Electronic Companion for "Online Allocation of Throughput-Constrained Resources Using Proxy Assignments"

Appendix A: Additional Discussion of Modeling Assumptions

A.1. Mapping Target Backlogs to Cumulative Deviation Costs

To see why cumulative — and not epoch-specific — deviation costs are used to incentivize target adherence, consider the case-breaking problem described in Section 1.1.3, and suppose there is a single resource (also referred to as a *worker*) processing arriving cases over $T = 400$ periods, with $K = 2$ epochs. (Each epoch therefore contains $T/K = 200$ periods.) Suppose moreover that the worker's maximum processing rate is 100 cases per epoch, and the decision-maker seeks to maintain a stable target backlog of 50 cases at the end of each epoch. Formally, letting A_k denote the number of cases assigned to the worker in epoch k, and B_k the backlog at the end of the epoch, the backlog of unprocessed work evolves according to

$$B_k = (B_{k-1} + A_k - 100)^+, \tag{EC.1}$$

with $B_0 = 50$ (i.e., the system is initialized with 50 cases). Equation (EC.1) implies that, in order to maintain a backlog of $B_k = 50$ for all k, the decision-maker should assign 100 cases to the worker in each epoch. In line with this reasoning, we map this to our model by letting $\rho_k^{\text{epoch}} = \frac{100}{200} = \frac{1}{2}$ be the epoch-specific target assignment rate for each epoch k. In a similar vein, we let $\rho_1^{\text{cumulative}} = \frac{100}{200} = \frac{1}{2}$ be the cumulative target assignment rate at the end of epoch 1, and $\rho_2^{\text{cumulative}} = \frac{200}{400} = \frac{1}{2}$ be the cumulative target assignment rate at the end of epoch 2.

We first illustrate how cumulative deviation costs appropriately penalize *over-assignment*. Consider a sample path such that the decision-maker assigns $A_1 = 110$ cases to the worker in the first epoch (which would occur if, for instance, the assignment costs for the 10 excess assignments were very low). Then, the backlog at the end of the first epoch B_1 would be given by $B_1 = 50 + 110 - 100 = 60$, exceeding the target of 50 by 10 cases. In this case, both the epoch-specific and cumulative assignment rates are given by $\frac{110}{200} = 0.55 > 0.5$, and both types of deviation cost functions would penalize these 10 excess assignments.

Now, suppose the decision-maker assigned exactly $A_2 = 100$ cases in the second epoch. An epoch-specific deviation cost would register no deviation in the second epoch, since the corresponding epoch-specific assignment rate would be $\frac{100}{200} = \frac{1}{2} = \rho_2^{\text{epoch}}$. However, under this assignment, the backlog at the end of the second epoch would be $B_2 = 60 + 100 - 100 = 60$, which would again exceed the target backlog of 50 by 10 cases. Herein lies the main issue with epoch-specific deviation costs: they ignore the reality that excess assignments in the first epoch spill over to the second epoch; they do not simply disappear. In contrast, under a cumulative deviation cost function, in order to exactly hit the cumulative target assignment rate, the decision-maker would need to assign $A_2 = 90$ cases, so that $\frac{A_1 + A_2}{400} = \frac{1}{2}$. Assigning anything more

than this would be penalized, thereby accurately capturing the idea that spillover from the first epoch does not disappear in the second epoch.

To see how cumulative deviation costs appropriately penalize *under-assignment*, suppose now that the decision-maker assigned $A_1 = 90$ cases to the worker in the first epoch. In this case, the worker processes 10 of the cases with which the system was initialized, and the backlog drops to $B_1 = 50 + 90 - 100 = 40$. The backlog is now 10 cases below its target value of 50, increasing the risk that the worker will idle if future arrivals fall short. As above, both epoch-specific and cumulative deviation costs would penalize this risk identically. Now, if the decision-maker assigned exactly $A_2 = 100$ cases to the worker in the second epoch (therefore incurring zero deviation costs under the epoch-specific model), the end-of-epoch backlog would be $B_2 = 40 + 100 - 100 = 40$, and the deficit of 10 cases relative to the target backlog would remain. To rebuild the backlog to its target level of 50 cases, the decision-maker would have to *over-assign* (in particular, assign 110 cases) in the second epoch. This is precisely the behavior induced by cumulative deviation costs, under which the assignment that exactly achieves the cumulative target assignment rate is $\frac{A_1+A_2}{400} = \frac{1}{2} \iff A_2 = 110$.

A.2. Linearly Scaled Deviation Costs

Recall, the deviation cost function $g_{ki}(a)$ can be interpreted as the per-unit cost associated with operating resource i at cumulative assignment *rate* a, while the linear scaling by kT/K converts this normalized loss into a total milestone penalty over the cumulative flow audited at epoch k, reflecting the reality that, in many applications, the operational costs incurred from target deviation are *volume*, rather than *rate*, dependent.

The importance of linearly scaling the deviation cost function to reflect this operational reality is apparent in the above example. For this example, consider a sample path in which the decision-maker assigns 110 cases in each epoch. In the first epoch, the cumulative deviation from $\rho_1 = \frac{1}{2}$ is $\frac{110}{200} - \frac{1}{2} = 0.05$. In the second epoch, the cumulative deviation from $\rho_2 = \frac{1}{2}$ is $\frac{220}{400} - \frac{1}{2} = 0.05$. Observe, however, that while the rate deviation is the same for both epochs, the actual backlog at the end of each epoch differs. We have $B_1 = 50 + 110 - 100 = 60$ at the end of epoch 1 (an excess of 10 over the target backlog of 50), compared to $B_2 = 60 + 110 - 100 = 70$ at the end of epoch 2 (an excess of 20 over the target backlog of 50). In other words, the rate deviation is identical at the two checkpoints, but the cumulative physical imbalance is twice as large at the second checkpoint. Scaling the deviation cost by the volume of arrivals to the end of the second epoch appropriately reflects this reality.

Appendix B: Dual Mirror Descent for Single-Epoch Target-Following

We present the dual mirror descent algorithm developed in Balseiro et al. (2025) for the single-epoch problem below. For ease of presentation, we assume the reference function used for the mirror descent step is the ℓ_2-norm, yielding online gradient descent in the dual space.

ALGORITHM 3: Dual Gradient Descent for Single-Epoch Target-Following

Input: Stepsize η, initial shadow prices $\mu^1 \in \mathbb{R}^m$

Output: Sequence of assignment decisions

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

Observe arrival type $j = j^t$, and make primal assignment decision:

$$x^t \in \arg\min_{x \in \mathcal{X}^t} \quad \sum_{i \in [m]} x_i (c_{ji} - \mu_i^t) \tag{EC.2}$$

Compute idealized average consumption:

$$a^t \in \arg\min_{a \in [0,1]^m} \quad \sum_{i \in [m]} g_{1i}(a_i) + \sum_{i \in [m]} \mu_i^t a_i \tag{EC.3}$$

Update shadow prices via OGD step:

$$\mu^{t+1} = \mu^t + \eta(a^t - x^t). \tag{EC.4}$$

end

Appendix C: Extension to Nonstationary Arrival Rates

Motivated by the reality that arrival rates may vary throughout the decision-making horizon in a number of applications, in this section we relax the assumption that the number of arrivals is equal across all epochs. Specifically, we assume that there are $l_k T$ arrivals in each epoch $k \in [K]$, for some known vector $l \in (0,1]^K$ such that $\sum_{k \in [K]} l_k = 1$. For simplicity, we assume $l_k T$ is integral for all k. Observe that this setting subsumes the basic setting presented in Section 2, in which $l_k = 1/K$ for all $k \in [K]$. This generalization is also able to model "peak" and "non-peak" periods throughout the horizon (e.g., $K = 3$, with $l_1 = 1/4$, $l_2 = 1/2$, $l_3 = 1/4$).

We consider two possible models for the cumulative targets: (i) *per-arrival* targets, which pace the average assignment over all arrivals by the end of each epoch, and (ii) *per-period* targets, which pace the average assignment per unit time. In the example above, for instance, per-arrival targets would average the number of assignments over the $T/4$ arrivals in the first epoch; on the other hand, per-period targets would average the number of assignments over $T/3$ periods, assuming for simplicity that all epochs have equal "time-length."

In the following two sections, we demonstrate a reduction of each of these settings to our basic setup. For ease of notation, we let $l_{\leq k} = \sum_{k' \leq k} l_{k'}$. Moreover, for any policy π, we let $Z_{ji}^{\pi}(t)$ denote the number of type j arrivals assigned to resource i after the t-th arrival; as before, we let $Z_i^{\pi}(t) = \sum_{j \in [n]} Z_{ji}^{\pi}(t)$.

C.1. Per-Arrival Targets

At a high level, we reduce any nonstationary instance to a problem over a stationary instance by partitioning the larger epochs in the original instance in such a way that the final constructed instance is composed

of epochs of equal length. Defining the deviation cost functions associated with these new "sub-epochs" to be identically zero, we observe that these two instances are in fact equivalent. Therefore, applying our algorithm to the constructed stationary instance immediately provides a sublinear regret guarantee for the original instance. We formalize these ideas below.

In this case, the total cost incurred by any policy π over a sample path of arrivals ω is given by:

$$V^{\pi}[\omega] = \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z^{\pi}_{ji}(T) + \sum_{k\in[K]}\sum_{i\in[m]} l_{\leq k} T g_{ki}\left(\frac{Z^{\pi}_{i}(l_{\leq k}T)}{l_{\leq k}T}\right). \tag{EC.5}$$

Let $\mathcal{I}$ denote the instance defined by the K epochs with unequal arrival rates and deviation cost functions $(g_{ki}, k\in[K], i\in[m])$. We construct an equivalent stationary instance, denoted by $\mathcal{I}_{new}$, as follows. Let $l_{new}T$ be the greatest common divisor of $(l_1T, l_2T, \ldots, l_KT)$. For simplicity we assume $l_{new}T = \Theta(T)$, and that $1/l_{new}$ is integral. Then, define $\mathcal{I}_{new}$ to be an instance with $K_{new} = 1/l_{new}$ epochs, and $l_{new}T$ arrivals per epoch.

EXAMPLE EC.1. Consider any instance $\mathcal{I}$ with $K=3$ and $l_1 = 1/4$, $l_2 = 1/2$, $l_3 = 1/4$. Then, $K_{new} = 4$, with $l_{new} = 1/4$.

Let $(\widetilde{g}_{ki}, k\in[K_{new}], i\in[m])$ denote the set of deviation cost functions for this new instance. To define this set of functions, we let $f:[K]\to[K_{new}]$ be the mapping from the set of epochs in the original instance to the set of epochs in the new instance, such that, for all $k\in[K], k_{new}\in[K_{new}]$, $f(k) = k_{new}$ if and only if $l_{\leq k}T = k_{new}l_{new}T$. Equivalently, we have $f(k) = \frac{l_{\leq k}}{l_{new}}$ for all $k\in[K]$. In words, k_{new} is such that the number of arrivals by the end of the k-th epoch in the original instance is equal to the number of arrivals by the end of the k_{new}-th epoch in the new instance. Let $\mathcal{K}_{new} = \{f(k), k\in[K]\}$. Then, for all $i\in[m]$, we define:

$$\widetilde{g}_{ki}(a) = \begin{cases} g_{f^{-1}(k),i}(a) & \forall\, k\in\mathcal{K}_{new} \\ 0 & \text{otherwise.}\end{cases}$$

In words, $\mathcal{I}_{new}$ is such that the deviation cost is zero at each new epoch endpoint that does not coincide with an original epoch endpoint; for all other epochs, the target is the same.

EXAMPLE EC.2 (CONTINUED FROM EXAMPLE EC.1). In this case, $f(1)=1$, $f(2)=3$ and $f(3)=4$, with $\mathcal{K}_{new} = \{1,3,4\}$. Moreover, $\widetilde{g}_{1,i}(a) = g_{1,i}(a)$, $\widetilde{g}_{2,i}(a) = 0$, $\widetilde{g}_{3,i}(a) = g_{2,i}(a)$, and $\widetilde{g}_{4,i}(a) = g_{3,i}(a)$.

The following proposition establishes that the problem under $\mathcal{I}$ reduces to that under $\mathcal{I}_{new}$. We defer the proof of Proposition EC.1 to Appendix C.1.1.

PROPOSITION EC.1. *Fix a sample path of arrivals ω and a sequence of assignment decisions, denoted by Z^{π}. Let $V^{\pi}[\omega]$ and $V^{\pi}_{new}[\omega]$ respectively denote the total cost incurred by Z^{π} in $\mathcal{I}$ and $\mathcal{I}_{new}$. Then, $V^{\pi}[\omega] = V^{\pi}_{new}[\omega]$.*

Consider now the algorithm which, given nonstationary instance $\mathcal{I}$, constructs stationary instance $\mathcal{I}_{new}$, and runs Algorithm 2 on $\mathcal{I}_{new}$, implementing the same assignment decisions on the true instance, for all $t \in [T]$. We refer to this algorithm as NS-PROXY-DUAL-GD. We have the following result, which is an immediate corollary of Proposition EC.1.

COROLLARY EC.1. *For $\eta = \Theta\left(\sqrt{K_{new}/T}\right)$, the gap between the expected cost incurred by* NS-PROXY-DUAL-GD *and the offline benchmark is upper bounded by $\mathbb{E}[\text{REG}] = O(K_{new}^{5/2}\sqrt{T})$.*

C.1.1. Proof of Proposition EC.1

Proof. The total cost incurred by Z^π in the constructed instance is given by:

$$\begin{aligned} V_{new}^\pi[\omega] &= \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z_{ji}^\pi(T) + \sum_{k\in[K_{new}]}\sum_{i\in[m]} kl_{new}T\widetilde{g}_{ki}\left(\frac{Z_i^\pi(kl_{new}T)}{kl_{new}T}\right) \\ &= \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z_{ji}^\pi(T) + \underbrace{\sum_{k\in\mathcal{K}_{new}}\sum_{i\in[m]} kl_{new}T\widetilde{g}_{ki}\left(\frac{Z_i^\pi(kl_{new}T)}{kl_{new}T}\right)}_{(I)}, \end{aligned}$$

where the second equality follows from the fact that $\widetilde{g}_{ki}(\cdot) = 0$ for all $k \notin \mathcal{K}_{new}$. Recall, for all $k \in \mathcal{K}_{new}$, by construction we have:

$$\begin{cases} l_{\le f^{-1}(k)}T = kl_{new}T \\ \widetilde{g}_{ki}(\cdot) = g_{f^{-1}(k),i}(\cdot). \end{cases}$$

Hence, we can re-write (I) in the space of epochs of the original instance, i.e.,

$$(I) = \sum_{k\in[K]}\sum_{i\in[m]} l_{\le k}T g_{ki}\left(\frac{Z_i^\pi(l_{\le k}T)}{l_{\le k}T}\right).$$

Plugging this back into $V_{new}^\pi[\omega]$ and comparing this to (EC.5), we obtain the result. □

C.2. Per-Period Targets

Consider now the setting in which targets (and consequently deviation costs) are specified per-period; for simplicity we assume that each epoch has "time-length" T/K in this setting. In this case, we trivially reduce to the nonstationary setting with *per-arrival* targets by appropriately rescaling the deviation cost functions. In particular, with per-period targets, the total cost incurred by any policy π over a sample path of arrivals ω is given by:

$$V^\pi[\omega] = \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z_{ji}^\pi(T) + \sum_{k\in[K]}\sum_{i\in[m]} \frac{kT}{K} g_{ki}\left(\frac{Z_i^\pi(l_{\le k}T)}{kT/K}\right). \tag{EC.6}$$

Proposition EC.2 below establishes that we can construct an equivalent instance in which targets are per-arrival, as in Appendix C.1. Namely, for $k \in [K], i \in [m]$, define deviation cost function $\widetilde{g}_{ki}(a) = \frac{1}{l_{\le k}} \cdot \frac{k}{K} g_{ki}\left(\frac{l_{\le k}}{k/K}a\right)$. Consider an instance $\mathcal{I}_{new}$ defined by the set of epochs $[K]$ and deviation cost functions $(\widetilde{g}_{ki}, k \in [K], i \in [m])$.

PROPOSITION EC.2. *Fix a sample path of arrivals ω and a sequence of assignment decisions, denoted by Z^π. Then,*

$$V^\pi[\omega] = \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z^\pi_{ji}(T) + \sum_{k\in[K]}\sum_{i\in[m]} l_{\le k} T \widetilde{g}_{ki}\left(\frac{Z^\pi_i(l_{\le k}T)}{l_{\le k}T}\right).$$

We omit the straightforward proof of this fact. The following corollary emerges as a direct result of Proposition EC.2.

COROLLARY EC.2. *Consider the algorithm that takes in the original instance defined by per-period targets, converts it into a "per-arrival" instance, and runs* NS-PROXY-DUAL-GD *on the per-arrival instance. For $\eta = \Theta(\sqrt{K_{new}/T})$, this algorithm incurs $\mathbb{E}[\text{REG}] = O(K_{new}^{5/2}\sqrt{T})$, where K_{new} is defined as in Appendix C.1.*

Appendix D: Omitted Proofs

D.1. Proof of Proposition 1

Proof. Consider an instance in which there is $m = 1$ resource, $n = 1$ type and deterministic assignment cost $c_1 = -1$. Let $K = 2$, with deviation cost functions $g_1(a) = \delta a$ and $g_2(a) = \delta(1-a)$, for all $a \in [0,1]$, and $\delta > 1$ (i.e., epochs 1 and 2 respectively have target consumptions 0 and 1). For ease of notation, we suppress the dependence of all quantities on the type and resource. Plugging the assignment and deviation costs for this instance into (1), for any policy π we have:

$$\begin{aligned} V^\pi[\omega] &= -Z^\pi(T) + \delta\cdot\frac{T}{2}\left(\frac{Z^\pi(T/2)}{T/2} + 2\left(1 - \frac{Z^\pi(T)}{T}\right)\right) \\ &= \left(-Z^\pi(T/2) + \delta\cdot\frac{T}{2}\cdot\frac{Z^\pi(T/2)}{T/2}\right) + \left(-Z^\pi(T/2:T) + \delta\cdot\frac{T}{2}\cdot 2\left(1 - \frac{Z^\pi(T/2)+Z^\pi(T/2:T)}{T}\right)\right). \end{aligned} \tag{EC.7}$$

Notice that the first term in parentheses is lower bounded by 0, since $\delta > 1$. Again using $\delta > 1$, we have that the optimal myopic solution in epoch 1 is to reject all arrivals, yielding $V_1^{\text{MYO-OFF}}[\omega_1] = 0$. Moreover, since π has myopic vanishing regret, this implies that, in epoch 1:

$$\mathbb{E}\left[-Z^\pi(T/2) + \delta\cdot\frac{T}{2}\cdot\frac{Z^\pi(T/2)}{T/2}\right] = O(T^\alpha) \iff \mathbb{E}[Z^\pi(T/2)] = O(T^\alpha).$$

Taking the expectation of $V^\pi[\omega]$ over the randomness in the algorithm's decisions, plugging these facts back into (EC.7) and re-arranging, there exists a constant $C > 0$ such that:

$$\begin{aligned} \mathbb{E}[V^\pi[\omega]] &\ge \mathbb{E}\left[-Z^\pi(T/2:T) + \delta\cdot\frac{T}{2}\cdot 2\left(1 - \frac{Z^\pi(T/2:T)}{T}\right)\right] - \delta C T^\alpha \\ &\ge V_2^{\text{MYO-OFF}}[\omega_2 \mid 0] - \delta C T^\alpha, \end{aligned}$$

where the second inequality follows from the fact that $Z^\pi(T/2:T)$ is feasible for $V_2^{\text{MYO-OFF}}[\omega_2 \mid 0]$, and removes the expectation as there is no randomness in the arrival sequence.

Hence, it suffices to show that $V_2^{\text{MYO-OFF}}[\omega_2 \mid 0] - V^{\text{OFF}}[\omega] = \Omega(T)$. Re-writing the objective of the myopic offline optimum for epoch 2, we have:

$$V_2^{\text{MYO-OFF}}[\omega_2 \mid 0] = \min\ -Z(T/2:T) + \delta T\left(1 - \frac{Z(T/2:T)}{T}\right) = -T/2 + \delta T/2,$$

achieved at $Z(T/2:T) = T/2$.

For the true offline optimum (which optimizes across both epochs), we have:

$$V^{\text{OFF}}[\omega] = \min\ -Z(T) + \delta\left(Z\left(\frac{T}{2}\right) + (T - Z(T))\right) = \min\ \delta T - Z(T) - \delta \cdot Z\left(\frac{T}{2}:T\right) = -T + \delta T/2.$$

Putting these two equalities together, we obtain $V_2^{\text{MYO-OFF}}[\omega_2 \mid 0] - V^{\text{OFF}}[\omega] = T/2$, completing the proof. □

D.2. Proof of Proposition 2

Proof. Consider an instance in which $m = 1$, $K = 2$, and with a single type that has an assignment cost of 0. For ease of notation, in the remainder of the instance we omit the dependence of all quantities on the index of the single resource. The deviation costs for each epoch are given by $g_1(a_1) = \delta|a_1 - \rho_1|$, $g_2(a_2) = \delta|a_2 - \rho_2|$, and large $\delta > 0$. Suppose moreover that $\rho_1 \in [0,1]$ and $2\rho_2 - \rho_1 \in [0,1]$. For any such ρ_1, ρ_2, the assignment that accepts ρ_1 fraction of assignments in the first $T/2$ periods and $2\rho_2 - \rho_1$ fraction of assignments in the last $T/2$ periods achieves the two cumulative targets in each epoch.[12] Finally, let δ be large enough that Algorithm 1 yields $a_k^t = \rho_k$ for all t.[13]

Since assignment costs are uniformly 0 throughout the horizon, for this instance we have $V^{\text{OFF}}[\omega] = 0$, with the optimal solution accepting ρ_1 fraction of arrivals in the first epoch, and $2\rho_2 - \rho_1$ fraction of arrivals in the second epoch.

We analyze $\frac{\sum_{t \le T/2} x^t}{T/2}$, the average consumption of the resource in the first epoch under Algorithm 1. Since $a_1^t = \rho_1$, $a_2^t = \rho_2$ for all $t \le T/2$, the dual variables satisfy:

$$\begin{aligned} &\mu_1^{t+1} + \mu_2^{t+1} = \mu_1^t + \mu_2^t + \eta(\rho_1 + \rho_2 - 2x^t) \\ \implies &x^t = \frac{\rho_1 + \rho_2}{2} + \frac{(\mu_1^t + \mu_2^t) - (\mu_1^{t+1} + \mu_2^{t+1})}{2\eta} \\ \implies &\frac{\sum_{t \le T/2} x^t}{T/2} = \frac{\rho_1 + \rho_2}{2} + \frac{(\mu_1^1 + \mu_2^1) - (\mu_1^{T/2+1} + \mu_2^{T/2+1})}{\eta T}. \end{aligned} \tag{EC.8}$$

Notice that $x^t = \mathbb{1}\{\mu_1^t + \mu_2^t > 0\}$ for all $t \le T/2$. The dual update step in Algorithm 1 implies that $\mu_1^t + \mu_2^t$ is a process with negative drift whenever it is positive, and positive drift whenever it is negative. Since a_k^t

[12] We assume for technical simplicity that $\frac{T}{2}\rho_1$ and $\frac{T}{2}(2\rho_2 - \rho_1)$ are integral.

[13] The assumption that $a_k^t = \rho_k$ for all t is for expositional simplicity. As long as δ is a large enough constant, for any $\eta = O(1)$ there exists a constant t_0 past which $a_k^t = \rho_k$ for all $t > t_0$, which is all we need to prove our result.

and x^t are constants that do not depend on T, $\mu_1^t + \mu_2^t$ must be bounded in absolute value by a constant, for any $\eta = O(1)$. So, for $\eta T = \Theta(T^\epsilon)$, for some $\epsilon > 0$:

$$\lim_{T\to\infty} \frac{(\mu_1^1 + \mu_2^1) - (\mu_1^{T/2+1} + \mu_2^{T/2+1})}{\eta T} = 0. \tag{EC.9}$$

Plugging this fact into (EC.8), this then implies that $\frac{\sum_{t\le T/2} x^t}{T/2}$ converges to $\frac{\rho_1+\rho_2}{2}$ as T grows large.

Separately, for $\eta T = \Theta(T^{-\epsilon})$ for some $\epsilon \ge 0$, $\frac{\sum_{t\le T/2} x^t}{T/2}$ either converges to $\frac{\rho_1+\rho_2}{2}$, or fails to converge.

In the second epoch, similar arguments give us that $\frac{\sum_{t=T/2+1}^{T} x^t}{T/2}$ converges to ρ_2, or fails to converge as T grows large. Hence, the *running* average consumption at the end of the second epoch either converges to $\frac{0.5(\rho_1+\rho_2)}{2} + \frac{\rho_2}{2} = \frac{1}{4}\rho_1 + \frac{3}{4}\rho_2$, or fails to converge.

In all cases, there is at least a constant gap between the target and actual running average consumptions in both epochs, for large enough T. We thus obtain $\textsc{Reg}^\pi[\omega] = \Omega(T)$. □

D.3. Proof of Proposition 3

Proof. By construction, $\widetilde{Z}^\pi_{ji,k}\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right) = Z^\pi_{ji}\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)$. Using this observation, we re-write the cumulative proxy cost in epoch k as:

$$\begin{aligned}
\widetilde{V}^\pi_k\left[\omega_k \mid Z^\pi\left(\frac{(k-1)T}{K}\right)\right] &= \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} \sum_{k'\ge k} \widetilde{Z}^\pi_{ji,k}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right) \\
&\quad + \frac{T}{K}\sum_{k'\ge k}\sum_{i\in[m]} k' g_{k'i}\left(\frac{Z^\pi_i\left(\frac{(k-1)T}{K}\right) + \widetilde{Z}^\pi_{i,k}\left(\frac{(k-1)T}{K} : \frac{k'T}{K}\right)}{k'T/K}\right) \\
&= \left(\sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z^\pi_{ji}\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)\right) + \frac{T}{K}\sum_{i\in[m]} k g_{ki}\left(\frac{Z^\pi_i(kT/K)}{kT/K}\right) \\
&\quad + \left(\sum_{j\in[n]}\sum_{i\in[m]} c_{ji}\widetilde{Z}^\pi_{ji,k}\left(\frac{kT}{K} : T\right)\right) \\
&\quad + \frac{T}{K}\sum_{k'>k}\sum_{i\in[m]} k' g_{k'i}\left(\frac{Z^\pi_i\left(\frac{kT}{K}\right) + \widetilde{Z}^\pi_{i,k}\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right).
\end{aligned}$$

Summing over all k and applying the definition of $V^\pi[\omega]$ (see (1)) we obtain:

$$\begin{aligned}
\sum_{k\in[K]} \widetilde{V}^\pi_k\left[\omega_k \mid Z^\pi\left(\frac{(k-1)T}{K}\right)\right] &= V^\pi[\omega] + \left(\sum_{k\in[K]}\sum_{j\in[n]}\sum_{i\in[m]} c_{ji}\widetilde{Z}^\pi_{ji,k}\left(\frac{kT}{K} : T\right)\right) \\
&\quad + \sum_{k\in[K]}\frac{T}{K}\sum_{k'>k}\sum_{i\in[m]} k' g_{k'i}\left(\frac{Z^\pi_i\left(\frac{kT}{K}\right) + \widetilde{Z}^\pi_{i,k}\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right).
\end{aligned} \tag{EC.10}$$

Re-arranging, we obtain the result. □

D.4. Proof of Lemma 1

Proof. Fix $k \leq K-1$. We use the proxy assignments made in epoch k to construct a feasible solution to $\widetilde{V}^{\text{OFF}}_{k+1}\left[\omega_{k+1} \mid Z^\pi\left(\frac{kT}{K}\right)\right]$. For all $k' \geq k+1$, define the solution that proxy assigns a type-j arrival to resource i in epoch k' in the same proportion as our algorithm did in epoch k, i.e.,

$$Z_{ji}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right) = \left\lfloor \Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) \cdot \frac{\widetilde{Z}^\pi_{ji,k}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right)}{\Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)} \right\rfloor. \tag{EC.11}$$

Under this solution, the proxy assignment costs are upper bounded by:

$$\begin{aligned}
&\sum_{j\in[n]}\sum_{i\in[m]} c_{ji} \sum_{k'\geq k+1} Z_{ji}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right) \\
&\leq \left(\sum_{j\in[n]}\sum_{i\in[m]} c_{ji} \sum_{k'\geq k+1} \Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) \cdot \frac{\widetilde{Z}^\pi_{ji,k}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right)}{\Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)}\right) + nmc_{\max}(K-k) && \text{(EC.12)} \\
&= \sum_{j\in[n]}\sum_{i\in[m]} c_{ji}\widetilde{Z}^\pi_{ji,k}\left(\frac{kT}{K} : T\right) \\
&\quad + \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} \sum_{k'\geq k+1} \widetilde{Z}^\pi_{ji,k}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right)\left(\frac{\Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right)}{\Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)} - 1\right) + nmc_{\max}(K-k) && \text{(EC.13)} \\
&\leq \sum_{j\in[n]}\sum_{i\in[m]} c_{ji}\widetilde{Z}^\pi_{ji,k}\left(\frac{kT}{K} : T\right) \\
&\quad + Kmc_{\max} \sum_{j\in[n]} \left|\Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) - \Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)\right| + nmc_{\max}(K-k), && \text{(EC.14)}
\end{aligned}$$

where (EC.12) bounds the loss from relaxing the integrality of (EC.11), and (EC.13) adds and subtracts $\widetilde{Z}^\pi_{ji,k}\left(\frac{kT}{K} : T\right) = \sum_{k'\geq k+1} \widetilde{Z}^\pi_{ji,k}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right)$. Finally, (EC.14) follows from factoring $\Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)$ in the denominator, bounding the difference by its absolute value and using that $\frac{\widetilde{Z}^\pi_{ji,k}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right)}{\Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)} \leq 1$ for $k' \geq k+1$ by construction.

We apply similar arguments to the proxy deviation costs under this solution. In particular, since $g_{k'i}$ is Lipschitz, for all $k' \geq k+1$:

$$\begin{aligned}
\frac{k'T}{K} \cdot g_{k'i}\left(\frac{Z^\pi_i\left(\frac{kT}{K}\right) + Z_i\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right) &\leq \frac{k'T}{K} \cdot g_{k'i}\left(\frac{Z^\pi_i\left(\frac{kT}{K}\right) + \widetilde{Z}^\pi_{i,k}\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right) \\
&\quad + L\left|\sum_{j\in[n]} \Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) \cdot \frac{\sum_{k''=k+1}^{k'} \widetilde{Z}^\pi_{ji,k}\left(\frac{(k''-1)T}{K} : \frac{k''T}{K}\right)}{\Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)}\right. \\
&\quad \left. - \sum_{j\in[n]} \sum_{k''=k+1}^{k'} \widetilde{Z}^\pi_{ji,k}\left(\frac{(k''-1)T}{K} : \frac{k''T}{K}\right)\right| + Ln(k'-k)
\end{aligned}$$

$$\leq \frac{k'T}{K} \cdot g_{k'i}\left(\frac{Z_i^\pi\left(\frac{kT}{K}\right) + \widetilde{Z}_{i,k}^\pi\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right)$$
$$+ L(k'-k)\sum_{j\in[n]}\left|\Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) - \Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)\right|$$
$$+ Ln(k'-k), \tag{EC.15}$$

where the extra additive loss of $Ln(k'-k)$ is due to rounding (EC.11).

Summing (EC.15) over all i and $k' \geq k+1$, and adding to (EC.14), we obtain:

$$\widetilde{V}_{k+1}^{\text{OFF}}\left[\omega_{k+1} \mid Z^\pi\left(\frac{kT}{K}\right)\right]$$
$$\leq \left(\sum_{j\in[n]}\sum_{i\in[m]} c_{ji}\sum_{k'\geq k+1} Z_{ji}\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right)\right) + \left(\sum_{k'\geq k+1}\sum_{i\in[m]}\frac{k'T}{K}\cdot g_{k'i}\left(\frac{Z_i^\pi\left(\frac{kT}{K}\right) + Z_i\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right)\right)$$
$$\leq \sum_{j\in[n]}\sum_{i\in[m]} c_{ji}\widetilde{Z}_{ji,k}^\pi\left(\frac{kT}{K} : T\right) + \frac{T}{K}\sum_{k'\geq k+1}\sum_{i\in[m]} k' g_{k'i}\left(\frac{Z_i^\pi\left(\frac{kT}{K}\right) + \widetilde{Z}_{i,k}^\pi\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right)$$
$$+ \left(Kmc_{\max} + LK^2 m\right)\sum_{j\in[n]}\left|\Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) - \Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)\right|$$
$$+ nmc_{\max}K + LnmK^2, \tag{EC.16}$$

where the first inequality follows from the fact that Z is a feasible solution to $\widetilde{V}_{k+1}^{\text{OFF}}\left[\omega_{k+1} \mid Z^\pi\left(\frac{kT}{K}\right)\right]$. Letting $C = Lm + mc_{\max}$, we obtain the result. □

D.5. Proof of Lemma 2

Proof. Recall, by Proposition 3:

$$V^\pi[\omega] = \sum_{k\in[K]} \widetilde{V}_k^\pi\left[\omega_k \mid Z^\pi\left(\frac{(k-1)T}{K}\right)\right]$$
$$- \left[\sum_{k\in[K]}\sum_{j\in[n]}\sum_{i\in[m]} c_{ji}\widetilde{Z}_{ji,k}^\pi\left(\frac{kT}{K} : T\right) + \sum_{k\in[K]}\frac{T}{K}\sum_{k'>k}\sum_{i\in[m]} k' g_{k'i}\left(\frac{Z_i^\pi\left(\frac{kT}{K}\right) + \widetilde{Z}_{i,k}^\pi\left(\frac{kT}{K} : \frac{k'T}{K}\right)}{k'T/K}\right)\right].$$

Applying Lemma 1 to the above, we have:

$$V^\pi[\omega] \leq \sum_{k\in[K]} \widetilde{V}_k^\pi\left[\omega_k \mid Z^\pi\left(\frac{(k-1)T}{K}\right)\right] - \left(\sum_{k\leq K-1} \widetilde{V}_{k+1}^{\text{OFF}}\left[\omega_{k+1} \mid Z^\pi\left(\frac{kT}{K}\right)\right]\right) \tag{EC.17}$$
$$+ CK^2\sum_{k\leq K-1}\sum_{j\in[n]}\left|\Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) - \Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)\right| + nCK^2$$
$$\implies \underbrace{V^\pi[\omega] - \widetilde{V}_1^\pi[\omega_1] - \sum_{k\geq 2}\widetilde{\text{REG}}_k[\omega_k]}_{(I)} \leq CK^2\sum_{k\leq K-1}\sum_{j\in[n]}\left|\Lambda_j\left(\frac{kT}{K} : \frac{(k+1)T}{K}\right) - \Lambda_j\left(\frac{(k-1)T}{K} : \frac{kT}{K}\right)\right|$$
$$+ nCK^2, \tag{EC.18}$$

where (EC.18) follows from the definition of $\widetilde{\text{REG}}_k[\omega_k]$ as the difference between the proxy online costs and the proxy offline benchmark for epoch k. Taking expectations on both sides, we have:

$$\mathbb{E}[(I)] \leq CK^2 \sum_{k\leq K-1}\sum_{j\in[n]} \mathbb{E}\left[\left|\Lambda_j\left(\frac{kT}{K}:\frac{(k+1)T}{K}\right) - \Lambda_j\left(\frac{(k-1)T}{K}:\frac{kT}{K}\right)\right|\right] + nCK^2.$$

Let $Y_{jk} = \Lambda_j\left(\frac{(k-1)T}{K}:\frac{kT}{K}\right)$, and observe that $\mathbb{E}[Y_{jk}] = \mathbb{E}[Y_{j,k+1}]$ since Y_{jk} and $Y_{j,k+1}$ are i.i.d. Then,

$$\begin{aligned}
\mathbb{E}[(I)] &\leq CK^2\left(\sum_{k\leq K-1}\sum_{j\in[n]} \mathbb{E}\left[\left|Y_{jk} - \mathbb{E}[Y_{jk}]\right|\right] + \mathbb{E}\left[\left|Y_{j,k+1} - \mathbb{E}[Y_{j,k+1}]\right|\right]\right) + nCK^2 \\
&\leq 2CK^2 \sum_{k\leq K-1}\sum_{j\in[n]} \sqrt{\mathbb{E}\left[\left(Y_{jk} - \mathbb{E}[Y_{jk}]\right)^2\right]} + nCK^2 \qquad \text{(EC.19)} \\
&= 2CK^2 \sum_{k\leq K-1}\sum_{j\in[n]} \sqrt{\text{Var}[Y_{jk}]} + nCK^2, \qquad \text{(EC.20)}
\end{aligned}$$

where (EC.19) uses the fact that Y_{jk} and $Y_{j,k+1}$ are i.i.d., along with Jensen's inequality. Recalling that arrivals are drawn from a multinomial distribution with constant p_j (with respect to T), we have that $\text{Var}[Y_{jk}] = O\left(\frac{T}{K}\right)$. Plugging this back into (EC.20) we obtain our final bound, i.e.

$$\begin{aligned}
\mathbb{E}[V^\pi[\omega]] &\leq \mathbb{E}\left[\widetilde{V}_1^\pi[\omega_1] + \sum_{k\geq 2}\widetilde{\text{REG}}_k[\omega_k]\right] + O\left(K^{5/2}\sqrt{T}\right) \\
&= \sum_{k\in[K]} \mathbb{E}\left[\widetilde{\text{REG}}_k[\omega_k]\right] + \mathbb{E}\left[\widetilde{V}_1^{\text{OFF}}[\omega_1]\right] + O\left(K^{5/2}\sqrt{T}\right),
\end{aligned}$$

where the final equality follows from adding and subtracting $\mathbb{E}\left[\widetilde{V}_1^{\text{OFF}}[\omega_1]\right]$. □

D.6. Proof of Lemma 3

Proof. As in the proof of Lemma 2, we construct a feasible solution to $\widetilde{V}_1^{\text{OFF}}[\omega_1]$ using an optimal solution to $V^{\text{OFF}}[\omega]$. For any such optimal solution, let $Z_{ji}^{\text{OFF}}\left(\frac{(k-1)T}{K}:\frac{kT}{K}\right)$ denote the number of type j arrivals assigned to resource i in epoch k. Then, for all $k \in [K]$, define

$$Z_{ji,1}\left(\frac{(k-1)T}{K}:\frac{kT}{K}\right) = \left\lfloor \Lambda_j(T/K) \cdot \frac{Z_{ji}^{\text{OFF}}\left(\frac{(k-1)T}{K}:\frac{kT}{K}\right)}{\Lambda_j\left(\frac{(k-1)T}{K}:\frac{kT}{K}\right)} \right\rfloor,$$

the solution that proxy assigns the type j arrival to resource i for epoch k in the same proportion as the offline solution. We can similarly bound the proxy assignment costs under this solution:

$$\sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z_{ji,1}(T) \leq \sum_{j\in[n]}\sum_{i\in[m]} c_{ji} Z_{ji}^{\text{OFF}}(T) + mc_{\max}\sum_{j\in[n]}\sum_{k\in[K]}\left|\Lambda_j\left(\frac{T}{K}\right) - \Lambda_j\left(\frac{(k-1)T}{K}:\frac{kT}{K}\right)\right| + CK, \qquad \text{(EC.21)}$$

for some constant $C > 0$, where the additional $O(K)$ loss is due to rounding the feasible solution, as in the proof of Lemma 2.

For the proxy deviation costs, we similarly have:

$$\frac{T}{K}\sum_{k\in[K]}\sum_{i\in[m]} kg_{ki}\left(\frac{Z_i(kT/K)}{kT/K}\right) \le \frac{T}{K}\sum_{k\in[K]}\sum_{i\in[m]} kg_{ki}\left(\frac{Z_i^{\text{OFF}}(kT/K)}{kT/K}\right) + Lm\sum_{j\in[n]}\sum_{k\in[K]}\sum_{k'\le k}\left|\Lambda_j\left(\frac{T}{K}\right) - \Lambda_j\left(\frac{(k'-1)T}{K} : \frac{k'T}{K}\right)\right| + CnK^2. \tag{EC.22}$$

Putting (EC.21) and (EC.22) together, taking expectations on both sides, and applying Jensen's inequality, as in Lemma 2, we obtain:

$$\mathbb{E}\left[\widetilde{V}_1^{\text{OFF}}[\omega_1]\right] \le \mathbb{E}[V^{\text{OFF}}[\omega]] + O(K^{3/2}\sqrt{T}).$$

□

D.7. Proof of Lemma 4

Proof. To see the reduction from our proxy problem to the Balseiro et al. (2025) setup, consider an online resource allocation problem over a horizon of length T/K, with $(K-k+1)m$ resources, indexed by (k',i), for $k' \ge k$, $i \in [m]$, with budget $b_{k'i} = 1$. For each arrival, the decision-maker makes an assignment decision $x^t = (\tilde{x}^t_{k'i}, k' \ge k, i \in [m])$, where $(\tilde{x}^t_{k'i}, i \in [m]) \in \mathcal{X}^t$ for all $k' \ge k$. For this decision x^t, it incurs a bounded linear cost $f^t(x^t)$, defined by:

$$f^t(x^t) = \sum_{k'\ge k}\sum_{i\in[m]} \tilde{x}^t_{k'i}\left(\sum_{j\in[n]} \mathbb{1}\{j = j^t\}c_{ji}\right).$$

There is a single regularizer on the time-average consumption of all $(K-k+1)m$ resources:

$$r(a) = \sum_{k'\ge k}\sum_{i\in[m]} k' g_{k'i}\left(\frac{z_i}{k'T/K} + \frac{\sum_{k''=k}^{k'} a_{k'',i}}{k'}\right).$$

Note that in this case, since $a \in [0,1]^{(K+1-k)\times m}$, the set of dual variables for which the conjugate of r is bounded is $\mathcal{D} = \mathbb{R}$, and it is easy to see that Assumption 2 in Balseiro et al. (2025) holds.[14] Then, applying a stepsize $\eta = \Theta\big((T/K)^{-1/2}\big)$ to Theorem 1 in Balseiro et al. (2025), we obtain the claim. □

Appendix E: Computational Experiments: Additional Details

E.1. Benchmark Implementations

Algorithm 4 describes MYOPIC-EPOCH and SMART MYOPIC-EPOCH, for a given epoch k. Given epoch k, these two algorithms assume the decision-making horizon is $\mathcal{T}_k = \{(k-1)T/K + 1, \ldots, kT/K\}$, and take as input the epoch-specific targets $\tilde{\rho}_k$, respectively given in (9) and (10). They run Algorithm 3 over this shorter horizon, disregarding all future targets.

[14] The idealized consumption problem in the dual mirror descent algorithm in Balseiro et al. (2025) optimizes over all $a \le b$; in contrast, we also impose a nonnegativity constraint on our auxiliary variables. All of the same arguments go through assuming nonnegativity.

ALGORITHM 4: Myopic Benchmark Algorithms Implemented in Section 5

Input: Current epoch k, stepsize η, initial shadow prices $\mu^1 \in \mathbb{R}^m$, target consumptions $\tilde{\rho}_k \in [0,1]^m$

Output: Sequence of assignment decisions

for $t = (k-1)T/K + 1, \ldots, kT/K$ **do**

Observe arrival $j = j^t$, and make assignment decision:

$$x^t \in \arg\min_{x \in \mathcal{X}^t} \quad \sum_{i \in [m]} x_i (c_{ji} - \mu_i^t) \tag{EC.23}$$

Compute idealized average consumption:

$$a^t \in \arg\min_{a \in [0,1]^m} \quad \delta \sum_{i \in [m]} |a_i - \tilde{\rho}_{ki}| + \sum_{i \in [m]} \mu_i^t a_i \tag{EC.24}$$

Update shadow prices via OGD step:

$$\mu^{t+1} = \mu^t + \eta(a^t - x^t). \tag{EC.25}$$

end

E.2. Description of Maximum Likelihood Estimation Procedure for Case-Breaking Dataset

Recall, we assume that the number of arriving shipments in each two-minute interval is drawn from a Poisson distribution with parameter λ, where λ is the average number of shipments observed in the dataset for each two-minute interval. Then, under the soft-min approximation (12), the ideal quantity $q_i^*(t)$ is drawn independently from a Poisson distribution with parameter λ_i, defined as:

$$\lambda_i = \lambda \sum_{j \in [n]} p_j \frac{e^{-c_{ji}}}{1 + \sum_{i' \in [m]} e^{-c_{ji'}}}.$$

We abuse notation and use T to denote the number of two-minute intervals across the 14 days represented in the dataset. We moreover let $q^* = (q_i^*(t),\ \forall\ i \in [m], \forall\ t \in [T])$ denote the vector of all ideal quantities observed in the data, across all two-minute intervals and resources. The associated likelihood function is given by:

$$\begin{aligned} \mathcal{L}(q^*; p, c) &= \prod_{t=1}^{T} \prod_{i=1}^{m} e^{-\lambda_i} \cdot \frac{\lambda_i^{q_i^*(t)}}{q_i^*(t)!} \\ \implies \log \mathcal{L}(q^*; p, c) &= -\sum_{i=1}^{m} \lambda_i T + \sum_{t=1}^{T} \sum_{i=1}^{m} q_i^*(t) \log \lambda_i - \sum_{t=1}^{T} \sum_{i=1}^{m} \log(q_i^*(t)!), \end{aligned} \tag{EC.26}$$

where we hide the explicit dependence of λ_i on (p, c) for ease of notation.

We find the maximum likelihood estimates $(\hat{p}, \hat{c})$ by running projected gradient descent on the negative log-likelihood (EC.26), constraining $c_{ji} \in [-50, 50]$ for all $j \in [n], i \in [m]$. We let the procedure run for $n_{\text{iter}} = 10^4$ iterations, with a stepsize $\eta = 0.2$. Due to the non-convexity of the objective in (p, c), we run

δ	Algorithm 2	SMART-ME
0.001	0.009	0.007
0.01	0.034	0.291
0.1	0.841	49.989
1	17.687	2727.329

Table EC.1 Relative cost difference (in %) versus OFF, for $\delta \in \{0.001, 0.01, 0.1, 1\}$

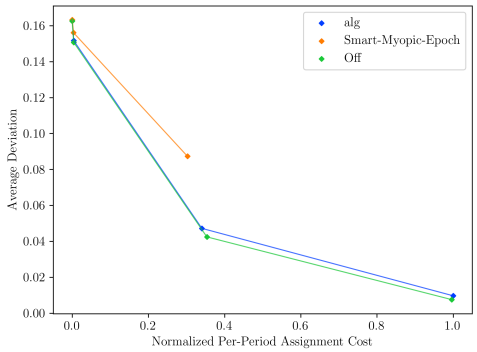


Figure EC.1 Mean absolute deviation from target versus normalized assignment cost per period. Dots from left to right correspond to deviation scalars $\delta = 0.001, 0.01, 0.1, 1$**, respectively.**

the procedure for 1,000 random initial solutions, and choose the solution associated with the highest log-likelihood.

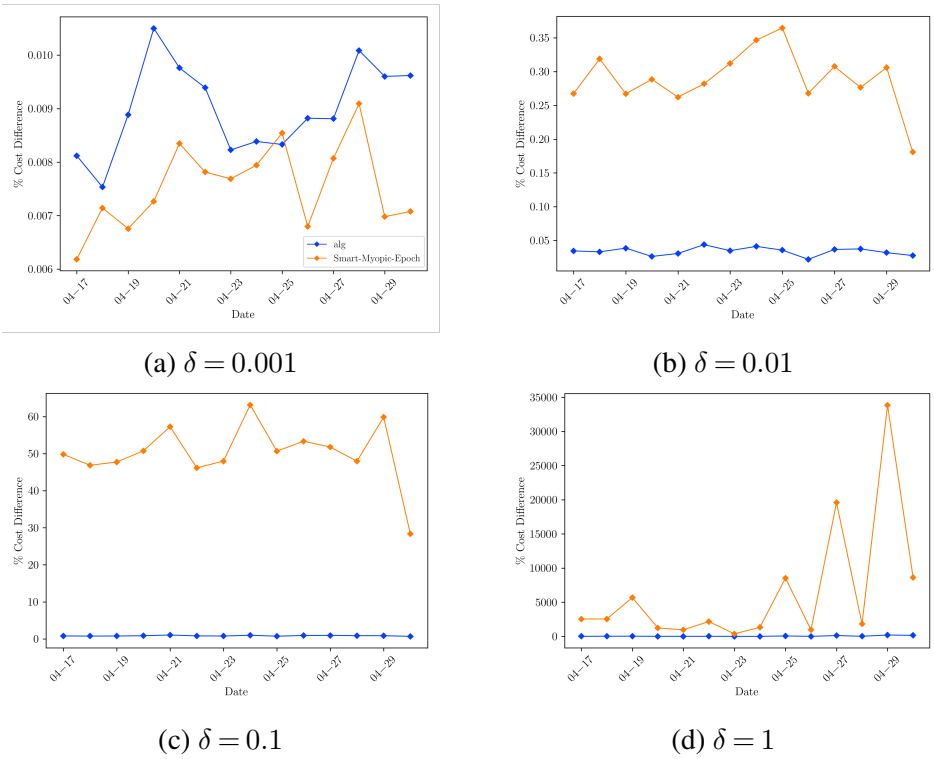
E.3. Additional Results for Case-Breaking Dataset

In this section we include results for the second warehouse in the dataset, for the same experimental setup as in Section 5.2.

As for the first warehouse, Table EC.1 demonstrates our policy's strong performance relative to the myopic target-following algorithm. The insights are similar: for negligible values of δ, both policies are very close to the offline optimum, with performance deteriorating as δ increases.

Compared to Warehouse 1 data, for $\delta \leq 0.1$, both policies perform better relative to the offline optimum. For $\delta = 1$, on the other hand, their performance is significantly worse (8 percentage point decrease in performance for our policy, and an over 5-fold decrease for the myopic policy). Figure EC.1 helps to explain this, as compared to Figure 6. For $\delta = 0.1$, the offline optimum is more biased toward assignment cost-minimizing solutions, incurring slightly higher average target deviation per period. Given that target-following is the main challenge for any online algorithm, both policies are indeed expected to perform better in this case. For $\delta = 1$, however, we are back in a case where the optimal solution remains very close to the hourly target, on average. This is more challenging for both policies to achieve, as the assignment costs for Resources 1 and 3 are significantly lower, on average ($(c_1, c_3) = (-0.74, -0.001)$ for Warehouse 2, as opposed to $(c_1, c_3) = (-0.33, 0.21)$ for Warehouse 1).

Finally, Figure EC.2 illustrates the strong performance of our policy across *all* instances for the majority of values of δ, as for Warehouse 1. The variability in the myopic policy's performance is even more evident for this warehouse when $\delta = 1$, with the myopic policy incurring over 350 times the cost of the offline optimum on April 29. Figure EC.3 shows the targets and running average assignments under the myopic policy throughout that day. Similar to the bad instance for Warehouse 1, in this instance the respective targets of Resources 1 and 3 are flipped, with the cheaper resource (Resource 1) having a lower target than the more expensive resource (Resource 3) for the majority of the horizon. In this case, the myopic policy fails to see the consequences of respecting this swap in future epochs, and overshoots assignments to Resource 1 by over 15 percentage points for the majority of the horizon.



(a) $\delta = 0.001$ (b) $\delta = 0.01$

(c) $\delta = 0.1$ (d) $\delta = 1$

Figure EC.2 **Daily relative cost difference (in %) versus OFF, for $\delta \in \{0.001, 0.01, 0.1, 1\}$. The blue and orange curves respectively represent the performance of Algorithm 2 and SMART-ME for each instance.**

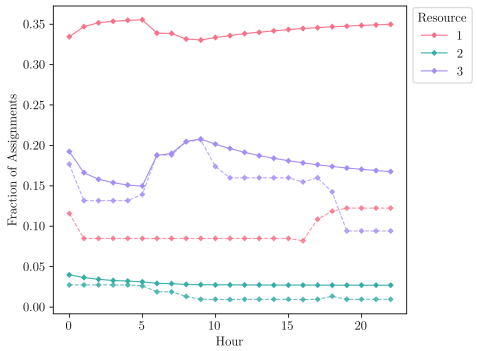


Figure EC.3 **Running average assignment across resources under SMART-ME throughout a single day in the dataset (the bad instance), for $\delta = 1$. Dotted lines correspond to hourly targets for each resource.**