perish}} \lesssim L_T^{-1}$.

For the bound on Δ_{EF}, we use the facts that $L_T \gtrsim 1/\sqrt{T}$ and $\mathcal{L}^{\mathsf{perish}} \lesssim 1/\sqrt{T}$ to obtain:

$$\Delta_{EF} \lesssim \max\{L_T, \mathcal{L}^{\mathsf{perish}} + 1/\sqrt{T}\} \lesssim L_T.$$

Suppose now $T\mathcal{L}^{\mathsf{perish}} \gtrsim L_T^{-1}$. In this case:

$$L_T^{-1} + \sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}} \lesssim T\mathcal{L}^{\mathsf{perish}}.$$

Using the fact that $\mathcal{L}^{\mathsf{perish}} \lesssim 1/\sqrt{T}$, we obtain:

$$\Delta_{\mathit{efficiency}} \lesssim \min\left\{\sqrt{T}, L_T^{-1} + \sqrt{T\mathcal{L}^{\mathsf{perish}} L_T^{-1}}\right\} + T\mathcal{L}^{\mathsf{perish}} \lesssim T\mathcal{L}^{\mathsf{perish}}.$$

For the bound on Δ_{EF}, we similarly have $\Delta_{EF} \lesssim L_T$, since $\mathcal{L}^{\mathsf{perish}} \lesssim 1/\sqrt{T} \lesssim L_T$, by assumption. □

B.4 Section 4.2 omitted proofs

For ease of notation, we let $\nu_t = \mathbb{E}[P_{<t}]$ for all $t \in \{2, \ldots, T\}$.

B.4.1 Proof of Proposition 4.6

Proof. Let $t \in [T]$ be such that $\mathbb{E}[P_{<t}] > t - 1$. Then:

$$\mathbb{P}(P_{<t} \leq t - 1 \ \forall\ t \geq 2) \leq \mathbb{P}(P_{<t} \leq t - 1) = \mathbb{P}\left(\sum_{b \in \mathcal{B}} \mathbb{1}\{T_b < t\} \leq t - 1\right), \tag{20}$$

where the second equality is by definition.

Consider first the case where $\mathcal{B}_{<t}^{rand} = \emptyset$. In this case, if b perishes before t with strictly positive probability, it must be that $b \in \mathcal{B}_{<t}^{det}$. Then:

$$\mathbb{P}\left(\sum_{b\in\mathcal{B}} \mathbb{1}\{T_b < t\} \leq t-1\right) = \mathbb{P}\left(\sum_{b\in\mathcal{B}_{<t}^{det}} \mathbb{1}\{T_b < t\} \leq t-1\right) = \mathbb{P}\Big(|\mathcal{B}_{<t}^{det}| \leq t-1\Big), \tag{21}$$

where the second equality follows from the fact that items in $\mathcal{B}_{<t}^{det}$ perish before t with probability 1. By the same reasoning:

$$t-1 < \mathbb{E}[P_{<t}] = \sum_{b\in\mathcal{B}} \mathbb{P}(T_b < t) = \sum_{b\in\mathcal{B}_{<t}^{det}} \mathbb{P}(T_b < t) = |\mathcal{B}_{<t}^{det}| \implies \mathbb{P}(|\mathcal{B}_{<t}^{det}| \leq t-1) = 0.$$

Plugging this back into (20), we obtain $\mathbb{P}(P_{<t} \leq t-1 \; \forall \; t \geq 2) = 0$.

Consider now the case where $\mathcal{B}_{<t}^{rand} \neq \emptyset$. The goal is to show the existence of ϵ such that $\mathbb{P}(P_{<t} \leq t-1 \; \forall t \geq 2) \leq \epsilon$. Define the random variable:

$$Y_b = \mathbb{1}\{T_b < t\} - \mathbb{P}(T_b < t), \quad b \in \mathcal{B}_{<t}^{rand}.$$

By construction, $\mathbb{E}[Y_b] = 0, 0 < \mathbb{E}[Y_b^2] \leq 1$, and $\mathbb{E}[|Y_b|^3] \leq 1$. We have:

$$\mathbb{P}(P_{<t} \leq t-1) = \mathbb{P}\left(\sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{1}\{T_b < t\} \leq t-1-|\mathcal{B}_{<t}^{det}|\right) = \mathbb{P}\left(\sum_{b\in\mathcal{B}_{<t}^{rand}} Y_b \leq t-1-|\mathcal{B}_{<t}^{det}| - \sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{P}(T_b < t)\right).$$

By assumption, $\mathbb{E}[P_{<t}] = |\mathcal{B}_{<t}^{det}| + \sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{P}(T_b < t) > t-1$. Hence,

$$\mathbb{P}(P_{<t} \leq t-1) \leq \mathbb{P}\left(\sum_{b\in\mathcal{B}_{<t}^{rand}} Y_b \leq 0\right) = \mathbb{P}\left(\frac{\sum_{b\in\mathcal{B}_{<t}^{rand}} Y_b}{\sqrt{\sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{E}[Y_b^2]}} \leq 0\right).$$

Let $\Phi(\cdot)$ denote the cdf of the standard normal distribution. By the Berry-Esseen Theorem,

$$\begin{aligned}
\mathbb{P}\left(\frac{\sum_{b\in\mathcal{B}_{<t}^{rand}} Y_b}{\sqrt{\sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{E}[Y_b^2]}} \leq 0\right) &\leq \Phi(0) + \left(\sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{E}[Y_b^2]\right)^{-3/2} \cdot \sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{E}[|Y_b|^3] \\
&= \frac{1}{2} + \left(\sum_{b\in\mathcal{B}_{<t}^{rand}} \mathrm{Var}[\mathbb{1}\{T_b < t\}]\right)^{-3/2} \cdot \sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{E}[|Y_b|^3] \\
&= \frac{1}{2} + \left(\mathrm{Var}\left[\sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{1}\{T_b < t\}\right]\right)^{-3/2} \cdot \sum_{b\in\mathcal{B}_{<t}^{rand}} \mathbb{E}[|Y_b|^3] \\
&\leq \frac{1}{2} + \mathrm{Var}[P_{<t}]^{-3/2} \cdot T \\
&= \frac{1}{2} + \mathrm{Std}[P_{<t}]^{-3} \cdot T.
\end{aligned}$$

Putting this all together, we obtain:

$$\mathbb{P}(P_{<t} \leq t-1 \; \forall t \geq 2) \leq \frac{1}{2} + \mathrm{Std}[P_{<t}]^{-3} \cdot T.$$

As a result, the perishing process cannot be δ-offset expiring for any $\delta < \frac{1}{2} - \mathrm{Std}[P_{<t}]^{-3} \cdot T$.

$\square$

B.4.2 Proof of Proposition 4.7

Proof. Recall, $P_{<t} = |\mathcal{B}^{det}_{<t}| + \sum_{b \in \mathcal{B}^{rand}_{<t}} \mathbb{1}\{T_b < t\}$. By Chebyshev's inequality:

$$\mathbb{P}(P_{<t} > t-1) = \mathbb{P}(P_{<t} - \nu_t \geq t - \nu_t) \leq \left(\frac{\mathrm{Std}[P_{<t}]}{t - \nu_t}\right)^2.$$

Similarly, by Hoeffding's inequality we have:

$$\mathbb{P}(P_{<t} > t-1) = \mathbb{P}\left(\sum_{b \in \mathcal{B}^{rand}_{<t}} \mathbb{1}\{T_b < t\} - \mathbb{P}(T_b < t) \geq t - \nu_t\right) \leq \exp\left(-\frac{2(t-\nu_t)^2}{|\mathcal{B}^{rand}_{<t}|}\right).$$

Then, via a straightforward union bound we have:

$$\mathbb{P}(P_{<t} \leq t-1 \; \forall\, t \geq 2) \geq 1 - \sum_{t=2}^{T} \mathbb{P}(P_{<t} > t-1) \geq 1 - \sum_{t=2}^{T} \min\left\{\left(\frac{\mathrm{Std}[P_{<t}]}{t-\nu_t}\right)^2, \exp\left(-\frac{2(t-\nu_t)^2}{|\mathcal{B}^{rand}_{<t}|}\right)\right\}.$$

□

B.5 Section 4.3 omitted proofs

B.5.1 Proof of Lemma 4.8

Proof. Fix $t' < t$. Recall, for all $\tau \in [T]$, $\rho_{\tau,\theta} \geq |N_{\tau,\theta} - \mathbb{E}[N_{\tau,\theta}]|$, which implies

$$N_{\tau,\theta} \in [\mathbb{E}[N_{\tau,\theta}] - \rho_{\tau,\theta}, \mathbb{E}[N_{\tau,\theta}] + \rho_{\tau,\theta}].$$

Thus, from a simple application of Hoeffding's inequality (Lemma C.1):

$$\mathbb{P}(|N_{(t,t']} - \mathbb{E}\big[N_{(t,t']}\big]| \geq \epsilon) \leq 2\exp\left(-\frac{2\epsilon^2}{\sum_\theta \sum_{\tau \in (t,t']} 4\rho^2_{\tau,\theta}}\right) \tag{22}$$

We now consider our desired bound.

$$\mathbb{P}(|N_{(t,t']} - \mathbb{E}\big[N_{(t,t']}\big]| \leq \epsilon \,\forall\, t,t') \geq 1 - \sum_{t,t'} \mathbb{P}(|N_{(t,t']} - \mathbb{E}\big[N_{(t,t']}\big]| \geq \epsilon)$$

$$\geq 1 - \sum_{t,t'} 2\exp\left(-\frac{2\epsilon^2}{\sum_\theta \sum_{\tau \in (t,t']} 4\rho^2_{\tau,\theta}}\right) \geq 1 - \sum_{t,t'} 2\exp\left(-\frac{\epsilon^2}{2|\Theta|\rho^2_{\max}(t'-t)}\right)$$

where the first inequality follows from a union bound, the second inequality by plugging in Hoeffding's bound (22), and the third inequality by upper bounding $\rho_{\tau,\theta}$ by $\rho_{\max}$, for all $\tau \in (t,t']$. Solving for ϵ such that $2\exp\left(-\frac{\epsilon^2}{2|\Theta|\rho^2_{\max}(t'-t)}\right) = \delta/T^2$, we obtain our result. □

B.5.2 Proof of Lemma 4.10

Proof. The final high-probability bound follows from straightforward algebra, putting Lemmas 4.8 and 4.9 together. Indeed, we have that:

$$\mathbb{P}(\mathcal{E}) = 1 - \mathbb{P}(\mathcal{E}^c) \geq 1 - \mathbb{P}(\mathcal{E}^c_{oe}) - \mathbb{P}((\mathcal{E}_N \cap \mathcal{E}_{\overline{P}})^c) \geq 1 - \delta - \mathbb{P}((\mathcal{E}_N \cap \mathcal{E}_{\overline{P}})^c).$$

We have:

$$\mathbb{P}(\mathcal{E}_N \cap \mathcal{E}_{\overline{P}}) = \mathbb{P}(\mathcal{E}_{\overline{P}} \mid \mathcal{E}_N)\mathbb{P}(E_N) \geq (1-\delta)^2 \geq 1 - 2\delta$$
$$\implies \mathbb{P}((\mathcal{E}_N \cap \mathcal{E}_{\overline{P}})^c) \leq 2\delta.$$

Plugging this in above, we obtain $\mathbb{P}(\mathcal{E}) \geq 1 - 3\delta$. □

B.5.3 Proof of Lemma 4.12

Proof. Consider first the case in which the algorithm always allocates $\underline{X}$ (i.e., $t_0 = T$). Then, the inequality is trivially satisfied and it suffices to prove the lower bound for $t_0 < T$. We have:

$$\begin{aligned}
B_T^{alg} &= B_{t_0}^{alg} - N_{t_0}\underline{X} - \mathrm{PUA}_{t_0}^{alg} - N_{(t_0,T)}\overline{X} - \mathrm{PUA}_{>t_0}^{alg} \\
&< N_{t_0}\overline{X} + \overline{N}_{>t_0}\underline{X} + \overline{P}_{t_0} - N_{t_0}\underline{X} - \mathrm{PUA}_{t_0}^{alg} - N_{(t_0,T)}\overline{X} - \mathrm{PUA}_{>t_0}^{alg} \\
&= N_{t_0}L_T + (\overline{N}_{>t_0} - N_{>t_0})\underline{X} + N_T\underline{X} - N_{(t_0,T)}L_T + \overline{P}_{t_0} - \mathrm{PUA}_{\geq t_0}^{alg} \qquad (23)
\end{aligned}$$

where the first inequality follows from the fact that $B_{t_0}^{alg} < N_{t_0}\overline{X} + \overline{N}_{>t_0}\underline{X} + \overline{P}_{t_0}$ since $X_{t_0}^{alg} = \underline{X}$, and the second inequality uses $\overline{X} = \underline{X} + L_T$ and re-arranges terms. Since $\overline{X}$ was allocated at T, $B_T^{alg} - N_T\overline{X} \geq 0$, which then implies that $B_T^{alg} - N_T\underline{X} \geq N_T L_T$. Plugging this fact into (23) and re-arranging, we obtain:

$$N_{t_0}L_T + (\overline{N}_{>t_0} - N_{>t_0})\underline{X} - N_{>t_0}L_T + \overline{P}_{t_0} - \mathrm{PUA}_{\geq t_0}^{alg} > 0 \qquad (24)$$

We now upper bound the left-hand side of (24). Using the facts that $\mathrm{PUA}_{\geq t_0}^{alg} \geq 0$, $\underline{X} \leq \beta_{avg}$ by construction, and $\overline{P}_{t_0} \leq \overline{P}_1 = \overline{\Delta}(\underline{X})$, we have, for $C = \sqrt{2|\Theta|\rho_{max}^2 \log(2T^2/\delta)}$:

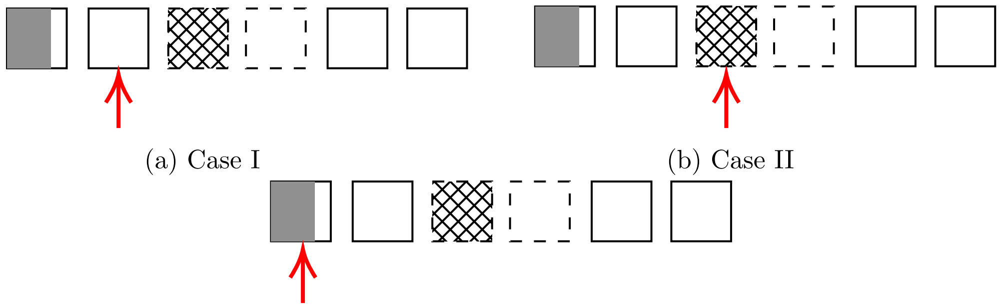
$$\begin{aligned}
0 < N_{t_0}L_T + (\overline{N}_{>t_0} - N_{>t_0})\underline{X} - N_{>t_0}L_T + \overline{P}_{t_0} - \mathrm{PUA}_{\geq t_0}^{alg} &\leq \rho_{\max}L_T + 2C\beta_{avg}\sqrt{T - t_0} \\
&\quad -L_T(T - t_0) + \overline{\Delta}(\underline{X}). \qquad (25)
\end{aligned}$$

Consider now the quadratic function $f(x) = -L_T x^2 + 2C\beta_{avg}x + \rho_{\max}L_T + \overline{\Delta}(\underline{X})$, which has a positive root at:

$$\begin{aligned}
x^+ &= \frac{2C\beta_{avg} + \sqrt{4C^2\beta_{avg}^2 + 4L_T\big(\rho_{\max}L_T + \overline{\Delta}(\underline{X})\big)}}{2L_T} \\
&= \frac{C\beta_{avg}}{L_T} + \sqrt{\frac{C^2\beta_{avg}^2}{L_T^2} + \frac{\rho_{\max}L_T + \overline{\Delta}(\underline{X})}{L_T}} \\
&\leq 2\frac{C\beta_{avg}}{L_T} + \sqrt{\frac{\overline{\Delta}(\underline{X})}{L_T}} + \sqrt{\rho_{\max}} \\
&< c\left(\frac{1}{L_T} + \sqrt{\frac{\overline{\Delta}(\underline{X})}{L_T}}\right),
\end{aligned}$$

for some $c \in \widetilde{\Theta}(1)$. Thus, for all $x \geq c\left(\frac{1}{L_T} + \sqrt{\frac{\overline{\Delta}(\underline{X})}{L_T}}\right)$, $f(x) \leq 0$. Letting $x = \sqrt{T - t_0}$, we obtain that the right-hand side of (25) is non-positive for all t_0 such that $T - t_0 \geq c^2\left(\frac{1}{L_T} + \sqrt{\frac{\overline{\Delta}(\underline{X})}{L_T}}\right)^2 \iff t_0 \leq T - c^2\left(\frac{1}{L_T} + \sqrt{\frac{\overline{\Delta}(\underline{X})}{L_T}}\right)^2$, which would lead to a contradiction.

Concluding, we have $t_0 > T - c^2\left(\frac{1}{L_T} + \sqrt{\frac{\overline{\Delta}(\underline{X})}{L_T}}\right)^2 \geq T - \tilde{c}^2\left(\frac{1}{L_T} + \sqrt{\frac{T\mathcal{L}^{\mathsf{perish}}}{L_T}}\right)^2$, where the final inequality follows from the fact that $\overline{\Delta}(\underline{X}) \leq B - \overline{N}\underline{X} = \overline{N}\mathcal{L}^{\mathsf{perish}} \lesssim T\mathcal{L}^{\mathsf{perish}}$. □



(a) Case I

(b) Case II

(c) Case III

Figure 6: Illustration of the three cases in the induction step of the proof of the fact that $\mathrm{PUA}_{t+1}^{alg} \leq \mathrm{PUA}_{t+1}(\underline{X})$ (Lemma 4.13). Here, we assume $\mathcal{B}_{t+1}^{alg} \subseteq \mathcal{B}_{t+1}(\underline{X})$. The squares across all three plots show the resources in $\mathcal{B}_{t+1}(\underline{X})$, ordered left to right according to σ; the dashed squares correspond to $\mathcal{B}_{t+1}^{alg}$. The gray-shaded region corresponds to resource fractionally allocated by the $\underline{X}$ process at the beginning of $t+1$, and the cross-hatched region to the set of resources allocated by our algorithm. Finally, the red arrow corresponds to the resource b considered in each case.

B.5.4 Proof of Lemma 4.13

Proof of Lemma 4.13. We show the two properties by induction on t.
Base Case $t = 1$. By definition, $\mathcal{B}_1^{alg} = \mathcal{B} = \mathcal{B}_t(\underline{X})$. We now argue that $\mathrm{PUA}_1^{alg} \leq \mathrm{PUA}_1(\underline{X})$. Suppose there exists a resource b which perished at the end of $t = 1$. Then, either:

1. b was neither allocated by our algorithm, nor under the $\underline{X}$ allocation. Hence, it perished unallocated under both allocations.

2. b was allocated by our algorithm but not by the $\underline{X}$ allocation. Hence, it perished unallocated under $\underline{X}$ but not our algorithm.

3. b was allocated under the $\underline{X}$ allocation but not by our algorithm. This could never hold, since both algorithms begin with the same set of resources and our algorithm allocated (weakly) more than $\underline{X}$, and under the same ordering σ.

Step case $t \to t+1$. We first show that $\mathcal{B}_{t+1}^{alg} \subseteq \mathcal{B}_{t+1}(\underline{X})$. Let $b \in \mathcal{B}_{t+1}^{alg}$. Then b did not perish, and moreover $b \in \mathcal{B}_t^{alg}$. Then, by the inductive hypothesis, $b \in \mathcal{B}_t(\underline{X})$. Consider the following cases:

1. b was not allocated under the $\underline{X}$ process. Then, $b \in \mathcal{B}_{t+1}(\underline{X})$, since it did not perish.

2. b was allocated under the $\underline{X}$ process. In this case, since the algorithm allocated (weakly) more than $\underline{X}$ according to the same ordering, this resource *must* have been available to both the algorithm and the $\underline{X}$ process. This then contradicts that b was not allocated by the algorithm.

We now argue that $\mathrm{PUA}_{t+1}^{alg} \leq \mathrm{PUA}_{t+1}(\underline{X})$. Suppose there exists a resource b that perished at time $t+1$. We consider the following cases (see Figure 6 for an illustration):

1. b was neither allocated by our algorithm, nor under the $\underline{X}$ allocation. Hence, it perished unallocated under both allocations.

2. b was allocated by our algorithm but not by the $\underline{X}$ allocation. Hence, it perished unallocated under $\underline{X}$ but not our algorithm.

3. b was allocated under the $\underline{X}$ allocation but not by our algorithm. Then, b *must* have either perished or been allocated before $t+1$, since the set of remaining resources under our algorithm is (weakly) nested in the set of remaining resources under the $\underline{X}$ for all $t' \leq t$, by the inductive hypothesis. Thus, b could not have perished at the end of $t+1$ under our algorithm's sample path.

□

B.6 Section 5 omitted proofs

B.6.1 Proof of Proposition 5.1

Proof. For all $b \in \mathcal{B}$, we have:

$$\mathbb{P}(T_b \leq t-1) = 1-(1-p)^{t-1} \implies \begin{cases} \nu_t & = T\big(1-(1-p)^{t-1}\big) \\ \mathrm{Var}[P_{<t}] & = T\big(1-(1-p)^{t-1}\big)(1-p)^{t-1}. \end{cases} \tag{26}$$

By Proposition 4.7, δ-offset expiry holds for all $\delta \geq \sum_{t=2}^{T}\left(\frac{\mathrm{Std}[P_{<t}]}{t-\nu_t}\right)^2$. By (26):

$$\left(\frac{\mathrm{Std}[P_{<t}]}{t-\nu_t}\right)^2 = \frac{T\big(1-(1-p)^{t-1}\big)(1-p)^{t-1}}{\big(t-T(1-(1-p)^{t-1})\big)^2}$$

Using the fact that $(1-p)^{t-1} \geq 1-(t-1)p$, we have:

$$t - T(1-(1-p)^{t-1}) \geq t - T(t-1)p > 0,$$

where the final inequality follows from the assumption that $p \leq 1/T$. Taking derivatives, it is easy to show that the function $f(x) = \frac{(1-x)x}{(t-T(1-x))^2}$ is decreasing for $t - T(1-x) > 0$. Hence, leveraging the same lower bound on $(1-p)^{t-1}$ we have:

$$\begin{aligned}
\left(\frac{\mathrm{Std}[P_{<t}]}{t-\nu_t}\right)^2 &\leq \frac{T(t-1)p(1-(t-1)p)}{(t-T(t-1)p)^2} \leq \frac{Ttp(1-(t-1)p)}{t^2(1-Tp)^2} \leq \frac{Tp(1-(t-1)p)}{(t-1)(1-Tp)^2} \\
\implies \sum_{t=2}^{T}\left(\frac{\mathrm{Std}[P_{<t}]}{t-\nu_t}\right)^2 &\leq \frac{Tp}{(1-Tp)^2}\sum_{t=2}^{T}\left(\frac{1}{t-1} - p\right) \\
&= \frac{Tp}{(1-Tp)^2}\left(\sum_{t=1}^{T-1}\frac{1}{t} - p(T-1)\right) \\
&\leq \frac{Tp}{(1-Tp)^2}\cdot 2\log T.
\end{aligned}$$

Hence, the perishing process if δ-offset expiring for all $\delta \geq 2\log T \cdot \frac{Tp}{(1-Tp)^2}$.

We conclude by showing the lower bound on $\underline{X}$. By definition,

$$\begin{aligned}
\overline{\Delta}(X) &= \mu(X) + \frac{1}{2}\left(\log(3\log(T)/\delta) + \sqrt{\log^2(3\log(T)/\delta) + 8\mu(X)\log(3\log(T)/\delta)}\right) \\
&\leq \mu(X) + \frac{1}{2}\left(2\log(3\log(T)/\delta) + \frac{1}{2\log(3\log(T)/\delta)}\cdot 8\mu(X)\log(3\log(T)/\delta)\right) \\
&= 3\mu(X) + \log(3\log(T)/\delta),
\end{aligned}$$

where the inequality follows from concavity. Moreover:

$$\mu(X) = \sum_b \mathbb{P}(T_b < \min\{T, \tau_b(1 \mid X, \sigma)\}) \leq \sum_b \mathbb{P}(T_b < T) = T(1-(1-p)^{T-1})$$
$$\implies \overline{\Delta}(X) \leq 3T(1-(1-p)^{T-1}) + \log(3\log(T)/\delta)$$
$$\leq 3T(T-1)p + \log(3\log(T)/\delta).$$

Since any feasible stationary allocation X must satisfy $X \leq \frac{T-\overline{\Delta}(X)}{T}$, it suffices to have:

$$X \leq \frac{T - 3T(T-1)p - \log(3\log(T)/\delta)}{T} = 1 - 3(T-1)p - \frac{\log(3\log(T)/\delta)}{T}. \tag{27}$$

Noting that the right-hand side of (27) is non-negative for $\delta \geq 3\log T \cdot \exp(-(T-3T^2p))$, and that $2\log T \cdot \frac{Tp}{(1-Tp)^2} \geq 3\log T \cdot \exp(-(T-3T^2p))$ for $p = o(1)$, we obtain the result. □

B.7 Proof of Proposition 5.2

Proof. For ease of notation, we let $\mu_b = \mathbb{E}[T_b]$. For a stationary allocation $X = 1 - T^{-\alpha}$, let $\mu(X) = \sum_b \mathbb{P}(T_b < \min\{T, \lceil \frac{\sigma(b)}{X} \rceil\})$, and $\mu_b = \mathbb{E}[T_b]$. By Chebyshev's inequality, we have:

$$\mu(X) \leq \sum_b \mathbb{P}\left(T_b - \mu_b \leq \min\left\{T, \lceil \frac{\sigma(b)}{X} \rceil\right\} - \mu_b\right) \leq \sum_b \mathbb{P}\left(|T_b - \mu_b| \geq \mu_b - \min\left\{T, \lceil \frac{\sigma(b)}{X} \rceil\right\}\right)$$
$$\leq \sum_b \frac{\mathrm{Var}[T_b]}{\left(\mu_b - \min\{T, \lceil \frac{\sigma(b)}{X} \rceil\}\right)^2}, \tag{28}$$

where we used the assumption that $\mu_b > \min\{T, \lceil \frac{\sigma(b)}{X} \rceil\}$ for all $b \in \mathcal{B}$. However, by assumption we have

$$\mu(X) \leq \sum_b \frac{\mathrm{Var}[T_b]}{\left(\mu_b - \min\{T, \lceil \frac{\sigma(b)}{X} \rceil\}\right)^2} \leq \frac{1}{2} T^{1-\alpha}$$

Moreover, by definition:

$$\overline{\Delta}(X) = \mu(X) + \frac{1}{2}\left(\log(3\log(T)/\delta) + \sqrt{\log^2(3\log(T)/\delta) + 8\mu(X)\log(3\log(T)/\delta)}\right)$$
$$\leq \mu(X) + \log(3\log(T)/\delta) + \sqrt{2\mu(X)\log(3\log(T)/\delta)}.$$

Since any feasible X must satisfy $X \leq 1 - \overline{\Delta}(X)/T$, we have that $\overline{\Delta}(X) \leq T^{1-\alpha}$ for $X = 1 - T^{-\alpha}$. Thus, it suffices for $\mu(X)$ to satisfy

$$\mu(X) + \sqrt{2\mu(X)\log(3\log(T)/\delta)} + \log(3\log(T)/\delta) \leq T^{1-\alpha}$$

We have that $\mu(X) \leq \frac{1}{2} T^{1-\alpha}$ satisfies this inequality for all $\delta \geq 3\log(T) e^{-\frac{1}{8}T^{1-\alpha}}$. □

C Useful lemmas

We use the following standard theorems throughout the proof. See, e.g. Vershynin (2018) for proofs and further discussion.

Lemma C.1 (Hoeffding's Inequality (Vershynin, 2018)). *Let $X_1, \ldots, X_n$ be independent random variables such that $a_i \leq X_i \leq b_i$ almost surely, with $S_n = \sum_i X_i$. Then, for all $t > 0$:*

$$\mathbb{P}(|S_n - \mathbb{E}[S_n]| \geq t) \leq 2\exp\left(-\frac{2t^2}{\sum_i (b_i - a_i)^2}\right).$$

Lemma C.2 (Chernoff Bound for Sum of Bernoulli Random Variables (Mitzenmacher and Upfal, 2017)). *Consider a sequence of Bernoulli random variables $(X_i)_{i\in[N]}$, independently distributed with probability of success $p_i \in (0,1)$. Let $X = \sum_i X_i$, and let $\mu = \mathbb{E}[X] = \sum_i p_i$. Then, for all $\epsilon > 0$:*

$$\mathbb{P}(X \geq (1+\epsilon)\mu) \leq \exp\left(-\frac{\epsilon^2}{2+\epsilon}\mu\right).$$

Corollary C.3. *Consider a sequence of Bernoulli random variables $(X_i)_{i\in[N]}$, independently distributed with probability of success $p_i \in (0,1)$. Let $X = \sum_i X_i$, and let $\mu = \mathbb{E}[X] = \sum_i p_i$. Then, for any $\delta > 0$, with probability at least $1-\delta$ we have:*

$$X \leq \mu + \frac{1}{2}\left(\log(1/\delta) + \sqrt{\log^2(1/\delta) + 8\mu\log(1/\delta)}\right).$$

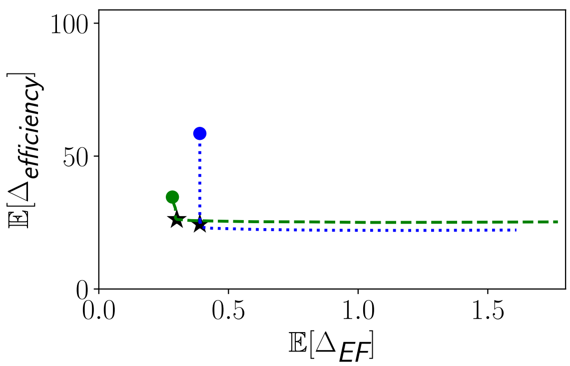
Proof. Setting the right hand side equal to δ in Lemma C.2 and solving for ϵ, we have:

$$\begin{aligned} \frac{\epsilon^2}{2+\epsilon}\mu = \log(1/\delta) &\iff \epsilon^2\mu - \epsilon\log(1/\delta) - 2\log(1/\delta) = 0 \\ \iff \epsilon = \frac{\log(1/\delta) + \sqrt{\log^2(1/\delta) + 8\mu\log(1/\delta)}}{2\mu}&. \end{aligned}$$

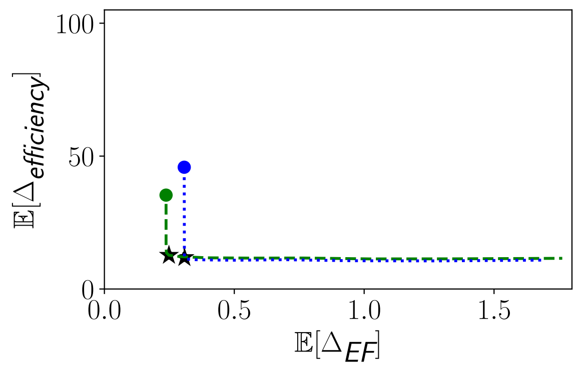
Plugging this value of ϵ into $(1+\epsilon)\mu$ we have the result. □

D Simulation details

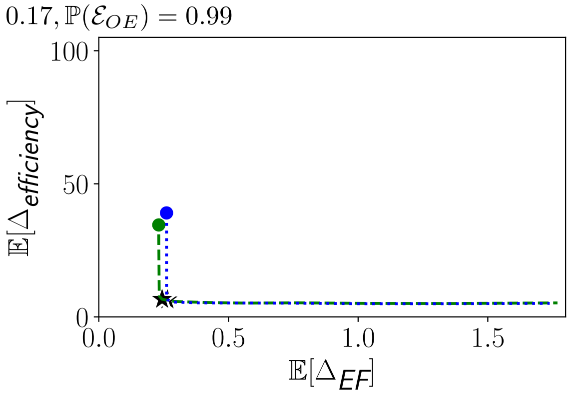
Computing Infrastructure: The experiments were conducted on a personal computer with an Apple M2, 8-core processor and 16.0GB of RAM.



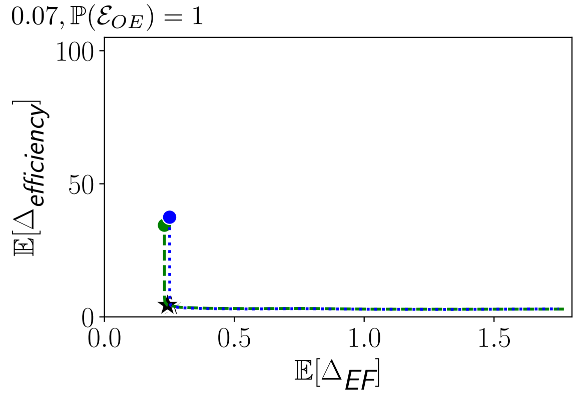
(a) $T_b \sim Geometric(T^{-1.3}) : \underline{X} = 0.93, \mathcal{L}^{\text{perish}} = 0.17, \mathbb{P}(\mathcal{E}_{OE}) = 0.99$



(b) $T_b \sim Geometric(T^{-1.5}) : \underline{X} = 1.03, \mathcal{L}^{\text{perish}} = 0.07, \mathbb{P}(\mathcal{E}_{OE}) = 1$

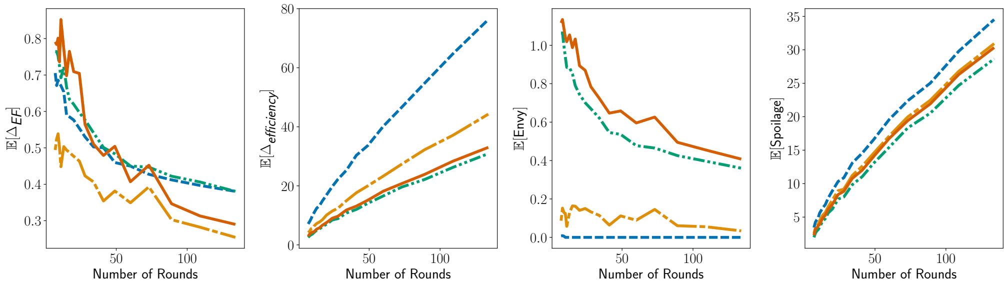


(c) $T_b \sim Geometric(T^{-1.7}) : \underline{X} = 1.06, \mathcal{L}^{\text{perish}} = 0.03, \mathbb{P}(\mathcal{E}_{OE}) = 1$

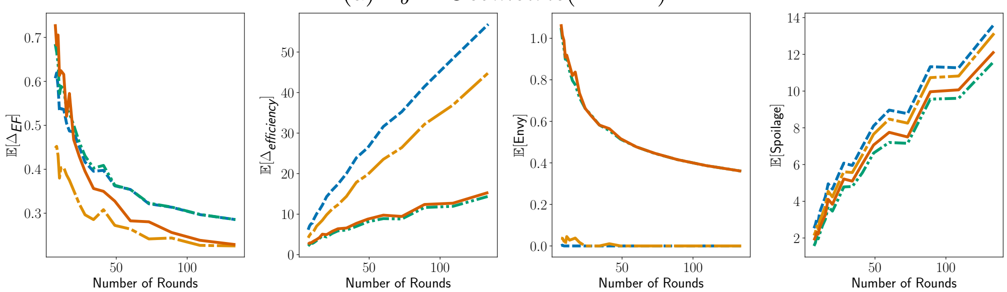


(d) $T_b \sim Geometric(T^{-1.9}) : \underline{X} = 1.07, \mathcal{L}^{\text{perish}} = 0.02, \mathbb{P}(\mathcal{E}_{OE}) = 1$

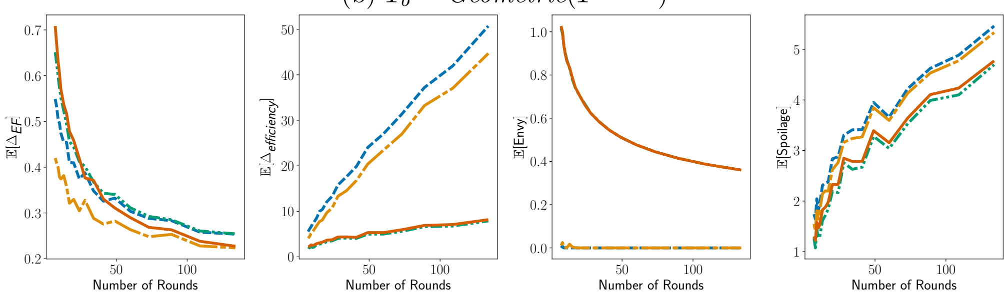
Figure 7: Empirical trade-off between $\Delta_{\mathit{efficiency}}$ and Δ_{EF} for the different algorithms under various values of L_T.



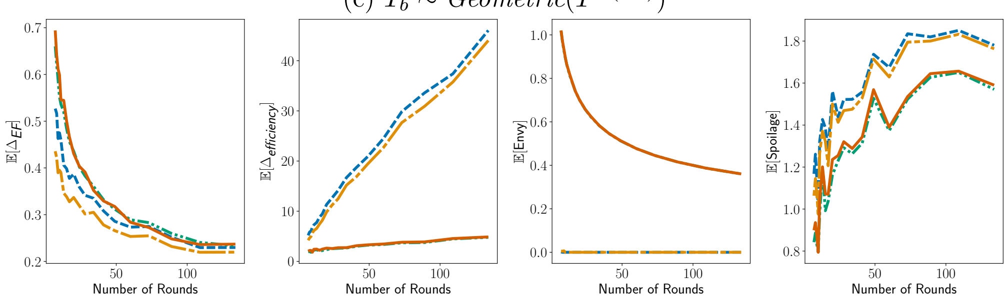
(a) $T_b \sim Geometric(T^{-(1.3)})$



(b) $T_b \sim Geometric(T^{-(1.5)})$



(c) $T_b \sim Geometric(T^{-(1.7)})$



(d) $T_b \sim Geometric(T^{-(1.9)})$

Figure 8: Numerical results for $T_b \sim Geometric(p)$ as described in Section 5.1.