Deadline-Constrained Online Allocation

Kamiar Asgari*

Abstract

We study online allocation in a lossless scalar storage system with adversarial arrivals in $[0, 1]$ and a hard service deadline. Each arrival must be consumed fully within the next H actions. The controller chooses its action before observing the current arrival and a jointly convex cost of storage and consumption. Over a horizon of $T \geq H$ rounds, we measure regret against the best fixed allocation of every arrival across the H admissible service ages. Our controller organizes service ages in a balanced binary tree and chooses each allocation split only when its interval begins. This progressive allocation guarantees deadline feasibility and regret $O(L_u\sqrt{HT} + L_x H^{3/2}\sqrt{T})$, where L_x, L_u bound the state and action components of joint subgradients. We also derive a prediction-error guarantee for causal forecasts and show how uniformly positive storage marginal costs can justify a shorter deadline, with an explicit correction for the finite cost horizon. A lower bound matches the storage-dependent term when state marginal costs may have either sign. For consumption-only costs, we prove an $\Omega(L_u\sqrt{T})$ lower bound; the optimal dependence on the deadline remains open.

1 Introduction

Online allocation with storage requires a controller to decide both how much resource to use and when to use it. Resource feasibility alone does not ensure timely service: a controller may consume only available resource while deferring its use indefinitely. Even a bound on expected backlog need not guarantee a maximum waiting time for each arrival. A hard service deadline imposes an individual obligation: every arrival must be fully served within a prescribed number of rounds. This paper studies how to learn from changing costs while meeting that obligation.

We consider a lossless scalar storage system with arrivals in $[0, 1]$. Each arrival becomes available in the next round and must be consumed completely within the next H actions. At each round, the controller chooses its consumption before observing the current arrival and cost function. The cost is jointly convex in the stored resource and the action, allowing holding and consumption costs to interact. Both arrivals and costs may be adversarial.

The benchmark consists of policies that allocate the same fraction of every arrival to each admissible service age. With a two-round deadline, for example, a fixed policy could consume half of each arrival in the next round and half in the following round. Regret is the difference between the controller's cumulative cost and that of the best fixed age-allocation policy chosen in hindsight. Each policy is evaluated on its own storage and consumption trajectory under the same realized arrivals and costs.

Delay-constrained packet scheduling provides a concrete motivation. In a fluid communication buffer, arrivals represent incoming data, consumption represents transmission, and the deadline bounds the service delay of each arrival. Spreading transmissions over time can reduce a convex

*AI usage disclaimer: Apart from the overarching idea for the problem setup, which was provided by the author, this paper was generated entirely by AI using Astra GPT-6 at the author's request.

energy cost, but every arrival must still be transmitted on time. Fixed age-allocation policies correspond to scheduling filters. Section 4.1 develops this connection and its modeling assumptions.

The central difficulty is that a service decision affects costs over several rounds. Delaying consumption changes both its service cost and the intervening storage costs. Committing to an entire schedule upon arrival would fix every allocation before any of these costs is observed. Our controller instead organizes the service ages in a balanced binary tree. At each internal node, it divides the assigned resource between earlier and later age intervals, choosing the split only when that node's interval begins. Later splits can therefore use newly observed costs. Conservation at every split guarantees complete service by the deadline, while delayed mirror-descent updates learn how to allocate resource between the two child intervals.

1.1 Contributions and organization

The analysis combines established scheduling-filter and sequential-learning ideas with a causal service-age construction. In particular, the fixed filter class has precedents in deadline scheduling [29], and the local-to-global regret identity specializes laminar regret decomposition [17]. Our claims concern the resulting guarantees for the storage model and benchmark specified below; Section 2 details the closest overlaps and distinctions.

Let T be the cost horizon, assume $1 \le H \le T$, and let L_x, L_u bound the state and action components of joint subgradients. Our main results are as follows.

1. **Deadline feasibility and adversarial regret.** The controller serves every arrival by its deadline and competes with the full class of fixed age-allocation policies. For jointly convex state–action costs, Theorem 6.5 gives the regret bound

$$O\Big(L_u\sqrt{HT} + L_x H^{3/2}\sqrt{T}\Big).$$

   For fixed H, L_x, L_u, this yields $O(\sqrt{T})$ regret.

2. **Shorter deadlines under positive storage costs.** If the storage marginal cost has a known positive lower bound, accumulated holding costs can outweigh the benefit of delaying consumption. We give an explicit service deadline $K \le H$ that can shorten the allocation horizon while retaining the original benchmark. Theorem 7.3 bounds the resulting regret and accounts for arrivals near the end of the cost horizon through an additional comparison term of at most $L_u K$ when $K < H$.

3. **Prediction and correction.** Causal forecasts guide the current allocation split, and the observed costs subsequently update the learner. Theorem 6.4 bounds regret in terms of errors in local marginal-cost comparisons between earlier and later service. Section 8 constructs such predictors from cost forecasts and known state-cost components.

4. **Lower bounds and the remaining gap.** For $H \ge 3$ and $T \ge 4$, Theorem 10.2 proves an $\Omega(L_x H^{3/2}\sqrt{T})$ lower bound using linear storage costs whose marginals may have either sign. Thus the storage-dependent term is unavoidable in the worst case. For consumption-only costs with $H, T \ge 2$, Theorem 10.1 gives an $\Omega(L_u\sqrt{T})$ lower bound, leaving a gap of $\sqrt{H}$ from the upper bound. The optimal deadline dependence in this case remains open.

The paper proceeds from related work in Section 2 to the model and applications in Sections 3 and 4. Sections 5 and 6 present the controller and its regret analysis. The shorter-deadline and prediction refinements follow in Sections 7 and 8. Sections 9 and 10 address implementation and lower bounds; Sections 11 and 12 discuss future directions and conclude.

2 Related Work

The relevant literature spans deadline scheduling, online control, optimization with memory, and regret minimization on trees. Several ingredients used here have established precedents: finite scheduling filters, simplex policy parameters, local regret decompositions, delayed updates, and optimistic mirror descent. The results of this paper concern their use in a specific combination: lossless storage with adversarial arrivals, complete service of every arrival within a common deadline, jointly convex costs revealed after the action, and regret against the entire class of fixed age distributions evaluated on their own trajectories. We distinguish that combination from its individual ingredients below.

2.1 Deadline-feasible scheduling filters

Khojastepour and Sabharwal [29] provide a particularly close antecedent. They study power-efficient communication subject to a maximum delay for each packet and represent feasible schedulers using finite, potentially time-varying linear filters. Their time-invariant class has nonnegative coefficients summing to one. With the appropriate shift for when arrivals become available, this is precisely the fixed age-allocation class in Section 3.3. Under i.i.d. arrivals and their long-run Gaussian-channel power criterion, their optimal time-invariant scheduler is the uniform moving-average filter, independently of the arrival distribution. Thus neither normalized scheduling filters nor their deadline-feasibility interpretation originates in the present paper.

Xu et al. [45] develop a related finite-filter viewpoint for server scheduling, studying workload smoothing and response-time constraints through frequency-domain analysis. These applications show that the filter interpretation extends beyond communication buffers. Our benchmark is the best fixed filter on the realized sequence, whereas the controller constructs a time-varying service allocation. Its learning guarantee concerns arbitrary revealed-after-action convex costs, including costs of the queue itself. The specific additional mechanism is to choose each interval split when that interval begins, so that later choices for the same arrival can use intervening observations. The older filter representation alone does not supply the regret analysis for these progressively chosen splits.

2.2 Energy-efficient processor and packet scheduling

Yao, Demers, and Shenker [46] established the classical speed-scaling model: jobs have release times, workloads, and deadlines, and processor power is a convex function of speed. Bansal, Kimbrel, and Pruhs [10] develop competitive algorithms and bounds for energy and temperature management. The consumption-only specialization of our model has the same basic incentive to spread divisible service over time, and a common relative deadline is a special case of job deadlines. These works compare energy with an unrestricted offline schedule under a prescribed power function. Our comparator is restricted to fixed age distributions, while the cost function may change adversarially and is observed only after service. Consequently, the competitive ratios and the additive regret bounds answer different questions. Their competitive guarantees do not directly establish additive regret to the fixed-filter benchmark under our information pattern.

For wireless transmission, Uysal-Biyikoglu, Prabhakar, and El Gamal [44] analyze energy-efficient packet scheduling and lazy transmission policies. Fu, Modiano, and Tsitsiklis [21] study energy and deadline constraints over fading channels, including a model in which current channel information is unavailable until after transmission. Lee and Jindal [33] study the complete transmission of a fixed amount of data within a finite number of slots under causal channel information.

The finite-horizon stochastic formulations of Fu et al. and Lee and Jindal establish important precedents for allocating service across a deadline window. Our formulation has overlapping arrival cohorts and adversarial revealed costs, with complete service imposed deterministically. Post-action revelation by itself is therefore not the distinction from all earlier packet-scheduling work.

2.3 Learning and prediction under service deadlines

Learning-based delay-constrained scheduling also predates this work. Salodkar et al. [43] learn energy-efficient policies over fading channels without knowing the arrival and channel distributions, using stochastic approximation with an average-delay constraint. An average-delay guarantee does not imply the individual completion requirement used here. Cayci and Eryilmaz [12] analyze learning for dynamic coding with deadline-constrained traffic and unknown channel statistics. Their regret concerns the revenue of channel-allocation policies, including the possibility of untransmitted packets, rather than obligatory complete service of each arrival.

More directly, Mlika, Driouch, and Ajib [38] combine competitive and learning algorithms for deadline scheduling in energy-harvesting networks. Their bandit learner selects among a finite set of frame-level energy-allocation rules to increase the number of users served on time. This is a precedent for regret guarantees in deadline scheduling, but users may be dropped and the benchmark is a finite collection of rules. Liu and Fang [35] learn which task types to activate when completion times and rewards are unknown; their deadline mechanism interrupts tasks, together with a throughput constraint. Neither interpretation of a deadline is equivalent to conservation of every divisible arrival across its allowed service ages.

Predictions have also been studied in models with full service and common relative deadlines. Bamas et al. [9] give learning-augmented speed-scaling algorithms, including a model in which each workload must be processed within a common delay. Their guarantees interpolate between accurate workload predictions and worst-case competitive performance. Antoniadis, Jabbarzade, and Shahkarami [4] develop a different prediction setup for online speed scaling. These works optimize a known energy law and assess error in predicted jobs or workloads. Here, forecasts predict local comparisons of marginal service and storage costs, and the bound is additive regret against fixed age filters. Shortening a virtual deadline to create scheduling slack, as in learning-augmented speed scaling, is also distinct from our comparison argument that bounds the cost of shortening the comparator's support using positive holding marginals.

Kong et al. [30] study deadline-aware fine-tuning on spot and on-demand computing instances, with forecasts of market conditions and regret for selecting among scheduling policies. The finite policy-selection benchmark and computing-allocation model differ from our continuum of age distributions. In another recent preprint, Zhang and Mo [48] combine per-slot safety for a fluid peak-age-of-information state with static regret under a viability assumption. Their modeled age safety requires additional assumptions to imply packet-level guarantees. It does not express the same obligation as completing every arrival in the present storage system. The distinction lies in the service obligations, information pattern, and comparator, rather than in the general combination of deadlines and no-regret learning.

2.4 Deferrable loads, queueing, and energy storage

Neely, Saber Tehrani, and Dimakis [39] allocate renewable energy to delay-tolerant consumers using Lyapunov optimization and a virtual queue that yields a worst-case delay bound. Their model allows purchases from an outside supply and relates average performance to the resulting delay bound, with extensions to arbitrary sample paths. Thus queue-based control can guarantee

maximum delay, not only average backlog. Here the deadline is an exogenous requirement, arrivals are the resource to be served, and the performance comparison is finite-horizon regret to a fixed scheduling filter.

Gan et al. [22] use model predictive control for deferrable loads with arrival uncertainty and updated renewable-generation forecasts. Their model allows heterogeneous deadlines and charging-rate limits and studies aggregate-load variability. Lechowicz et al. [32] study online smoothed demand management with energy storage, flexible demand deadlines, and penalties for changing purchase and delivery rates. They obtain competitive guarantees and study learning from historical instances. These models have physical features absent here, including additional capacity or smoothness constraints. Our common-deadline, divisible-service model isolates a different question: learning with arbitrary jointly convex state–action costs while retaining exact per-arrival conservation.

Asgari and Neely [6] combine Bregman-based online optimization with energy-harvesting constraints and unknown i.i.d. energy arrivals. Their guarantees connect regret to battery capacity. Chau, Zhang, and Chen [13] study energy-storage management through worst-case competitive guarantees against a fully informed offline policy. Neither battery availability alone nor a bound on its capacity enforces a maximum residence time for each arriving unit. Conversely, compulsory consumption is not an innocuous assumption for all energy-storage applications: the application must have a reason to serve every arrival by its deadline.

2.5 Simplex disturbance-action policies and nonstochastic control

Agarwal et al. [2] develop online control with adversarial disturbances and convex state–action costs using disturbance-action policies, with regret against strongly stable linear controllers. Golowich et al. [23] use simplex-structured policies for online population control. These works establish both the disturbance-filter approach and the use of simplex geometry in online control. In our problem the filter coordinates index service ages and must conserve each arrival, rather than population proportions or unconstrained disturbance responses.

The recent work of Asgari and Neely [7] is especially close: it treats scalar storage, adversarial nonnegative arrivals, revealed-after-action joint costs, and fixed simplex disturbance-action comparators. Its main dynamics have retention $\alpha \in (0, 1)$; filter coefficients form a subprobability simplex, and clipping ensures online resource availability. Our fixed policies agree algebraically with the unit-sum, $\alpha = 1$ boundary of that filter family. However, its main regret theorem relies on contraction and does not specialize to $\alpha = 1$. Its discussion of noncontracting storage instead adds a stabilizing consumption baseline. Such a baseline excludes, for example, for $H > 1$, the pure age-H policy before its service time. It also does not enforce complete consumption of every arrival at a finite age. The present deadline guarantee and comparison with all of Δ_H therefore require a separate argument; the simplex representation and general adversarial storage-learning formulation are shared antecedents.

2.6 Exact constraints versus cumulative violation

Online convex optimization with long-term constraints, exemplified by Mahdavi, Jin, and Yang [37], trades regret against cumulative constraint violation. Liu, Yang, and Ying [36] study nonstochastic control with adversarial and static constraints through constrained optimization with memory. Bounds on cumulative violation cannot by themselves certify every individual service deadline.

There are also online-control algorithms that maintain exact safety. Li, Das, and Li [34] use tightened policy sets to satisfy affine state and action constraints, under assumptions including a

strictly safe controller. Zhou and Tzoumas [50] study nonstochastic control under time-varying convex safety constraints. Thus exact feasibility with regret is already an established goal and result in constrained control. Our distinction is the particular conservation construction and comparator: every split preserves assigned resource, leaf service exhausts it by the deadline, and all age distributions remain admissible. The construction works from an empty store and allows operation on the boundary, without a strict feasibility margin. Merely expressing availability as an affine constraint does not establish the required age-by-age completion property.

2.7 Optimization with memory and the meaning of regret

Anava, Hazan, and Mannor [3] study convex losses that depend on several past decisions. Kumar, Dean, and Kleinberg [31] extend this viewpoint to unbounded memory and characterize regret through the influence of past decisions. Their finite-memory specialization is also relevant to the deadline dependence: Appendix D, Theorem D.1 gives a lower bound of order $Lh^{3/2}\sqrt{T}$ for a general memory-h loss class with an ℓ_1 history norm. Accordingly, the exponent $3/2$ is not new as a general memory phenomenon. The lower bound in Theorem 10.2 realizes this order within a more restricted class: scalar storage, linear signed state costs, exact deadlines, and fixed age-policy comparators. A generic memory lower bound does not automatically apply to every physically constrained subclass.

Memory methods also clarify why convexity of the fixed-policy objective is insufficient for an immediate online algorithm. Updating a filter parameter every round need not conserve the resource assigned by earlier parameters; actual service and state are determined by past commitments. The per-arrival representation explicitly records those commitments. Zhao et al. [49] study changing comparators through dynamic policy regret in optimization with memory and nonstochastic control. Our age-distribution comparator stays fixed; adversarially changing costs and progressively changing online decisions do not turn the guarantee into dynamic regret.

There is a further distinction when the environment reacts to actions. Arora, Dekel, and Tewari [5] distinguish ordinary regret on the realized loss sequence from policy regret that reevaluates an adaptive adversary along alternative histories. Here, the realized arrival and cost sequence is held fixed for all comparisons, while each fixed age policy generates its own state. Thus our guarantee addresses the physical trajectory dependence for a fixed realized environment; it does not compare different reactions of an adaptive environment to counterfactual policies.

2.8 Tree decompositions and local regret minimization

Zinkevich et al. [51] introduced counterfactual regret minimization for extensive-form games. Farina, Kroer, and Sandholm [17] develop laminar regret decomposition for sequential decision processes and convex losses. The principle is to combine local learners at decision nodes, evaluating each alternative with the learner's continuation decisions. Our binary tree identity is a specialization of this local-to-global viewpoint: the comparator's mass M_I weights the local difference, and the node feedback compares the two child continuation values. Computing hypothetical feedback even at unused nodes is consistent with that interpretation. The identity and the general idea of decomposing regret on a tree are not separate novelty claims here.

Tree-based geometry and prediction have likewise been studied together. Hoda et al. [26] construct smoothing geometries for sequential-game strategy spaces. Farina et al. [16] combine predictive local learners with counterfactual regret minimization, while Farina, Kroer, and Sandholm [18, 19] study optimistic updates and improved regularization on sequential decision spaces. Fan, Kroer, and Farina [15] analyze dilated entropy and regret lower bounds for extensive-form

games. These results provide context for the local quadratic and entropy choices, but the associated strategy-space or equilibrium guarantees do not directly give a physical service-age bound. In particular, node-dependent feedback copies mean that our implementation should not be identified without further argument with one synchronous global mirror-descent step.

The additional structure is temporal. Nodes correspond to consecutive service ages, their decisions begin at different physical times, and feedback is complete only after their intervals end. The same interval length d_I controls both feedback delay and the storage-cost difference between descendant ages. Balancing the tree makes the sums of $\sqrt{d_I}$ and $d_I^{3/2}$ along a path scale respectively as $\sqrt{K}$ and $K^{3/2}$ for a tree on K ages. This is the source of the separate action and storage contributions, rather than a generic dependence on the number of tree nodes. The proof also has to link these local losses to the actual jointly convex storage cost and to physically feasible service.

2.9 Delayed feedback, optimism, and predictions

Mirror descent and its Bregman geometry are classical [11]. Joulani, György, and Szepesvári [27] provide general reductions for learning with delayed feedback. The separate learner copies used here are a bounded-delay device: a copy is reused only after its preceding comparison has become available. The physical round cost is observed immediately after service, but the complete comparison of two service intervals depends on later rounds. It is this derived feedback, rather than the original cost observation, that is delayed. Quanrud and Khashabi [41] and Joulani, György, and Szepesvári [28] analyze other delay-tolerant gradient and mirror-descent methods. Separate copies are therefore a convenient implementation here, rather than a necessary feature of all delayed-learning algorithms.

Rakhlin and Sridharan [42] develop learning with predictable sequences. Flaspohler et al. [20] study optimism and delay together, showing how predictions can offset missing feedback. These are direct methodological antecedents of our prediction–correction bound. More recently, Abdullah et al. [1] combine constraints, memory, and predictions through a delayed optimistic formulation, with regret and cumulative-violation guarantees. The present contribution in this direction is the choice of local marginal-cost prediction targets and their aggregation under exact deadline feasibility. Predictions are causal and may be inaccurate; the bound measures their realized local errors rather than assuming access to future costs. Cheng et al. [14] also provide a predictor–corrector approach to policy optimization with error-dependent regret. The prediction–correction principle is shared; the service obligations and local continuation comparisons are specific to the present construction.

2.10 Inventory learning and budgeted online allocation

Inventory learning provides another connection between optimization and state-dependent implementability. Zhang, Chao, and Shi [47] study learning a base-stock policy for perishable inventory under censored demand, with holding, lost-sales, and outdating costs. Hihat et al. [25] extend inventory learning beyond i.i.d. demand under non-degeneracy conditions. Pineci and Xu [40] develop a hidden-target approach for feasible inventory learning on general convex sets. In these problems the controller replenishes stock to meet demand, and unsold items may persist or perish. Here arrivals are exogenous and the controller must consume all of them on time. A product's shelf life therefore should not be identified with our mandatory service deadline.

Budgeted online allocation is also related in its use of mirror descent, but the resource obligation is different. Balseiro, Lu, and Mirrokni [8] study dual mirror descent for reward-maximizing allocation under total resource budgets, with guarantees across stochastic and adversarial input models. A total budget limits aggregate resource expenditure; it does not require each incoming

unit to be allocated completely within a residence-time bound. The shared term "online allocation" therefore covers different feasibility structures and benchmarks.

2.11 Scope of the contribution

The contribution supported by these comparisons is a deadline-feasible instantiation and analysis of progressive allocation for adversarial joint costs, with the explicit bound $O(L_u\sqrt{HT} + L_x H^{3/2}\sqrt{T})$ against all fixed age filters. The accompanying results identify a matching signed-storage lower bound inside this model, construct local cost predictors, and bound the finite-horizon cost of reducing the comparator's deadline under positive holding marginals. We make no separate priority claim for scheduling filters, simplex control, hierarchical regret decomposition, delayed mirror descent, or the general coexistence of deadlines and learning. The matching lower bound applies to signed state marginals; it does not settle the optimal deadline dependence for consumption-only costs or for the class with nonnegative state marginals.

3 Model, Information, and Benchmark

We write $\mathbb{R}_+$ for the nonnegative reals, $[n] = \{1, \dots, n\}$, $\mathbb{E}$ for expectation, and $\mathbf{1}_{\{A\}}$ for the indicator of an event A. Bold symbols denote vectors, and ∂f denotes the subdifferential. All logarithms are natural unless a base is shown. Empty sums are zero, and $0 \log 0 = 0$ in entropy expressions. For scalar clipping, write $[v]_\ell^r = \min\{r, \max\{\ell, v\}\}$.

3.1 Lossless dynamics and joint costs

Let T be the cost horizon and H the consumption deadline, with integers $1 \le H \le T$ throughout. Write x_t, u_t, w_t for the stored resource, consumption, and arrival in round t. The system starts empty and evolves as

$$x_1 = 0, \qquad x_{t+1} = x_t - u_t + w_t, \qquad 0 \le u_t \le x_t, \qquad w_t \in [0, 1]. \tag{3.1}$$

At the beginning of round t, the controller knows the current state and all arrivals and costs revealed through round $t-1$. It selects u_t before observing w_t and c_t. After the action, the arrival and the entire cost function are revealed, and the controller incurs $c_t(x_t, u_t)$. The state in this cost is the beginning-of-round state; the arrival w_t first enters x_{t+1}.

Each c_t is a finite jointly convex function on

$$\mathcal{D}_H = \{(x, u) : 0 \le u \le x \le H\}. \tag{3.2}$$

Assume a joint-subgradient oracle: at every $(x, u) \in \mathcal{D}_H$, it can return one pair (ζ^x, ζ^u) such that

$$c_t(x', u') \ge c_t(x, u) + \zeta^x(x' - x) + \zeta^u(u' - u) \quad \text{for all } (x', u') \in \mathcal{D}_H, \tag{3.3}$$

and

$$|\zeta^x| \le L_x, \qquad |\zeta^u| \le L_u, \tag{3.4}$$

for known finite constants $L_x, L_u \ge 0$. At the realized pair write

$$\boldsymbol{\zeta}_t \coloneqq (\zeta_t^x, \zeta_t^u)^\top \in \partial c_t(x_t, u_t) \tag{3.5}$$

for an oracle selection satisfying these bounds; its components are the state and action marginal costs. For nonsmooth costs the two numbers must belong to the *same joint subgradient*; unrelated

coordinatewise subgradients need not satisfy (3.3). Applying (3.3) in both directions between two feasible pairs and using (3.4) gives

$$|c_t(x,u) - c_t(x',u')| \le L_x|x-x'| + L_u|u-u'|. \tag{3.6}$$

The entire cost $c_t(x,u)$ may vary adversarially and may couple x and u. The residual-inventory example in Section 4.2 illustrates an admissible nonseparable cost and its signed marginals. The regret bound holds for every realized arrival and cost sequence, including sequences chosen in response to the controller. Once this sequence has been produced, we keep it fixed when evaluating each comparison policy. Each policy generates its own storage and consumption trajectory, but all policies face the same arrivals and cost functions.

3.2 Per-arrival service deadlines

Arrival w_s becomes available for the actions at times $s+1, s+2, \ldots$. Let $q_{s,i}$ be its fraction actually consumed at age i. Thus t indexes rounds, s arrivals, and i ages. Feasible deadline accounting requires

$$q_{s,i} \ge 0, \qquad \sum_{i=1}^{H} q_{s,i} = 1, \qquad u_t = \sum_{i=1}^{H} w_{t-i} q_{t-i,i}, \tag{3.7}$$

with $w_s = 0$ for $s \le 0$. There is no service of w_s outside $s+1, \ldots, s+H$. The complete vector $\mathbf{q}_s$ need not be chosen upon arrival: a causal controller may choose its fractions progressively, provided the service at each time is determined before that time's cost is revealed. For $w_s = 0$, any such allocation vector can be used.

Proposition 3.1 (State identity and exact feasibility)**.** *For any causal implementation of* (3.7)*,*

$$x_t = \sum_{i=1}^{H} w_{t-i} \sum_{j=i}^{H} q_{t-i,j}. \tag{3.8}$$

The dynamics (3.1) *hold, every arrival is completely consumed by its deadline, and* $0 \le u_t \le x_t \le H$*. If, for some integer* $1 \le K \le H$*, all fractions have support in* $1, \ldots, K$*, then* $0 \le u_t \le x_t \le K$*.*

Proof. At the beginning of round t, the unserved part of arrival w_{t-i} is

$$w_{t-i}\left(1 - \sum_{j=1}^{i-1} q_{t-i,j}\right) = w_{t-i} \sum_{j=i}^{H} q_{t-i,j}.$$

Summing over the possible ages gives (3.8), and

$$\begin{aligned} 0 \le u_t &= \sum_{i=1}^{H} w_{t-i} q_{t-i,i} \\ &\le \sum_{i=1}^{H} w_{t-i} \sum_{j=i}^{H} q_{t-i,j} = x_t. \end{aligned}$$

The state identity also gives

$$\begin{aligned} x_t - u_t + w_t &= w_t + \sum_{i=1}^{H} w_{t-i} \sum_{j=i+1}^{H} q_{t-i,j} \\ &= w_t + \sum_{i=2}^{H} w_{t+1-i} \sum_{j=i}^{H} q_{t+1-i,j} \\ &= \sum_{i=1}^{H} w_{t+1-i} \sum_{j=i}^{H} q_{t+1-i,j} = x_{t+1}, \end{aligned}$$

using $\sum_{j=1}^{H} q_{t,j} = 1$. Moreover,

$$0 \le \sum_{j=i}^{H} q_{t-i,j} \le 1, \qquad x_t \le \sum_{i=1}^{H} w_{t-i} \le H.$$

If $q_{s,i} = 0$ for $i > K$, the sums reduce to $1 \le i \le K$, giving $x_t \le K$. Since $\sum_{i=1}^{H} w_s q_{s,i} = w_s$, every arrival is exhausted by age H (by age K under the support restriction). □

3.3 Fixed probability-simplex policies

Define

$$\Delta_H = \left\{ \mathbf{m} \in \mathbb{R}_+^H : \sum_{i=1}^{H} m_i = 1 \right\}. \tag{3.9}$$

A fixed $\mathbf{m} \in \Delta_H$ assigns the same schedule to every arrival:

$$u_t(\mathbf{m}) = \sum_{i=1}^{H} m_i w_{t-i}, \qquad x_t(\mathbf{m}) = \sum_{i=1}^{H} w_{t-i} \sum_{j=i}^{H} m_j. \tag{3.10}$$

Both coordinates are affine in $\mathbf{m}$, and the trajectory is feasible from time one. The equivalent cumulative coordinates satisfy $y_i = \sum_{j=1}^{i} m_j$ and $0 = y_0 \le y_1 \le \cdots \le y_H = 1$. In these coordinates, the state formula becomes

$$x_t(\mathbf{m}) = \sum_{i=1}^{H} w_{t-i}(1 - y_{i-1}),$$

which expresses the stored resource through cumulative consumption, with no storage decay and complete consumption by age H.

Write

$$J_T^{\mathrm{on}} = \sum_{t=1}^{T} c_t(x_t, u_t), \qquad J_T(\mathbf{m}) = \sum_{t=1}^{T} c_t(x_t(\mathbf{m}), u_t(\mathbf{m})), \tag{3.11}$$

and

$$\mathrm{Reg}_T(\Delta_H) = J_T^{\mathrm{on}} - \min_{\mathbf{m} \in \Delta_H} J_T(\mathbf{m}). \tag{3.12}$$

Continuity from (3.6) and compactness ensure existence of the minimizer. The minimization chooses a common age distribution in hindsight. All cost functions and arrivals in (3.11) are the same realized functions and arrivals, but each policy uses its own state.

3.4 Service and costs after the horizon

Set $w_t = 0$ for $t > T$ and execute all outstanding service through the actual deadlines. Costs after T are excluded from (3.11); equivalently, use zero cost functions and zero gradients there solely for analysis and updates. If the costs of serving the remaining arrivals after the horizon are part of the objective, include those rounds in T for both policies.

For nontrivial learning let

$$N = T - 1. \tag{3.13}$$

For each integer $1 \le K \le H$, we identify a policy in Δ_K with its zero-padded vector in Δ_H when evaluating its trajectory and J_T. The arrival w_T cannot affect any cost through T and may be served at $T + 1$. Only arrivals $s = 1, \dots, N$ can affect costs included in the objective. Every state within this horizon is also bounded by $\min\{H, t - 1\}$.

For $T = 1$ or $H = 1$ regret is zero. The formulas below are stated for $T \ge 2$.

4 Applications and Modeling Scope

The model describes a divisible resource that can be stored without loss, with every arrival served completely within a common deadline of H rounds. There is no independent service-capacity limit beyond resource availability in (3.1): several arrivals may be served together, and no additional constraint $u_t \le 1$ is imposed. Available storage must accommodate the states permitted by (3.2).

4.1 Delay-constrained packet scheduling

In a fluid communication buffer, w_t is newly arriving data, x_t is the queued data before transmission, and u_t is the amount successfully transmitted in round t. An arrival at the end of round s must be fully transmitted during rounds $s + 1, \dots, s + H$. The constraint is therefore a maximum delay for each arrival, in addition to the requirement that the transmitter serve only data already in its buffer. Smoothing a burst over several rounds can reduce a convex transmission-energy cost, but the delay bound limits how long the scheduler can defer that service.

This is the scheduling setting motivating the connection to Khojastepour and Sabharwal's *Delay-Constrained Scheduling: Power Efficiency, Filter Design, and Bounds* [29]. The action of a fixed policy in (3.10) is a scheduling filter, with weights in (3.9). Nonnegative coefficients preserve causality and resource availability, and their unit sum ensures complete transmission within H rounds. The uniform choice $m_i = 1/H$ gives the moving-average schedule $u_t = H^{-1} \sum_{i=1}^{H} w_{t-i}$. The controller learns an adaptive allocation of each arrival across these ages and measures its performance against the best fixed filter on the realized sequence. Section 2.1 discusses the relationship to the earlier scheduling objective.

For a concrete energy-aware specialization, let

$$c_t(x, u) = \rho_t P(u) + \lambda x, \qquad 0 \le \rho_t \le \bar{\rho}, \qquad \lambda \ge 0,$$

where P is a known differentiable convex transmission-energy function with $0 \le P'(u) \le G_P$ on $[0, H]$. The coefficient ρ_t is an energy price or cost weight revealed after the action. The joint gradient is $(\lambda, \rho_t P'(u))$, so one may take $L_x = \lambda$ and $L_u = \bar{\rho} G_P$. The case $\lambda = 0$ uses the consumption-only guarantee in Corollary 6.6; Corollary 7.4 applies to the constant nonnegative state marginal cost λ. Setting $\rho_t = 1$ and $\lambda = 0$ recovers a fixed convex transmit-energy objective.

This specialization assumes successful service. Unknown channel capacity and failed transmissions would change both the service feasibility and the feedback.

4.2 Coupled penalties and the role of signed marginals

For a target residual inventory $d_t \in [0, H]$ revealed after service, consider

$$c_t(x, u) = \frac{(x - u - d_t)^2}{2H}.$$

This jointly convex cost penalizes deviations of the residual inventory $x - u$ from the target d_t. The residual is measured after service and before the new arrival. The cost's state and action marginals have magnitudes at most one, so $L_x = L_u = 1$ is valid. The state marginal can be negative when the residual is below the target. The general theorem applies, whereas the nonnegative state-marginal refinement need not. This example shows why a nonnegative *cost value* is different from a nonnegative *state marginal*.

The regret guarantee for all these applications uses the benchmark in Section 3.3.

5 Causal Allocation and Delayed Mirror Descent

Choose a deadline $1 \le K \le H$ for the controller. The general controller uses $K = H$; Section 7 gives a justified reduction when state marginals have a uniform positive lower bound.

5.1 Age intervals and causal routing

Build a balanced binary tree $\mathcal{T}_K$ on the ordered leaves $1, \dots, K$. Tree intervals $[a, b]$ denote the integer age sets $\{a, \dots, b\}$. For an internal node $I = [a, b]$, write $a_I = \min I = a$. Its length and children are

$$d_I = b - a + 1, \qquad r_I = \lfloor (a + b)/2 \rfloor + 1, \qquad I_{\mathrm{L}} = [a, r_I - 1], \quad I_{\mathrm{R}} = [r_I, b]. \tag{5.1}$$

Let $\mathcal{I}_K$ be its set of internal nodes. For arrival s, select $p_{s,I} \in [0, 1]$ at the end of round $s + a - 1$. Route fraction $p_{s,I}$ of the resource assigned to I to its left child, and the remainder to its right child. Retain right-child resource until its interval begins. Each leaf consumes all incoming resource at its age. Simultaneously starting decisions are processed from ancestors to descendants. The root receives w_s when it arrives.

The fractions are deterministic divisions of divisible resource, not randomized service choices. The final fraction $q_{s,i}$ is the product of the routing factors on the path to leaf i. All factors on that path have been chosen by the end of $s + i - 1$. They are nonnegative and their leaf products sum to one. Proposition 3.1 therefore proves causality, feasibility, and complete service by age $K \le H$, independently of how the splits are learned.

5.2 A four-age example

Figure 1 illustrates the timing for $K = 4$. Suppose one unit arrives at the end of round s. At that time, the root assigns half to ages $1, 2$ and half to ages $3, 4$. The left child immediately assigns one quarter of its incoming resource to age one. The right child's split can wait until the end of round $s + 2$, after two more costs have been observed. If that later split assigns three quarters to age three, the final allocation vector is

$$\mathbf{q}_s = (q_{s,1}, q_{s,2}, q_{s,3}, q_{s,4})^\top = \left(\frac{1}{8}, \frac{3}{8}, \frac{3}{8}, \frac{1}{8}\right)^\top.$$

This example uses illustrative split fractions to show how a feasible allocation is assembled as successive age intervals begin.

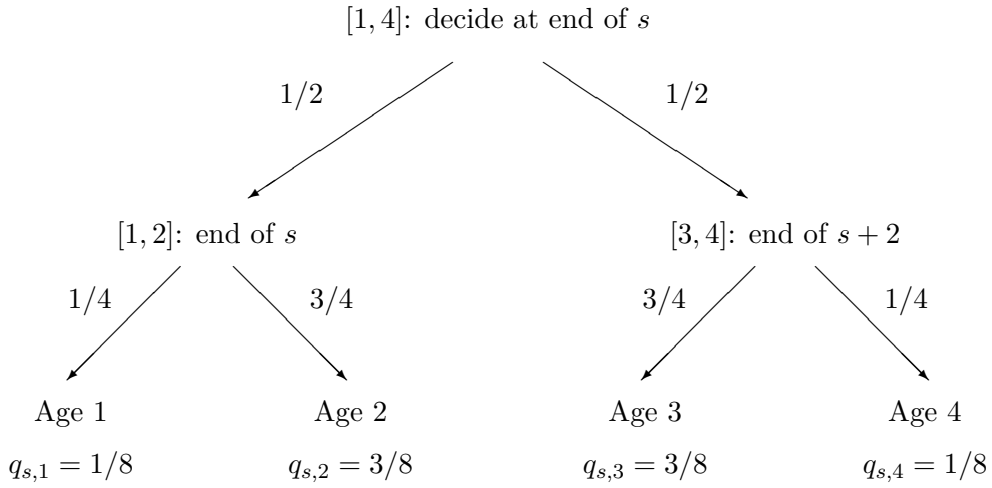


Figure 1: Allocation over four consumption ages. Edge labels are fractions of the resource entering their parent. Leaf i serves its assigned resource at time $s + i$; the right subtree can use information revealed after the earlier service decisions.

5.3 Marginal feedback from the actual joint cost

After each round within the cost horizon, obtain the joint pair (ζ_t^x, ζ_t^u) in (3.5); set both components to zero after T. For arrival $s \le N$ define

$$A_{s,i} = w_s \left(\zeta_{s+i}^u + \sum_{k=1}^{i} \zeta_{s+k}^x \right), \qquad i = 1, \dots, K. \tag{5.2}$$

In the linearized total cost, $A_{s,i}$ is the coefficient of the fraction of arrival s served at age i. Its service term comes from round $s+i$, while its storage terms come from rounds $s+1, \dots, s+i$. The storage sum includes the service round because the cost uses the beginning-of-round state. The factor w_s converts an allocation fraction into an amount of resource. All subgradients are evaluated along the controller's realized trajectory.

Define subtree values recursively by

$$V_{s,\{i\}} = A_{s,i}, \qquad V_{s,I} = p_{s,I} V_{s,I_{\mathrm{L}}} + (1 - p_{s,I}) V_{s,I_{\mathrm{R}}}. \tag{5.3}$$

The value $V_{s,I}$ is an average of the marginal-cost coefficients for ages in I, with weights determined by the controller's splits within that node's subtree. It gives the linearized cost coefficient for routing an amount w_s through I according to those splits. The scalar feedback is

$$g_{s,I} = V_{s,I_{\mathrm{L}}} - V_{s,I_{\mathrm{R}}}. \tag{5.4}$$

This feedback compares routing an amount w_s through the earlier child with routing that amount through the later child, using the observed cost slopes and the controller's splits within each child's subtree. A positive value favors sending less resource to the earlier interval on future uses of this learner; a negative value favors sending more.

At each node, we compare the costs of routing the full arrival amount through the earlier and later children. This is a hypothetical comparison, so we calculate it even when the controller sends no resource to that node. Split fractions must therefore be chosen in unused branches as well.

In the regret proof, each comparison is weighted by the fraction that the fixed comparison policy assigns to that node, denoted M_I; see Lemma 6.3.

At a node $I = [a, b]$ spanning d_I ages, complete feedback becomes available d_I rounds after the split is chosen. We therefore maintain d_I separate copies of the learning rule and use them in turn for successive arrivals. Each copy receives feedback from its previous decision before it is used again. In particular, the copy used for arrival s is next used for arrival $s + d_I$, at the end of round

$$(s + d_I) + a - 1 = s + b. \tag{5.5}$$

Bounded-delay reductions are standard [27]. The controller collects observations during the interval and updates the learner when the full comparison is available.

5.4 Updating a split with quadratic or entropy penalties

Each learner stores a reference split $z \in [0, 1]$, initially $1/2$. To choose a split, it balances the predicted cost against a penalty for moving away from z. The default penalty is the squared distance $\frac{1}{2}(p - z)^2$. We also analyze the binary relative entropy penalty $\mathrm{kl}(p\|z)$. We write $D_\psi(p, z)$ when either penalty can be used. Explicitly,

$$D_{\psi_Q}(p, z) = \frac{1}{2}(p - z)^2, \qquad D_{\psi_E}(p, z) = \mathrm{kl}(p\|z) = p \log \frac{p}{z} + (1 - p) \log \frac{1 - p}{1 - z}.$$

The entropy formula is defined for $p \in [0, 1]$ and $z \in (0, 1)$. Appendix A.1 gives the generating functions and their basic bounds. In this paper ψ denotes the regularizer and $F_{s,I}$ the predictor.

The function $F_{s,I}(p)$ predicts how the local cost changes with the fraction p sent to the earlier interval. Its derivative at the chosen split is the prediction of $g_{s,I}$ used in the regret bound. For example, a forecast $\widehat{g}$ of this marginal comparison gives the linear predictor $F_{s,I}(p) = \widehat{g}p$. The predictor must use only information available when the split is chosen.

At the decision time choose a convex differentiable predictor $F_{s,I} : [0, 1] \to \mathbb{R}$ with finite derivative satisfying this information requirement, and choose

$$p_{s,I} = \operatorname*{arg\,min}_{p \in [0,1]} \{\eta_I F_{s,I}(p) + D_\psi(p, z_{s,I})\}. \tag{5.6}$$

The basic algorithm uses $F_{s,I} = 0$, so $p_{s,I} = z_{s,I}$. When $g_{s,I}$ is complete, update the *reference value stored by the same learner* by

$$z_{s+d_I,I} = \operatorname*{arg\,min}_{z \in [0,1]} \{\eta_I g_{s,I} z + D_\psi(z, z_{s,I})\}. \tag{5.7}$$

Minimizing these scalar objectives gives the closed forms (see Appendix A.1)

$$z_{s+d_I,I} = [z_{s,I} - \eta_I g_{s,I}]_0^1, \qquad \text{quadratic,} \tag{5.8}$$

$$z_{s+d_I,I} = \frac{z_{s,I} \exp(-\eta_I g_{s,I})}{1 - z_{s,I} + z_{s,I} \exp(-\eta_I g_{s,I})}, \qquad \text{binary entropy.} \tag{5.9}$$

The prediction shifts the chosen split away from the reference $z_{s,I}$. Once feedback arrives, the next reference incorporates the observed marginal comparison starting from that original reference. For example, with quadratic regularization, a linear predictor $F(p) = \widehat{g}p$, and no active boundary constraint, the chosen split is $p = z - \eta_I \widehat{g}$. Updating from p would give $z - \eta_I(\widehat{g} + g)$, retaining the predicted step in addition to the observed step. Updating from z gives $z - \eta_I g$.

With the entropy penalty, the reference values and chosen splits remain strictly between zero and one for finite feedback and derivatives; quadratic iterates may equal zero or one. Both the split-selection and reference-update minimizations have unique solutions. The one-step bound is proved in Lemma A.1.

Algorithm 1 Deadline allocation with joint costs

Require: T, $K \le H$, tree $\mathcal{T}_K$, regularizer ψ, positive steps η_I, causal predictor rule F

If $K = 1$, serve every arrival at age one and stop.
Initialize every node's d_I reference copies to $1/2$; initialize resource and feedback records to zero.
for $t = 1, \dots, T + K$ **do**
    Execute the due leaf services, whose sum is u_t.
    if $t \le T$ **then**
        Observe w_t, c_t; incur $c_t(x_t, u_t)$; obtain one joint subgradient (ζ_t^x, ζ_t^u) at (x_t, u_t).
    else
        Set $w_t \leftarrow 0$, $\zeta_t^x \leftarrow 0$, and $\zeta_t^u \leftarrow 0$; add no new objective term.
    end if
    Set $x_{t+1} \leftarrow x_t - u_t + w_t$.
    Complete due leaf and subtree values using (5.2)–(5.3), descendants before ancestors.
    For each completed (s, I), form $g_{s,I}$ and update its copy by (5.7).
    For each starting (s, I) with $1 \le s \le N$ and $t = s + a_I - 1$, choose the split in (5.6) and route its mass, ancestors before descendants.
    If $t = T$, assign w_T to service at $T + 1$.
end for

6 Regret Analysis for General Joint Costs

6.1 Regularizer constants and a prediction-error bound at one node

The local prediction–correction estimate uses standard optimistic mirror-descent analysis [42, 20]. For the two regularizers define constants by

$$D_\psi(p, z) \ge \frac{\sigma_\psi}{2}(p - z)^2, \qquad \sup_{p \in [0,1]} D_\psi(p, 1/2) \le A_\psi, \qquad C_\psi = \sqrt{2A_\psi/\sigma_\psi}. \tag{6.1}$$

Valid values are

	σ_ψ	A_ψ	C_ψ
quadratic	1	$1/8$	$1/2$
binary entropy	4	$\log 2$	$\sqrt{\log 2/2}$

(6.2)

These constants follow from the scalar geometry bounds in Appendix A.1. Apply Lemma A.1 with regularizer ψ, predictor $F = F_{s,I}$, and feedback $g = g_{s,I}$. Writing $\widehat{g} = F'_{s,I}(p)$ and dropping the nonpositive term $-D_\psi(p, z)/\eta$ gives, for every $p^* \in [0, 1]$,

$$g(p - p^*) \le \frac{D_\psi(p^*, z) - D_\psi(p^*, z^+)}{\eta} + \frac{\eta}{2\sigma_\psi}(g - \widehat{g})^2. \tag{6.3}$$

The lemma covers quadratic boundary solutions; with the entropy penalty, the reference values and chosen splits remain strictly between zero and one because feedback and predictor derivatives are finite.

Lemma 6.1 (Local regret with delayed feedback)**.** *Set $e_{s,I} = g_{s,I} - F'_{s,I}(p_{s,I})$. For any fixed $p^* \in [0, 1]$,*

$$\sum_{s=1}^{N} g_{s,I}(p_{s,I} - p^*) \le \frac{d_I D_\psi(p^*, 1/2)}{\eta_I} + \frac{\eta_I}{2\sigma_\psi} \sum_{s=1}^{N} e_{s,I}^2. \tag{6.4}$$

If a known $B_I > 0$ bounds $|e_{s,I}|$ uniformly, the step size

$$\eta_I = \sqrt{\frac{2\sigma_\psi A_\psi d_I}{N B_I^2}} \tag{6.5}$$

gives

$$\sum_{s=1}^{N} g_{s,I}(p_{s,I} - p^*) \le C_\psi B_I \sqrt{d_I N}. \tag{6.6}$$

Proof. Fix I and abbreviate $d = d_I$, $\eta = \eta_I$. For $r \in [d]$, let

$$\mathcal{S}_r = \{s \in [N] : s \equiv r \pmod d\} = \{s_{r,1} < \cdots < s_{r,n_r}\}.$$

By (5.5), $z_{s_{r,k}+d,I} = z_{s_{r,k+1},I}$ whenever $k < n_r$. Summing (6.3) along one nonempty copy therefore gives

$$\begin{aligned}
\sum_{k=1}^{n_r} g_{s_{r,k},I}(p_{s_{r,k},I} - p^*) &\le \frac{1}{\eta}\sum_{k=1}^{n_r}\Big[D_\psi(p^*, z_{s_{r,k},I}) - D_\psi(p^*, z_{s_{r,k}+d,I})\Big] \\
&\quad + \frac{\eta}{2\sigma_\psi}\sum_{k=1}^{n_r} e_{s_{r,k},I}^2 \\
&= \frac{D_\psi(p^*, 1/2) - D_\psi(p^*, z_{s_{r,n_r}+d,I})}{\eta} + \frac{\eta}{2\sigma_\psi}\sum_{k=1}^{n_r} e_{s_{r,k},I}^2 \\
&\le \frac{D_\psi(p^*, 1/2)}{\eta} + \frac{\eta}{2\sigma_\psi}\sum_{k=1}^{n_r} e_{s_{r,k},I}^2.
\end{aligned}$$

Summing over the at most d nonempty residue classes gives (6.4). If $|e_{s,I}| \le B_I$, the step size (6.5) gives

$$\begin{aligned}
\frac{d_I A_\psi}{\eta_I} + \frac{\eta_I}{2\sigma_\psi}\sum_{s=1}^{N} e_{s,I}^2 &\le \frac{d_I A_\psi}{\eta_I} + \frac{\eta_I N B_I^2}{2\sigma_\psi} \\
&= 2B_I\sqrt{\frac{A_\psi d_I N}{2\sigma_\psi}} \\
&= C_\psi B_I \sqrt{d_I N}.
\end{aligned}$$

□

For any fixed step size, the regret bound depends on the prediction errors that actually occur. These errors need not be known in advance. To choose the step size before running the controller, we can use a known upper bound B_I on their magnitude, as in (6.5).

6.2 Joint convexity and the tree identity

Lemma 6.2 (Regret linearization)**.** *For every $\mathbf{m} \in \Delta_K$,*

$$J_T^{\mathrm{on}} - J_T(\mathbf{m}) \le \sum_{s=1}^{N}\sum_{i=1}^{K} A_{s,i}(q_{s,i} - m_i). \tag{6.7}$$

Proof. Put $\delta_{s,i} = q_{s,i} - m_i$ and retain the convention $\zeta_r^x = \zeta_r^u = 0$ for $r > T$. The subgradient inequality at the realized state–action pair gives

$$c_t(x_t, u_t) - c_t(x_t(\mathbf{m}), u_t(\mathbf{m})) \le \zeta_t^x(x_t - x_t(\mathbf{m})) + \zeta_t^u(u_t - u_t(\mathbf{m})).$$

The action contribution rearranges as

$$\begin{aligned}\sum_{t=1}^{T}\zeta_t^u\big(u_t-u_t(\mathbf{m})\big)&=\sum_{t=1}^{T}\sum_{i=1}^{K}\zeta_t^u w_{t-i}\delta_{t-i,i}\\&=\sum_{s=1}^{N}\sum_{i=1}^{K}w_s\zeta_{s+i}^u\delta_{s,i}.\end{aligned}$$

Similarly, the state contribution is

$$\begin{aligned}\sum_{t=1}^{T}\zeta_t^x\big(x_t-x_t(\mathbf{m})\big)&=\sum_{t=1}^{T}\sum_{i=1}^{K}\zeta_t^x w_{t-i}\sum_{j=i}^{K}\delta_{t-i,j}\\&=\sum_{s=1}^{N}w_s\sum_{i=1}^{K}\zeta_{s+i}^x\sum_{j=i}^{K}\delta_{s,j}\\&=\sum_{s=1}^{N}\sum_{j=1}^{K}w_s\left(\sum_{i=1}^{j}\zeta_{s+i}^x\right)\delta_{s,j}.\end{aligned}$$

Adding these contributions and using (5.2) proves the bound. For fixed realized costs, each (x_t, u_t) is affine in the final fractions; therefore $\sum_t c_t(x_t, u_t)$ is jointly convex in those fractions. □

We can describe a fixed policy using the same tree as the controller. At node I, M_I is the fraction of each arrival assigned to the ages in that interval. When $M_I > 0$, p_I^* is the proportion of that fraction assigned to the earlier child:

$$M_I=\sum_{i\in I}m_i,\qquad p_I^*=\frac{M_{I_{\mathrm{L}}}}{M_I}\quad(M_I>0).\tag{6.8}$$

If $M_I = 0$, the split can be chosen arbitrarily because the policy assigns no resource there.

The next identity expresses the linearized cost difference for one arrival as a sum of comparisons at individual nodes. Each comparison is weighted by the fraction M_I assigned there by the fixed policy. This is the binary terminal-cost specialization of the counterfactual and laminar regret decomposition viewpoint [51, 17]; we include its short proof to make the service-age analysis self-contained.

Lemma 6.3 (Tree identity)**.** *For every arrival s,*

$$\sum_{i=1}^{K}A_{s,i}(q_{s,i}-m_i)=\sum_{I\in\mathcal{I}_K}M_I(p_{s,I}-p_I^*)g_{s,I}.\tag{6.9}$$

Proof. If $M_I > 0$, then $M_I p_I^* = M_{I_{\mathrm{L}}}$ and $M_I(1 - p_I^*) = M_{I_{\mathrm{R}}}$. Hence

$$\begin{aligned}M_I(p_{s,I}-p_I^*)g_{s,I}&=(M_Ip_{s,I}-M_{I_{\mathrm{L}}})(V_{s,I_{\mathrm{L}}}-V_{s,I_{\mathrm{R}}})\\&=M_I[p_{s,I}V_{s,I_{\mathrm{L}}}+(1-p_{s,I})V_{s,I_{\mathrm{R}}}]-M_{I_{\mathrm{L}}}V_{s,I_{\mathrm{L}}}-M_{I_{\mathrm{R}}}V_{s,I_{\mathrm{R}}}\\&=M_IV_{s,I}-M_{I_{\mathrm{L}}}V_{s,I_{\mathrm{L}}}-M_{I_{\mathrm{R}}}V_{s,I_{\mathrm{R}}}.\end{aligned}$$

If $M_I = 0$, then $M_{I_{\mathrm{L}}} = M_{I_{\mathrm{R}}} = 0$, so the same identity holds for any p_I^*. Summing it over all internal nodes cancels every internal child term:

$$\begin{aligned}\sum_{I \in \mathcal{I}_K} M_I (p_{s,I} - p_I^*) g_{s,I} &= M_{\mathrm{root}} V_{s,\mathrm{root}} - \sum_{i=1}^{K} M_{\{i\}} V_{s,\{i\}} \\ &= V_{s,\mathrm{root}} - \sum_{i=1}^{K} m_i A_{s,i}.\end{aligned}$$

The routing recursion (5.3) gives

$$V_{s,\mathrm{root}} = \sum_{i=1}^{K} q_{s,i} V_{s,\{i\}} = \sum_{i=1}^{K} q_{s,i} A_{s,i}.$$

□

Theorem 6.4 (General prediction-error bound)**.** *For any causal differentiable convex predictors and fixed positive step sizes, Algorithm 1 is deadline feasible and, for every* $\mathbf{m} \in \Delta_K$*,*

$$\boxed{J_T^{\mathrm{on}} - J_T(\mathbf{m}) \le \sum_{I \in \mathcal{I}_K} M_I \left[\frac{d_I D_\psi(p_I^*, 1/2)}{\eta_I} + \frac{\eta_I}{2\sigma_\psi} \sum_{s=1}^{N} e_{s,I}^2 \right].} \tag{6.10}$$

With strictly positive error bounds B_I *and steps* (6.5)*, it also satisfies*

$$\mathrm{Reg}_T(\Delta_K) \le C_\psi \sqrt{N} \max_{1 \le i \le K} \sum_{\substack{I \in \mathcal{I}_K \\ i \in I}} B_I \sqrt{d_I}. \tag{6.11}$$

Proof. Proposition 3.1 gives deadline feasibility. For any $\mathbf{m} \in \Delta_K$, Lemmas 6.2 and 6.3 give

$$\begin{aligned}J_T^{\mathrm{on}} - J_T(\mathbf{m}) &\le \sum_{s=1}^{N} \sum_{I \in \mathcal{I}_K} M_I (p_{s,I} - p_I^*) g_{s,I} \\ &= \sum_{I \in \mathcal{I}_K} M_I \sum_{s=1}^{N} g_{s,I} (p_{s,I} - p_I^*).\end{aligned}$$

Since $M_I \ge 0$, Lemma 6.1 gives (6.10). With (6.5) and $|e_{s,I}| \le B_I$,

$$\begin{aligned}J_T^{\mathrm{on}} - J_T(\mathbf{m}) &\le C_\psi \sqrt{N} \sum_{I \in \mathcal{I}_K} M_I B_I \sqrt{d_I} \\ &= C_\psi \sqrt{N} \sum_{i=1}^{K} m_i \sum_{\substack{I \in \mathcal{I}_K \\ i \in I}} B_I \sqrt{d_I} \\ &\le C_\psi \sqrt{N} \max_{1 \le i \le K} \sum_{\substack{I \in \mathcal{I}_K \\ i \in I}} B_I \sqrt{d_I},\end{aligned}$$

because $\sum_i m_i = 1$. Apply the last display to a minimizer of J_T over Δ_K. □

6.3 General upper bounds

Consider two service ages $i < j$ in an interval I containing d_I ages. Both marginal-cost coefficients include the storage terms for ages $1, \dots, i$. These common terms cancel when we subtract $A_{s,i}$ from $A_{s,j}$, giving

$$A_{s,j} - A_{s,i} = w_s \left[\zeta^u_{s+j} - \zeta^u_{s+i} + \sum_{k=i+1}^{j} \zeta^x_{s+k} \right]. \tag{6.12}$$

The remaining sum contains $j-i$ storage terms. Since $j-i \le d_I-1$, their total absolute contribution is at most $w_s(d_I-1)L_x$. The two service terms contribute at most $2w_sL_u$. Each child-subtree value is an average of the coefficients for its service ages, so the difference between the two child values satisfies the same bound. Therefore, for the basic controller ($F = 0$),

$$|e_{s,I}| = |g_{s,I}| \le w_s G_I \le G_I, \qquad G_I = 2L_u + (d_I - 1)L_x. \tag{6.13}$$

Define

$$\Lambda_1 = 0, \qquad \Lambda_K = \sum_{j=0}^{\lceil \log_2 K \rceil - 1} \sqrt{\lceil K/2^j \rceil} \quad (K \ge 2). \tag{6.14}$$

At depth j every interval has length at most $\lceil K/2^j \rceil$. Consequently

$$\Lambda_K \le \sqrt{2K} \sum_{j \ge 0} 2^{-j/2} = 2(1+\sqrt{2})\sqrt{K}. \tag{6.15}$$

Theorem 6.5 (Adversarial jointly convex costs)**.** *Under* (3.3)–(3.4)*, use* $K = H$*,* $F = 0$*, and* (6.5) *with* $B_I = G_I$*. The controller satisfies*

$$\mathrm{Reg}_T(\Delta_H) \le C_\psi \sqrt{T-1} \max_{1 \le i \le H} \sum_{I \ni i} [2L_u + (d_I - 1)L_x] \sqrt{d_I} \tag{6.16}$$

$$\le C_\psi [2L_u + (H-1)L_x] \sqrt{T-1}\, \Lambda_H. \tag{6.17}$$

In particular, for quadratic updates,

$$\boxed{\mathrm{Reg}_T(\Delta_H) \le \frac{2L_u + (H-1)L_x}{2} \sqrt{T-1}\, \Lambda_H = O\big(L_u \sqrt{HT} + L_x H^{3/2} \sqrt{T}\big).} \tag{6.18}$$

Proof. With $K = H$ and $N = T-1$, (6.13) gives

$$B_I = G_I = 2L_u + (d_I - 1)L_x \le 2L_u + (H-1)L_x =: G_*.$$

For every leaf i, there is at most one interval containing i at each depth j, and its length is at most $\lceil H/2^j \rceil$. Therefore

$$\sum_{\substack{I \in \mathcal{I}_H \\ i \in I}} G_I \sqrt{d_I} \le G_* \sum_{j=0}^{\lceil \log_2 H \rceil - 1} \sqrt{\lceil H/2^j \rceil}$$

$$= G_* \Lambda_H.$$

Theorem 6.4 gives (6.16)–(6.17). For quadratic updates, $C_\psi = 1/2$; (6.15) then gives (6.18). □

Corollary 6.6 (Consumption-only costs)**.** *If* $L_x = 0$*, the quadratic controller satisfies*

$$\mathrm{Reg}_T(\Delta_H) \le L_u \sqrt{T-1}\, \Lambda_H \le 2(1+\sqrt{2}) L_u \sqrt{H(T-1)}. \tag{6.19}$$

Proof. Set $L_x = 0$ in (6.17) and use (6.15). State independence follows from the Lipschitz bound (3.6) with $L_x = 0$. □

For fixed H, L_x, L_u, the regret is $O(\sqrt{T})$. With fixed positive bounds and a growing deadline, $H^3 = o(T)$ is sufficient for the general upper bound to be sublinear, whereas $H = o(T)$ suffices when $L_x = 0$.

7 Deadline Selection under Nonnegative State Marginals

This section adds a structural assumption on the joint costs c_t. Assume that at every time and every feasible pair the oracle can select a joint subgradient with

$$0 \le \lambda \le \zeta^x \le \lambda + V, \qquad |\zeta^u| \le L_u, \qquad \lambda, V \ge 0. \tag{7.1}$$

The constant λ is a lower bound on the marginal cost of holding additional resource. The quantity V specifies how far above λ this marginal cost may rise. These same bounds apply at every round and every feasible stored and consumed amount. The bounds are known, although the cost functions themselves need not be known in advance.

7.1 A comparison inequality for coupled costs

Lemma 7.1 (Moving consumption earlier)**.** *Let two feasible pairs satisfy $\bar{x} \le x$. Under* (7.1)*,*

$$c_t(\bar{x}, \bar{u}) - c_t(x, u) \le -\lambda(x - \bar{x}) + L_u|\bar{u} - u|. \tag{7.2}$$

Proof. Use a bounded joint subgradient $(\bar{\zeta}^x, \bar{\zeta}^u)$ at $(\bar{x}, \bar{u})$ in (3.3), tested against (x, u):

$$\begin{aligned} c_t(\bar{x}, \bar{u}) - c_t(x, u) &\le \bar{\zeta}^x(\bar{x} - x) + \bar{\zeta}^u(\bar{u} - u) \\ &\le \lambda(\bar{x} - x) + |\bar{\zeta}^u|\,|\bar{u} - u| \\ &\le -\lambda(x - \bar{x}) + L_u|\bar{u} - u|, \end{aligned}$$

where the second line uses $\bar{x} - x \le 0$ and $\bar{\zeta}^x \ge \lambda$. □

To justify a shorter deadline, compare service at an age $i > K$ with service at age one. If the age-i service occurs by time T, moving one unit changes consumption in two rounds. Lemma 7.1 bounds the contribution of these action changes to the cost increase by $2L_u$. The move also removes $i - 1 \ge K$ rounds of storage, whose contribution to the same bound is at most $-\lambda K$. This motivates choosing K so that $\lambda K \ge 2L_u$ whenever the deadline is shortened. The next lemma accounts separately for arrivals whose service extends beyond the cost horizon. Choose

$$K = \begin{cases} H, & \lambda = 0, \\ \min\left\{H, \max\left\{1, \left\lceil \dfrac{2L_u}{\lambda} \right\rceil \right\}\right\}, & \lambda > 0. \end{cases} \tag{7.3}$$

This choice also gives $\lambda(K - 1) \le 2L_u$, so the part of the local feedback bound (7.7) proportional to λ is at most $2L_u$. When $K < H$, it satisfies $\lambda K \ge 2L_u$ as required for the preceding cost comparison. For $K = 1$ there are no split decisions to learn.

Lemma 7.2 (Cost of shortening the comparator's deadline)**.** *For any* $\mathbf{m} \in \Delta_H$*, move all its fractions at ages* $i > K$ *to age one, producing* $\bar{\mathbf{m}} \in \Delta_K$*. Then*

$$J_T(\bar{\mathbf{m}}) - J_T(\mathbf{m}) \le L_u K \, \mathbf{1}_{\{K<H\}}. \tag{7.4}$$

No modification is needed when $K = H$*.*

Proof. For $i > K$ and $s \le T-1$, put

$$v_{s,i} = w_s m_i, \qquad h_s = T - s.$$

Replacing age-i service of this amount by age-one service changes the actions within the cost horizon only at $s+1$ and, when $s+i \le T$, at $s+i$. Its action and state contributions are therefore

$$\begin{aligned}
\sum_{t=1}^{T} |\Delta u_t^{(s,i)}| &= v_{s,i}\big(1 + \mathbf{1}_{\{i \le h_s\}}\big), \\
\sum_{t=1}^{T} \big(x_t(\mathbf{m}) - x_t(\bar{\mathbf{m}})\big)^{(s,i)} &= v_{s,i} \sum_{r=2}^{\min\{i,h_s\}} 1 \\
&= v_{s,i}\big(\min\{i, h_s\} - 1\big).
\end{aligned}$$

In particular, $x_t(\bar{\mathbf{m}}) \le x_t(\mathbf{m})$ for every t. State reductions add linearly, while the action changes satisfy

$$\sum_{t=1}^{T} |u_t(\bar{\mathbf{m}}) - u_t(\mathbf{m})| \le \sum_{s=1}^{T-1} \sum_{i=K+1}^{H} v_{s,i}\big(1 + \mathbf{1}_{\{i \le h_s\}}\big).$$

After summing Lemma 7.1 over t,

$$J_T(\bar{\mathbf{m}}) - J_T(\mathbf{m}) \le \sum_{s=1}^{T-1} \sum_{i=K+1}^{H} v_{s,i} \left[L_u\big(1 + \mathbf{1}_{\{i \le h_s\}}\big) - \lambda\big(\min\{i, h_s\} - 1\big) \right]. \tag{7.5}$$

If $K = H$, the double sum is empty. Otherwise, (7.3) gives $\lambda K \ge 2L_u$, so for $i > K$,

$$L_u(1 + \mathbf{1}_{\{i \le h_s\}}) - \lambda(\min\{i, h_s\} - 1) \le \begin{cases} 2L_u - \lambda K \le 0, & h_s \ge K+1, \\ L_u - \lambda(h_s - 1) \le L_u, & 1 \le h_s \le K. \end{cases}$$

Consequently,

$$\begin{aligned}
J_T(\bar{\mathbf{m}}) - J_T(\mathbf{m}) &\le L_u \sum_{\substack{1 \le s \le T-1 \\ 1 \le T-s \le K}} \sum_{i=K+1}^{H} w_s m_i \\
&\le L_u \sum_{s=T-K}^{T-1} 1 \le L_u K.
\end{aligned}$$

□

For arrivals near the end of the cost horizon, moving service earlier can add a positive action term to the cost bound at or before T, while part of the compensating storage reduction occurs after T and is excluded from the objective. When $K < H$, only arrivals $s = T-K, \dots, T-1$ can contribute a positive excess in the preceding bound. Their total size is at most K, and each unit contributes at most L_u, giving the term $L_u K$ in (7.4). This comparison allows the controller to use deadline K while its regret is still measured against fixed policies with deadline H. The comparison bound can be replaced by zero when $K = H$, or when every nonzero arrival's full original lifetime lies inside the cost horizon.

7.2 Regret bound for the chosen deadline

For the basic controller, (6.12) and (7.1) imply

$$|g_{s,I}| \le w_s[2L_u + (\lambda + V)(d_I - 1)]. \tag{7.6}$$

A common bound is

$$B = 2L_u + (\lambda + V)(K - 1) \le 4L_u + (K - 1)V. \tag{7.7}$$

Theorem 7.3 (Regret bound with the chosen deadline)**.** *Assume* (7.1)*, use* K *in* (7.3)*,* $F = 0$*, and steps* (6.5) *with the common bound* (7.7)*. Then*

$$\boxed{\mathrm{Reg}_T(\Delta_H) \le C_\psi[2L_u + (\lambda + V)(K - 1)]\sqrt{T - 1}\,\Lambda_K + L_u K\,\mathbf{1}_{\{K<H\}}.} \tag{7.8}$$

In particular,

$$\mathrm{Reg}_T(\Delta_H) = O\big((L_u + KV)\sqrt{KT} + L_u K\big). \tag{7.9}$$

Proof. Let $\mathbf{m}^*$ minimize J_T over Δ_H, and let $\bar{\mathbf{m}}^* \in \Delta_K$ be its reduction from Lemma 7.2. With B from (7.7), (7.6) gives $|g_{s,I}| \le B$ at every node. Theorem 6.4 and the same depth calculation as in Theorem 6.5 give

$$J_T^{\mathrm{on}} - J_T(\bar{\mathbf{m}}^*) \le C_\psi B\sqrt{T - 1}\,\Lambda_K.$$

Therefore

$$\begin{aligned}
\mathrm{Reg}_T(\Delta_H) &= J_T^{\mathrm{on}} - J_T(\mathbf{m}^*) \\
&= J_T^{\mathrm{on}} - J_T(\bar{\mathbf{m}}^*) + J_T(\bar{\mathbf{m}}^*) - J_T(\mathbf{m}^*) \\
&\le C_\psi[2L_u + (\lambda + V)(K - 1)]\sqrt{T - 1}\,\Lambda_K + L_u K\mathbf{1}_{\{K<H\}}.
\end{aligned}$$

Equations (6.15) and (7.7) give (7.9). Using the node-dependent bounds in (7.6) directly in (6.11) gives the sharper path expression. □

Corollary 7.4 (Constant nonnegative state marginal)**.** *If the oracle selects* $\zeta^x = \lambda \ge 0$ *at every time and feasible pair, then the quadratic controller satisfies*

$$\mathrm{Reg}_T(\Delta_H) \le 2L_u\sqrt{T - 1}\,\Lambda_K + L_u K\,\mathbf{1}_{\{K<H\}} = O(L_u\sqrt{KT} + L_u K). \tag{7.10}$$

The implied constant is independent of λ*, and this is at most* $O(L_u\sqrt{HT})$*.*

Proof. With $V = 0$, $C_\psi = 1/2$, and (7.3),

$$C_\psi[2L_u + \lambda(K - 1)] \le \frac{1}{2}(2L_u + 2L_u) = 2L_u.$$

Apply Theorem 7.3, (6.15), and $K \le \sqrt{KT} \le \sqrt{HT}$. □

When $\lambda > 0$ and L_u are fixed, the deadline K is bounded independently of H. For example, $\lambda = L_u = 1$ gives $K = \min\{H, 2\}$. When only $\zeta^x \ge 0$ is known and no positive lower bound is available, (7.3) gives $K = H$: long zero-cost holding periods cannot be excluded by this argument. Section 10.3 shows why nonnegativity alone does not give a bound independent of the magnitude of adversarial state marginals.

8 Causal Prediction and Known State Costs

Forecasts of future costs can help the controller decide how much resource to consume in the earlier interval and how much to leave for the later interval. A forecast may use previously revealed costs, side information, or a known part of the cost function. It must be available when the split is chosen. Setting the prediction term F to zero gives the basic controller.

8.1 A hypothetical trajectory used for prediction

To use a forecast when choosing a split, we first give it a simple hypothetical consumption schedule. Part of the arrival is consumed at the start of the earlier interval, and the rest at the start of the later interval. Varying that fraction gives a predicted cost for each candidate split. The actual consumption times are still determined by later decisions in the tree.

Fix $I = [a, b]$, let $r = r_I$ be its right-child starting age, and consider arrival s. At the end of $s + a - 1$, choose a known background storage level $B^{\mathrm{bg}}_{s,I} \in [0, K - w_s]$; zero is always allowed. A hypothetical allocation of size w_s serves fraction p at age a and fraction $1 - p$ at age r. In the hypothetical states, actions, and forecast gradients below, we suppress the fixed indices (s, I); age k corresponds to physical round $s + k$. For $k = a, \dots, r$, define

$$\widehat{x}_k(p) = B^{\mathrm{bg}}_{s,I} + w_s\big[p\mathbf{1}_{\{k=a\}} + (1-p)\big], \tag{8.1}$$

$$\widehat{u}_k(p) = w_s\big[p\mathbf{1}_{\{k=a\}} + (1-p)\mathbf{1}_{\{k=r\}}\big]. \tag{8.2}$$

Every such pair is affine in p and belongs to $\mathcal{D}_K$. The background is retained and receives no service in the hypothetical schedule during this interval. The size is the full w_s, even if the node receives less actual mass, consistently with the normalization of (5.4).

For each $k = a, \dots, r$ with $s + k \le T$, let $\widehat{c}_{s,I,k}(x, u)$ be a causal differentiable jointly convex forecast for c_{s+k}. Put

$$F_{s,I}(p) = \sum_{k=a}^{\min\{r,T-s\}} \widehat{c}_{s,I,k}(\widehat{x}_k(p), \widehat{u}_k(p)), \tag{8.3}$$

with an empty sum equal to zero. Then (5.6) retains these full nonlinear functions inside a one-dimensional convex optimization problem.

To identify the predicted gradient, evaluate the forecast gradient $(\widehat{\zeta}^x_k, \widehat{\zeta}^u_k)$ at the hypothetical state–action pair for the chosen $p_{s,I}$. The chain rule gives

$$\widehat{g}_{s,I} = w_s \left[\mathbf{1}_{\{s+a\le T\}}\widehat{\zeta}^u_a - \mathbf{1}_{\{s+r\le T\}}\widehat{\zeta}^u_r - \sum_{k=a+1}^{\min\{r,T-s\}} \widehat{\zeta}^x_k \right]. \tag{8.4}$$

Omitted forecast terms are zero. Equation (5.7), centered at the old reference, performs the eventual correction using the actual $g_{s,I}$.

The forecast evaluates costs using a simplified consumption schedule and hypothetical storage levels. The actual controller may consume resource at different ages and reach different storage levels, so the resulting marginal comparisons can differ even when the forecast cost functions are exact. Thus $e_{s,I} = g_{s,I} - \widehat{g}_{s,I}$ includes both errors in the cost forecasts and differences caused by the hypothetical schedule. The regret bound in (6.10) depends on the sum of these squared feedback errors.

8.2 Error bounds and worst-case guarantees

Proposition 8.1 (Forecasts with bounded gradients)**.** *If every forecast gradient obeys* $|\widehat{\zeta}^x| \le L_x$ *and* $|\widehat{\zeta}^u| \le L_u$ *on its hypothetical pairs, then*

$$|e_{s,I}| \le 2w_s[2L_u + (d_I - 1)L_x]. \tag{8.5}$$

Thus steps based on twice the basic feedback bound preserve the general regret order of Theorem 6.5. If instead actual subgradients and forecast gradients both satisfy the bounds in (7.1)*, then*

$$|e_{s,I}| \le w_s[4L_u + (\lambda + V)(d_I - 1)]. \tag{8.6}$$

Together with (7.3)*, this preserves the rate* $O((L_u + KV)\sqrt{KT} + L_uK)$ *in Theorem 7.3.*

Proof. Write $g_{s,I} = g^u_{s,I} + g^x_{s,I}$ and $\widehat{g}_{s,I} = \widehat{g}^u_{s,I} + \widehat{g}^x_{s,I}$ for the action- and state-marginal terms. Equation (8.4) contains at most two action terms and at most $d_I - 1$ state terms, so

$$|\widehat{g}_{s,I}| \le w_s\big[2L_u + (d_I - 1)L_x\big].$$

Together with (6.13),

$$\begin{aligned} |e_{s,I}| &= |g_{s,I} - \widehat{g}_{s,I}| \\ &\le |g_{s,I}| + |\widehat{g}_{s,I}| \\ &\le 2w_s\big[2L_u + (d_I - 1)L_x\big]. \end{aligned}$$

Under (7.1), the subtree values are weighted averages of leaf values. For fixed (s, I), let π^{L}_i and π^{R}_i be the controller's final conditional leaf fractions in the left and right children, respectively, with their dependence on (s, I) suppressed. These probability weights give

$$g_{s,I} = \sum_{i \in I_{\mathrm{L}}} \sum_{j \in I_{\mathrm{R}}} \pi^{\mathrm{L}}_i \pi^{\mathrm{R}}_j (A_{s,i} - A_{s,j}).$$

For every $i \in I_{\mathrm{L}}$ and $j \in I_{\mathrm{R}}$,

$$A_{s,i} - A_{s,j} = w_s \left[\zeta^u_{s+i} - \zeta^u_{s+j} - \sum_{k=i+1}^{j} \zeta^x_{s+k}\right],$$

$$\left|w_s(\zeta^u_{s+i} - \zeta^u_{s+j})\right| \le 2L_u w_s,$$

$$-w_s(\lambda + V)(d_I - 1) \le -w_s \sum_{k=i+1}^{j} \zeta^x_{s+k} \le 0.$$

Averaging and applying (8.4) to the forecast gives

$$|g^u_{s,I}|, |\widehat{g}^u_{s,I}| \le 2L_u w_s, \qquad g^x_{s,I}, \widehat{g}^x_{s,I} \in [-w_s(\lambda + V)(d_I - 1), 0].$$

Hence

$$\begin{aligned} |e_{s,I}| &\le |g^u_{s,I} - \widehat{g}^u_{s,I}| + |g^x_{s,I} - \widehat{g}^x_{s,I}| \\ &\le w_s\big[4L_u + (\lambda + V)(d_I - 1)\big]. \end{aligned}$$

Finally,

$$4L_u + (\lambda + V)(K - 1) \le 6L_u + (K - 1)V$$

by $\lambda(K - 1) \le 2L_u$. Theorem 6.4, (6.15), and Lemma 7.2 therefore give $O((L_u + KV)\sqrt{KT} + L_uK)$. □

At each node, the error contribution in (6.10) is proportional to $\sum_s (g_{s,I} - \widehat{g}_{s,I})^2$. Accurate predictions of the marginal feedback can make this contribution small. Uniform gradient bounds alone do not ensure that improvement: they allow the actual and predicted marginals to differ by twice the basic feedback bound, which explains the larger general worst-case constant in Proposition 8.1.

These forecast bounds assume differentiable forecast functions and exact solutions of the split-selection and reference-update minimizations. The observed costs themselves may be nonsmooth.

8.3 A fixed known state-cost component

Corollary 8.2 (Known state cost)**.** *Suppose the joint cost has the structured form*

$$c_t(x, u) = \ell_t(u) + f(x), \tag{8.7}$$

where each ℓ_t is adversarial and convex with available subgradients bounded by $L_u > 0$ on $[0, H]$, while f is convex, differentiable, fixed, and known before the first decision. Assume $0 \le \lambda \le f'(x) \le \lambda + V$ on $[0, H]$ and choose K by (7.3). Then the basic controller satisfies Theorem 7.3 with the same constants.

When the storage cost f is known, we can evaluate it directly when choosing the split. In the hypothetical schedule, consuming fraction p at age a leaves $w_s(1-p)$ from this arrival in storage until age r_I. Use the causal joint forecasts $\widehat{c}_{s,I,k}(x, u) = f(x)$ in (8.3), the hypothetical states (8.1), and set

$$\gamma_{s,I} = \max\{0, \min\{r_I, T - s\} - a\}.$$

The quantity $\gamma_{s,I}$ counts the subsequent rounds through age r_I that fall within the cost horizon. It therefore determines how many times the storage cost of this remaining amount enters the prediction step. After dropping a term independent of p, the prediction step becomes

$$\boxed{p_{s,I} = \operatorname*{arg\,min}_{p \in [0,1]} \left\{ D_\psi(p, z_{s,I}) + \eta_I \gamma_{s,I} f\big(B^{\mathrm{bg}}_{s,I} + w_s(1-p)\big) \right\}.} \tag{8.8}$$

Keep the reference correction (5.7) unchanged and use the step sizes (6.5) with the bound (7.7). For either regularizer, this controller satisfies (7.8). The prediction-error bound (6.10) also applies. In particular, $f(x) = \lambda x + b$ with $\lambda \ge 0$ recovers Corollary 7.4 for the quadratic controller, including its constant independent of λ.

Proof. The joint subgradient $(f'(x_t), \zeta_t^u)$, with $\zeta_t^u \in \partial \ell_t(u_t)$ and $|\zeta_t^u| \le L_u$, satisfies (7.1). The basic-controller claim follows from Theorem 7.3.

For the forecast $\widehat{c}_{s,I,k}(x, u) = f(x)$, (8.1) gives

$$\widehat{x}_a(p) = B^{\mathrm{bg}}_{s,I} + w_s, \qquad \widehat{x}_k(p) = B^{\mathrm{bg}}_{s,I} + w_s(1-p) \quad (a < k \le r_I).$$

Consequently,

$$\begin{aligned} F_{s,I}(p) = {} & \mathbf{1}_{\{s+a \le T\}} f(B^{\mathrm{bg}}_{s,I} + w_s) \\ & + \gamma_{s,I} f(B^{\mathrm{bg}}_{s,I} + w_s(1-p)). \end{aligned}$$

The first term is independent of p, and substitution in (5.6) gives (8.8).

Here $\widehat{g}^u_{s,I} = 0$, so the component bounds in the proof of Proposition 8.1 give

$$\begin{aligned} |e_{s,I}| &\le |g^u_{s,I}| + |g^x_{s,I} - \widehat{g}^x_{s,I}| \\ &\le w_s[2L_u + (\lambda + V)(d_I - 1)]. \end{aligned}$$

Theorem 6.4 and Lemma 7.2 then give (7.8). For $f(x) = \lambda x + b$, take $V = 0$ and apply the quadratic specialization in Corollary 7.4. □

If a known convex f is not nondecreasing, (8.7) still falls under Theorem 6.5 whenever $|f'| \le L_x$, and the same predictor is covered by Proposition 8.1 with $K = H$. The refinement that can shorten the deadline then need not apply.

9 Implementation and Deadline Dependence

At a node of length d_I, store its d_I reference values and the outstanding splits and child values needed to complete feedback. Each node starts at most one new decision for an arrival per round, and its feedback is complete within d_I rounds. It therefore needs at most d_I unfinished feedback records. Each record stores a constant number of split, routing, and child-value entries, so the node uses $O(d_I)$ memory in total. Resource reserved for a later interval remains stored until that interval begins. Since intervals at a fixed depth are disjoint and have total length at most K,

$$\sum_{I \in \mathcal{I}_K} d_I = O(K \log(2K)). \tag{9.1}$$

Summing over the tree gives $O(K \log(2K))$ memory. Completed records can be discarded, so this requirement does not grow with T.

Compute the leaf values using a scalar prefix sum $P_t = \sum_{\tau=1}^{t} \zeta_\tau^x$ and its stored values at active arrival times:

$$A_{s,i} = w_s(\zeta_{s+i}^u + P_{s+i} - P_s).$$

Each node has at most one new split decision for an arrival and one completed feedback value per physical round. A direct implementation uses $O(K \log(2K))$ arithmetic per round. This bound excludes evaluating or solving arbitrary forecast problems; (8.3) may require several predicted cost evaluations.

Both regularizers use the same state, routing, feedback timing, and tree identity. Quadratic updates are additive in a fraction; entropic updates are additive in its log odds. They generally produce different trajectories. Nonlinear prediction is available with either regularizer.

The deadline affects this controller's regret bound in two ways. First, the root must wait K rounds for complete feedback and therefore uses K separate learner copies. Combining their regret bounds introduces a factor of $\sqrt{K}$. Second, changing the consumption age can change as many as $K-1$ storage-cost terms, so the feedback magnitude grows as $O(L_u + KL_x)$. The shorter intervals farther down the tree add only a constant factor. Together, these effects give the bound

$$O\big((L_u + KL_x)\sqrt{KT}\big) = O\big(L_u\sqrt{KT} + L_x K^{3/2}\sqrt{T}\big).$$

With $K = H$, these are the service and storage contributions in (6.18). The storage-cost lower bound in the next section matches the state contribution.

10 Lower Bounds and Minimax Implications

General optimization with memory already exhibits an $\Omega(Lh^{3/2}\sqrt{T})$ lower bound for a suitable memory-h loss class [31, Appendix D]. The construction here establishes the storage-dependent rate within the admissible scalar system, against its fixed age policies and for all causal deadline-feasible controllers; it is not an automatic consequence of that broader class.

All lower bounds below use oblivious instances: arrivals and cost functions are fixed before the first decision and are independent of the controller's internal randomization. Random signs

are used only to construct a distribution over such fixed instances. The expectation in the final statements for fixed input sequences is over the controller alone.

Appendix A.3 supplies the independent-sign moment bound and the averaging argument used below. In particular, $\mathbb{E}[|S_n|] \geq \sqrt{n/3}$ gives the constants used here.

10.1 The action-marginal lower bound

Theorem 10.1 (Consumption-only lower bound)**.** *For $T \geq 2$, $H \geq 2$, and $L_u > 0$, every causal deadline-feasible controller has a deterministic oblivious instance with consumption-only jointly affine costs such that*

$$\mathbb{E}[\mathrm{Reg}_T(\Delta_H)] \geq \frac{L_u}{2\sqrt{3}}\sqrt{T-1}. \tag{10.1}$$

Proof. Set $w_s = 1$ for odd s and $w_s = 0$ for even s. Draw independent signs $\xi_2, \ldots, \xi_T$ and set $c_1(x,u) = 0$, $c_t(x,u) = L_u \xi_t u$ for $t \geq 2$. Let $\mathcal{F}_{t-1}^{\mathcal{A}}$ contain the controller's random seed and all observations available before its round-t action. Then u_t is $\mathcal{F}_{t-1}^{\mathcal{A}}$-measurable and the fresh obliviously sampled sign satisfies $\mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}^{\mathcal{A}}] = 0$. Hence

$$\mathbb{E}[\xi_t u_t \mid \mathcal{F}_{t-1}^{\mathcal{A}}] = u_t \mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}^{\mathcal{A}}] = 0, \qquad \mathbb{E}[J_T^{\mathrm{on}}] = L_u \sum_{t=2}^{T} \mathbb{E}[\xi_t u_t] = 0.$$

For the policies $\mathbf{e}_1, \mathbf{e}_2$, put

$$C_1 = L_u \sum_{t=2}^{T} \xi_t w_{t-1}, \qquad C_2 = L_u \sum_{t=2}^{T} \xi_t w_{t-2}.$$

Then $\mathbb{E}[C_1] = \mathbb{E}[C_2] = 0$ and

$$w_{t-1} - w_{t-2} = (-1)^t \in \{-1, 1\}, \qquad C_1 - C_2 = L_u \sum_{t=2}^{T} (-1)^t \xi_t \overset{\mathrm{d}}{=} L_u S_{T-1}.$$

Using $\min\{a,b\} = (a + b - |a-b|)/2$,

$$\mathbb{E}[\min\{C_1, C_2\}] = -\frac{1}{2}\mathbb{E}[|C_1 - C_2|] = -\frac{L_u}{2}\mathbb{E}[|S_{T-1}|].$$

Since $\min_{\mathbf{m} \in \Delta_H} J_T(\mathbf{m}) \leq \min\{C_1, C_2\}$,

$$\begin{aligned} \mathbb{E}[\mathrm{Reg}_T(\Delta_H)] &\geq \mathbb{E}[J_T^{\mathrm{on}}] - \mathbb{E}[\min\{C_1, C_2\}] \\ &= \frac{L_u}{2}\mathbb{E}[|S_{T-1}|] \\ &\geq \frac{L_u}{2\sqrt{3}}\sqrt{T-1}, \end{aligned}$$

by Lemma A.2. Averaging over the finite sign family fixes a deterministic sign sequence with at least this expected regret for the given randomized controller. □

Table 1: Local rounds in the deadline-allocation lower-bound construction.

Local rounds r	Arrival w_t	Coefficient a_t in $c_t(x,u) = a_t x$
$1, \dots, b$	1	0
$b+1$	0	0
$b+2, \dots, 2b+1$	0	$L_x/2$
$2b+2, \dots, 3b+1$	0	$-L_x(1+\varepsilon_j)/2$
$3b+2, \dots, 4b$	0	0

10.2 A matching state-marginal lower bound

Theorem 10.2 (Lower bound for signed linear storage costs)**.** *For $H \geq 3$, $T \geq 4$, and $L_x > 0$, every causal deadline-feasible controller has a deterministic oblivious instance with costs $c_t(x,u) = a_t x$, $|a_t| \leq L_x$, such that*

$$\boxed{\mathbb{E}[\mathrm{Reg}_T(\Delta_H)] \geq \frac{L_x}{256\sqrt{3}}\, H^{3/2}\sqrt{T}.} \tag{10.2}$$

The state-marginal term in Theorem 6.5 is therefore sharp up to universal constants in this range.

Proof. Let

$$b = \min\{\lfloor H/3 \rfloor, \lfloor T/4 \rfloor\}, \qquad n = \lfloor T/(4b) \rfloor. \tag{10.3}$$

We have $b, n \geq 1$ and $4bn \leq T$. Use n consecutive blocks of $4b$ rounds, with zero costs and arrivals in any remaining rounds. In block j, draw a fresh independent symmetric sign ε_j. At local time r within the block, specify the arrivals and coefficients in Table 1; empty rows are omitted. The coefficient in the random phase is either 0 throughout or $-L_x$ throughout. All signs can be sampled before the interaction.

Let Y_j be the controller's state at the beginning of the first random-phase round. It is determined before ε_j is first revealed. There are no arrivals throughout the positive-cost phase or the random phase, so the state is nonincreasing there. Every positive-cost state is at least Y_j, and every random-phase state is at most Y_j. This remains true if resource from earlier blocks is still present. Therefore the block's online cost satisfies the pathwise bound

$$\begin{aligned} C_j^{\mathrm{on}} &\geq \frac{L_x}{2} bY_j - \frac{L_x}{2}(1+\varepsilon_j) bY_j \\ &= -\frac{L_x}{2} b\varepsilon_j Y_j. \end{aligned} \tag{10.4}$$

Let $\mathcal{F}_j$ be the history, including the controller's random seed, immediately before block j's random phase. Then Y_j is $\mathcal{F}_j$-measurable and $\mathbb{E}[\varepsilon_j \mid \mathcal{F}_j] = 0$. Thus

$$\mathbb{E}[C_j^{\mathrm{on}} \mid \mathcal{F}_j] \geq -\frac{L_x}{2} bY_j \mathbb{E}[\varepsilon_j \mid \mathcal{F}_j] = 0, \qquad \mathbb{E}[J_T^{\mathrm{on}}] = \sum_{j=1}^{n} \mathbb{E}[C_j^{\mathrm{on}}] \geq 0.$$

In particular, learning the sign during that phase cannot invalidate the pathwise lower bound.

Consider the fixed comparators $\mathbf{e}_1$ and $\mathbf{e}_{3b}$. Both belong to Δ_H. The immediate policy has consumed every new arrival by local round $b+1$, and has zero block cost. The delayed policy holds all b new units at the beginning of every positive and random cost round: its first service occurs

at $3b+1$, after that round's state cost. Its last service occurs at $4b$, so it carries nothing into the next block. Its block cost is exactly

$$\begin{aligned} C_j(\mathbf{e}_{3b}) &= \frac{L_x}{2}\, b \cdot b - \frac{L_x}{2}(1+\varepsilon_j) b \cdot b \\ &= -\frac{L_x}{2} b^2 \varepsilon_j. \end{aligned} \tag{10.5}$$

Consequently, with $S_n = \sum_j \varepsilon_j$,

$$\begin{aligned} \mathbb{E}[\mathrm{Reg}_T(\Delta_H)] &\geq \mathbb{E}[J_T^{\mathrm{on}}] - \mathbb{E}\left[\min\left\{0, -\frac{L_x b^2}{2} S_n\right\}\right] \\ &\geq -\mathbb{E}\left[\min\left\{0, -\frac{L_x b^2}{2} S_n\right\}\right] \\ &= \frac{L_x b^2}{4} \mathbb{E}[|S_n|] \geq \frac{L_x b^2}{4\sqrt{3}} \sqrt{n}. \end{aligned} \tag{10.6}$$

Since $H/3 \geq 1$ and $T/4 \geq 1$, $\lfloor H/3 \rfloor \geq H/8$ and $\lfloor T/4 \rfloor \geq T/8 \geq H/8$, where the last inequality uses $H \leq T$. Thus

$$b \geq \frac{H}{8}.$$

Also, $\lfloor q \rfloor \geq q/2$ for $q \geq 1$ gives

$$n = \left\lfloor \frac{T}{4b} \right\rfloor \geq \frac{T}{8b}.$$

Thus

$$\begin{aligned} \frac{L_x b^2}{4\sqrt{3}} \sqrt{n} &\geq \frac{L_x}{8\sqrt{6}} b^{3/2} \sqrt{T} \\ &\geq \frac{L_x}{256\sqrt{3}} H^{3/2} \sqrt{T}. \end{aligned}$$

Averaging over the finite sign family fixes a deterministic oblivious instance with the same expected-regret lower bound for the given controller. □

Remark 10.3 (Small deadlines and the sign of the marginal). The restriction $H \geq 3$ is only for the block construction. For $H = 2$, $T \geq 4$, let $w_t = 1$ and use independent coefficients $a_t = L_x \xi_t$ for $t \geq 3$, with earlier costs zero. Every controller again has zero expected cost, while the states of $\mathbf{e}_1, \mathbf{e}_2$ differ by one at each such time. Their comparison gives $L_x\sqrt{T-2}/(2\sqrt{3})$, which is the same order as $L_x H^{3/2}\sqrt{T}$ for $H = 2$. For $T = 2$, all states within the cost horizon are predetermined, so no positive state-only regret lower bound is asserted. For $H = 1$ regret is zero. The matching state lower bound allows increasing storage to either raise or lower the current cost. The construction uses negative storage slopes during some rounds. Its costs can nevertheless be made nonnegative by adding constants that do not depend on the controller's decisions. These constants leave regret and the storage slopes unchanged.

10.3 Why nonnegative state marginal costs still need a magnitude bound

Nonnegative state marginals can still produce unpredictable storage charges. The next construction shows that, when service also has a cost, no regret bound that is uniformly sublinear in T over this class can be independent of the largest allowed state marginal.

Proposition 10.4 (Unpredictable nonnegative spikes)**.** *For every integer $n \geq 2$ and $L_u > 0$, there is a distribution of oblivious instances with $H = 2$, $T = 3n$, nonnegative jointly affine costs, and $0 \leq \zeta^x \leq nL_u$, $0 \leq \zeta^u \leq L_u$, such that every causal controller has expected regret at least*

$$nL_u(1 - 1/n)^n \geq nL_u/4. \tag{10.7}$$

A deterministic instance with at least this expected regret therefore exists for every randomized controller.

Proof. Use n independent three-round cycles. One unit arrives in the first round; there are no other arrivals in the cycle. Set the first cost to zero, the second to $c_t(x, u) = L_u u$, and the third to $c_t(x, u) = A_j x$, where $A_j = nL_u$ with probability $1/n$ and $A_j = 0$ otherwise. These coefficients are sampled in advance and only revealed at their rounds. The deadline forces all cycle resource to be consumed in its second and third rounds, and all earlier resource is already gone. If the controller serves fraction p_j in the second round, its cycle cost is $C_j^{\mathrm{on}} = L_u p_j + A_j(1 - p_j)$. Since p_j is measurable before A_j is revealed and $\mathbb{E}[A_j] = (1/n)nL_u = L_u$,

$$\begin{aligned} \mathbb{E}[C_j^{\mathrm{on}} \mid p_j] &= L_u p_j + \mathbb{E}[A_j \mid p_j](1 - p_j) = L_u, \\ \mathbb{E}[J_T^{\mathrm{on}}] &= \sum_{j=1}^{n} \mathbb{E}[C_j^{\mathrm{on}}] = nL_u. \end{aligned}$$

For a fixed comparator $(p, 1 - p)^\top \in \Delta_2$,

$$J_T(p) = pnL_u + (1 - p) \sum_{j=1}^{n} A_j.$$

Thus

$$\min_{p \in [0,1]} J_T(p) = \min \left\{ nL_u, \sum_{j=1}^{n} A_j \right\} = nL_u \mathbf{1}_{\{\max_j A_j > 0\}}.$$

Therefore

$$\begin{aligned} \mathbb{E}[\mathrm{Reg}_T(\Delta_2)] &= nL_u \mathbb{P}(\max_j A_j = 0) \\ &= nL_u \left(1 - \frac{1}{n}\right)^n \geq \frac{nL_u}{4}, \end{aligned}$$

where $(1-1/n)^n \geq (1-1/2)^2 = 1/4$ for $n \geq 2$. Averaging over the finite family of spike realizations fixes a deterministic instance with at least this expected regret for the given randomized controller. □

In this construction, $H = 2$ and $T = 3n$. With $L_u > 0$ fixed, the allowed state marginals grow as large as $nL_u = (T/3)L_u$, while expected regret is at least $L_u T/12$. Thus a bound depending only on T, H, and L_u cannot be sublinear in T uniformly over all allowed magnitudes of the nonnegative state marginals.

10.4 Minimax bounds and unresolved deadline dependence

For each controller, consider its largest expected regret over admissible arrival and cost sequences fixed before the run begins. The expectation is over the controller's internal randomness. Taking

the infimum of these worst-case values over controllers that use only revealed information and meet every arrival's deadline defines $\mathfrak{R}_{T,H}(L_x, L_u)$. The admissible costs satisfy the joint-cost assumptions with bounds L_x and L_u. For $H \geq 3$, $T \geq 4$, the established results imply

$$\begin{aligned} \max\left\{\frac{L_u}{2\sqrt{3}}\sqrt{T-1},\ \frac{L_x}{256\sqrt{3}}H^{3/2}\sqrt{T}\right\} &\leq \mathfrak{R}_{T,H}(L_x, L_u), \\ \mathfrak{R}_{T,H}(L_x, L_u) &\leq \frac{2L_u + (H-1)L_x}{2}\sqrt{T-1}\,\Lambda_H. \end{aligned} \tag{10.8}$$

The two subclasses give separate lower bounds, so their maximum bounds the minimax regret from below. In particular,

$$\mathfrak{R}_{T,H}(L_x, 0) = \Theta(L_x H^{3/2}\sqrt{T}) \quad (H \geq 3,\ T \geq 4). \tag{10.9}$$

For fixed $L_x > 0$ and bounded ratio L_u/L_x, the same dependence on H, T follows for the general class, with constants allowed to depend on that ratio. These bounds do not establish matching dependence on L_x and L_u when they vary independently.

For consumption-only costs, the separate conclusion is

$$\frac{L_u}{2\sqrt{3}}\sqrt{T-1} \leq \mathfrak{R}_{T,H}(0, L_u) \leq 2(1+\sqrt{2})L_u\sqrt{H(T-1)} \qquad (H \geq 2,\ T \geq 2). \tag{10.10}$$

When costs depend only on consumption and the deadline is fixed, the bounds establish the optimal $\sqrt{T}$ growth. Their dependence on the deadline remains unresolved: the upper bound increases with $\sqrt{H}$, while the lower bound has no deadline factor. The state lower bound uses costs that depend on storage, so it cannot answer this question for consumption-only costs. The sharp regret rate under the additional nonnegative storage-marginal assumptions in (7.1) also remains open.

11 Future Research Directions

The main open question is the action contribution in (10.10). The current controller waits for an entire interval before updating one interleaved copy, although a single observed pair (ζ_t^x, ζ_t^u) informs several active arrivals and intervals.

Remark 11.1 (Progressively revealed split feedback). For a fixed (s, I) with $I = [a, b]$, let $\pi_i^{\mathrm{L}}, \pi_i^{\mathrm{R}}$ be the final conditional leaf fractions in its two children, extended by zero outside the corresponding child, and define

$$\delta_i = \pi_i^{\mathrm{L}} - \pi_i^{\mathrm{R}}, \qquad D_k = \sum_{i=k}^{b}\delta_i = -\sum_{i=a}^{k-1}\delta_i. \tag{11.1}$$

The two expressions for D_k agree because both conditional distributions have unit mass and hence $\sum_i \delta_i = 0$. Expanding (5.4) gives the exact identity

$$\boxed{g_{s,I} = w_s\left[\sum_{i=a}^{b}\delta_i\zeta_{s+i}^u + \sum_{k=a+1}^{b}D_k\zeta_{s+k}^x\right].} \tag{11.2}$$

Each conditional leaf fraction is fixed before its service, and D_k can be computed from earlier conditional services through (11.1). Hence the summands are revealed progressively even though the full $g_{s,I}$ is unavailable at the split.

Using partial feedback requires a new regret analysis. Each revealed piece of feedback belongs to a split that has already been chosen. An early update changes the learner's reference value while some feedback for that split is still missing. The analysis must therefore account for the full cost comparison associated with the original split and count each feedback contribution exactly once. Such an analysis might reduce the cost of maintaining separate learners for delayed feedback.

Another direction is to prove a stronger lower bound for consumption-only costs. A lower bound that grows with the deadline could show whether the current $\sqrt{H}$ factor is necessary.

12 Conclusion

This paper studies online allocation under resource constraints and individual service deadlines. The controller divides each arrival among age intervals as they begin, ensuring complete consumption by the deadline while allowing later decisions to use newly observed costs. Delayed mirror-descent updates provide regret guarantees against the best fixed age-allocation policy under adversarial jointly convex costs. Uniformly positive state marginal costs can justify a shorter service deadline, and known state costs can be incorporated directly into causal predictions.

A matching lower bound for signed linear storage costs shows that the storage-dependent term in the general regret bound is unavoidable. The optimal deadline dependence for consumption-only costs remains open. Exploiting the progressively revealed feedback shared by active arrivals and intervals is a direction for closing this gap.

A Supporting Lemmas

A.1 Scalar quadratic and entropy geometry

For a differentiable convex function ψ, its Bregman divergence is

$$D_\psi(p, z) = \psi(p) - \psi(z) - \psi'(z)(p - z).$$

The quadratic regularizer $\psi_Q(p) = p^2/2$ gives $D_{\psi_Q}(p, z) = (p - z)^2/2$ and $\sup_{p\in[0,1]} D_{\psi_Q}(p, 1/2) = 1/8$. For binary entropy, let

$$\psi_E(p) = p \log p + (1 - p) \log(1 - p), \qquad p \in [0, 1],$$

with the continuous convention at the endpoints. Its divergence is $\mathrm{kl}(p\|z)$ for $z \in (0, 1)$. Since

$$\psi''_E(p) = \frac{1}{p} + \frac{1}{1 - p} = \frac{1}{p(1 - p)} \geq 4 \qquad (0 < p < 1),$$

integration and continuity at the endpoints give

$$D_{\psi_E}(p, z) \geq 2(p - z)^2, \qquad D_{\psi_E}(p, 1/2) = \log 2 + \psi_E(p) \leq \log 2.$$

These are the binary specializations of standard entropy geometry bounds [11].

For the reference update, quadratic minimization gives $z^+ = [z - \eta g]_0^1$. Under entropy, first-order optimality gives

$$\eta g + \log \frac{z^+}{1 - z^+} - \log \frac{z}{1 - z} = 0,$$

hence

$$z^+ = \frac{z e^{-\eta g}}{1 - z + z e^{-\eta g}}.$$

Strict convexity gives uniqueness. For finite g and $z \in (0,1)$ this update is interior. The prediction objective with a finite continuous convex differentiable predictor also has an interior entropy minimizer: the entropy derivative tends to $-\infty$ and $+\infty$ at the two endpoints, respectively. Quadratic minimizers may lie on the boundary.

A.2 Prediction and correction from a common reference

The following scalar inequality is the prediction–correction comparison used in the analysis; compare [42, Lemma 3].

Lemma A.1 (Prediction and correction from a common reference)**.** *Let ψ be either regularizer in Appendix A.1, and let $\sigma > 0$ satisfy $D_\psi(v,w) \geq \sigma(v-w)^2/2$ on its domain. Let $\eta > 0$, let $F : [0,1] \to \mathbb{R}$ be convex and differentiable with finite derivative, and let $g \in \mathbb{R}$. Define*

$$p = \operatorname*{arg\,min}_{v\in[0,1]}\{\eta F(v) + D_\psi(v,z)\}, \qquad z^+ = \operatorname*{arg\,min}_{v\in[0,1]}\{\eta g v + D_\psi(v,z)\}.$$

For entropy assume $z \in (0,1)$; for the quadratic case allow $z \in [0,1]$. Writing $\widehat{g} = F'(p)$, every $u \in [0,1]$ satisfies

$$g(p-u) \leq \frac{D_\psi(u,z) - D_\psi(u,z^+)}{\eta} + \frac{\eta}{2\sigma}(g-\widehat{g})^2 - \frac{D_\psi(p,z)}{\eta}.$$

Proof. Direct expansion yields the three-point identity

$$(\psi'(a) - \psi'(b))(c-a) = D_\psi(c,b) - D_\psi(c,a) - D_\psi(a,b).$$

The correction optimality condition, tested at u, therefore gives

$$\eta g(z^+ - u) \leq D_\psi(u,z) - D_\psi(u,z^+) - D_\psi(z^+,z).$$

The prediction optimality condition, tested at z^+, similarly gives

$$\eta\widehat{g}(p - z^+) \leq D_\psi(z^+,z) - D_\psi(z^+,p) - D_\psi(p,z).$$

Both conditions allow quadratic boundary minimizers; the entropy minimizers are interior by Appendix A.1. Adding the inequalities and the remaining term $\eta(g-\widehat{g})(p-z^+)$ yields

$$\begin{aligned}\eta g(p-u) &\leq D_\psi(u,z) - D_\psi(u,z^+) - D_\psi(p,z) \\ &\quad - D_\psi(z^+,p) + \eta(g-\widehat{g})(p-z^+) \\ &\leq D_\psi(u,z) - D_\psi(u,z^+) - D_\psi(p,z) + \frac{\eta^2}{2\sigma}(g-\widehat{g})^2.\end{aligned}$$

The last step uses strong convexity and $\eta ab \leq \sigma b^2/2 + \eta^2 a^2/(2\sigma)$. Division by η proves the claim. □

A.3 Independent signs and deterministic instances

Lemma A.2 (Weighted independent-sign bound; cf. [24])**.** *For independent random signs $\xi_1, \ldots, \xi_n$, each equally likely to be -1 or $+1$, and deterministic real coefficients a_i, let $S = \sum_i a_i \xi_i$. Then*

$$\mathbb{E}[|S|] \geq \frac{1}{\sqrt{3}} \left(\sum_i a_i^2 \right)^{1/2}. \tag{A.1}$$

In particular, when all $a_i = 1$,

$$\mathbb{E}[|S|] \geq \sqrt{n/3}. \tag{A.2}$$

Proof. Put $v = \sum_i a_i^2$. Independence and the fact that products containing an odd power of any sign have zero expectation give

$$\begin{aligned}\mathbb{E}[S^2] &= \sum_i a_i^2 = v,\\ \mathbb{E}[S^4] &= \sum_i a_i^4 + 6\sum_{i<j} a_i^2 a_j^2\\ &= 3\left(\sum_i a_i^2\right)^2 - 2\sum_i a_i^4 \le 3v^2.\end{aligned}$$

Hölder's inequality with exponents $3/2$ and 3 gives

$$\begin{aligned}v = \mathbb{E}[|S|^2] &= \mathbb{E}[|S|^{2/3}|S|^{4/3}]\\ &\le (\mathbb{E}[|S|])^{2/3}(\mathbb{E}[|S|^4])^{1/3} \le (\mathbb{E}[|S|])^{2/3}(3v^2)^{1/3}.\end{aligned}$$

For $v > 0$, cubing and cancelling v^2 yields $\mathbb{E}[|S|] \ge \sqrt{v/3}$; for $v = 0$, both sides vanish. □

The classical Khintchine inequality gives the sharper bound [24]:

$$\mathbb{E}\left[\left|\sum_i a_i \xi_i\right|\right] \ge \frac{1}{\sqrt{2}}\left(\sum_i a_i^2\right)^{1/2}. \tag{A.3}$$

To obtain a deterministic instance, first fix a controller, including the rule for its random choices. Then draw an input sequence from a finite collection, independently of those choices; each sequence specifies all inputs before the run begins. If regret averaged over both the sequence draw and the controller's random choices is at least R, then some sequence in the collection has expected regret at least R when only the controller is randomized. This sequence can depend on which controller was fixed.

References

[1] Mohammed Abdullah, George Iosifidis, Salah Eddine Elayoubi, and Tijani Chahed. Constrained online convex optimization with memory and predictions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(24):19524–19532, 2026.

[2] Naman Agarwal, Brian Bullins, Elad Hazan, Sham Kakade, and Karan Singh. Online control with adversarial disturbances. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 111–119. PMLR, 2019.

[3] Oren Anava, Elad Hazan, and Shie Mannor. Online learning for adversaries with memory: Price of past mistakes. In *Advances in Neural Information Processing Systems*, volume 28, pages 784–792. Curran Associates, Inc., 2015.

[4] Antonios Antoniadis, Peyman Jabbarzade, and Golnoosh Shahkarami. A novel prediction setup for online speed-scaling. In *18th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2022)*, volume 227 of *Leibniz International Proceedings in Informatics*, pages 9:1–9:20. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2022.

[5] Raman Arora, Ofer Dekel, and Ambuj Tewari. Online bandit learning against an adaptive adversary: From regret to policy regret. In *Proceedings of the 29th International Conference on Machine Learning*, pages 1503–1510. Omnipress, 2012.

[6] Kamiar Asgari and Michael J. Neely. Bregman-style online convex optimization with energy harvesting constraints. *Proc. ACM Meas. Anal. Comput. Syst.*, 4(3), 11 2020.

[7] Kamiar Asgari and Michael J. Neely. Online constrained control of storage systems via simplex disturbance-action policies, 2026. Preprint, arXiv:2609.07690.

[8] Santiago R. Balseiro, Haihao Lu, and Vahab Mirrokni. The best of many worlds: Dual mirror descent for online allocation problems. *Operations Research*, 71(1):101–119, 2023.

[9] Étienne Bamas, Andreas Maggiori, Lars Rohwedder, and Ola Svensson. Learning augmented energy minimization via speed scaling. In *Advances in Neural Information Processing Systems*, volume 33, pages 15350–15359. Curran Associates, Inc., 2020.

[10] Nikhil Bansal, Tracy Kimbrel, and Kirk Pruhs. Speed scaling to manage energy and temperature. *Journal of the ACM*, 54(1), 2007.

[11] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[12] Semih Cayci and Atilla Eryilmaz. Optimal learning for dynamic coding in deadline-constrained multi-channel networks. *IEEE/ACM Transactions on Networking*, 27(3):1043–1054, 2019.

[13] Chi-Kin Chau, Guanglin Zhang, and Minghua Chen. Cost minimizing online algorithms for energy storage management with worst-case guarantee. *IEEE Transactions on Smart Grid*, 7(6):2691–2702, 2016.

[14] Ching-An Cheng, Xinyan Yan, Nathan Ratliff, and Byron Boots. Predictor-corrector policy optimization. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1151–1161. PMLR, 2019.

[15] Zhiyuan Fan, Christian Kroer, and Gabriele Farina. On the optimality of dilated entropy and lower bounds for online learning in extensive-form games. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[16] Gabriele Farina, Christian Kroer, Noam Brown, and Tuomas Sandholm. Stable-predictive optimistic counterfactual regret minimization. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1853–1862. PMLR, 2019.

[17] Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Online convex optimization for sequential decision processes and extensive-form games. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(1):1917–1925, 2019.

[18] Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Optimistic regret minimization for extensive-form games via dilated distance-generating functions. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[19] Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Better regularization for sequential decision spaces: Fast convergence rates for Nash, correlated, and team equilibria. *Operations Research*, 73(5):2430–2457, 2025.

[20] Genevieve E. Flaspohler, Francesco Orabona, Judah Cohen, Soukayna Mouatadid, Miruna Oprescu, Paulo Orenstein, and Lester Mackey. Online learning with optimism and delay. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 3363–3373. PMLR, 2021.

[21] Alvin Fu, Eytan Modiano, and John N. Tsitsiklis. Optimal transmission scheduling over a fading channel with energy and deadline constraints. *IEEE Transactions on Wireless Communications*, 5(3):630–641, 2006.

[22] Lingwen Gan, Adam Wierman, Ufuk Topcu, Niangjun Chen, and Steven H. Low. Real-time deferrable load control: Handling the uncertainties of renewable generation. In *Proceedings of the Fourth International Conference on Future Energy Systems*, pages 113–124. ACM, 2013.

[23] Noah Golowich, Elad Hazan, Zhou Lu, Dhruv Rohatgi, and Y. Jennifer Sun. Online control in population dynamics. In *Advances in Neural Information Processing Systems*, volume 37. Curran Associates, Inc., 2024.

[24] Uffe Haagerup. The best constants in the Khintchine inequality. *Studia Mathematica*, 70(3):231–283, 1981.

[25] Massil Hihat, Stéphane Gaïffas, Guillaume Garrigos, and Simon Bussy. Online inventory problems: Beyond the i.i.d. setting with online convex optimization. In *Advances in Neural Information Processing Systems*, volume 36, pages 20421–20440, 2023.

[26] Samid Hoda, Andrew Gilpin, Javier Peña, and Tuomas Sandholm. Smoothing techniques for computing Nash equilibria of sequential games. *Mathematics of Operations Research*, 35(2):494–512, 2010.

[27] Pooria Joulani, András György, and Csaba Szepesvári. Online learning under delayed feedback. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28(3) of *Proceedings of Machine Learning Research*, pages 1453–1461. PMLR, 2013.

[28] Pooria Joulani, András György, and Csaba Szepesvári. Delay-tolerant online convex optimization: Unified analysis and adaptive-gradient algorithms. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), 2016.

[29] Mohammad Ali Khojastepour and Ashutosh Sabharwal. Delay-constrained scheduling: Power efficiency, filter design, and bounds. In *IEEE INFOCOM 2004*, volume 3, pages 1938–1949. IEEE, 2004.

[30] Linggao Kong, Yuedong Xu, Lei Jiao, and Chuan Xu. Deadline-aware online scheduling for LLM fine-tuning with spot market predictions, 2025. Preprint, arXiv:2512.20967.

[31] Raunak Kumar, Sarah Dean, and Robert D. Kleinberg. Online convex optimization with unbounded memory. In *Advances in Neural Information Processing Systems*, volume 36, pages 26229–26270. Curran Associates, Inc., 2023.

[32] Adam Lechowicz, Nicolas Christianson, Mohammad Hajiesmaili, Adam Wierman, and Prashant Shenoy. Online smoothed demand management, 2026. Full version, arXiv:2511.18554v2; accepted to ACM SIGMETRICS 2026.

[33] Juyul Lee and Nihar Jindal. Energy-efficient scheduling of delay constrained traffic over fading channels. *IEEE Transactions on Wireless Communications*, 8(4):1866–1875, 2009.

[34] Yingying Li, Subhro Das, and Na Li. Online optimal control with affine constraints. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):8527–8537, 2021.

[35] Qingsong Liu and Zhixuan Fang. Learning to schedule tasks with deadline and throughput constraints. In *IEEE INFOCOM 2023—IEEE Conference on Computer Communications*, pages 1–10. IEEE, 2023.

[36] Xin Liu, Zixian Yang, and Lei Ying. Online nonstochastic control with adversarial and static constraints. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 22277–22288. PMLR, 2023.

[37] Mehrdad Mahdavi, Rong Jin, and Tianbao Yang. Trading regret for efficiency: Online convex optimization with long term constraints. *Journal of Machine Learning Research*, 13(81):2503–2528, 2012.

[38] Zoubeir Mlika, Elmahdi Driouch, and Wessam Ajib. Deadline scheduling in energy harvesting networks: Competitive and learning algorithms. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2019.

[39] Michael J. Neely, Arash Saber Tehrani, and Alexandros G. Dimakis. Efficient algorithms for renewable energy allocation to delay tolerant consumers. In *2010 First IEEE International Conference on Smart Grid Communications*, pages 549–554. IEEE, 2010.

[40] Anthony Pineci and Yunzong Xu. Optimal hidden-target learning for online inventory optimization on general convex sets, 2026. Preprint, arXiv:2606.14679.

[41] Kent Quanrud and Daniel Khashabi. Online learning with adversarial delays. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[42] Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In *Proceedings of the 26th Annual Conference on Learning Theory*, volume 30 of *Proceedings of Machine Learning Research*, pages 993–1019. PMLR, 2013.

[43] Nitin Salodkar, Abhijeet Bhorkar, Abhay Karandikar, and Vivek S. Borkar. An on-line learning algorithm for energy efficient delay constrained scheduling over a fading channel. *IEEE Journal on Selected Areas in Communications*, 26(4):732–742, 2008.

[44] Elif Uysal-Biyikoglu, Balaji Prabhakar, and Abbas El Gamal. Energy-efficient packet transmission over a wireless link. *IEEE/ACM Transactions on Networking*, 10(4):487–499, 2002.

[45] Cheng-Zhong Xu, Minghua Xu, Le Yi Wang, and George Yin. Filter design and analysis in frequency domain for server scheduling and optimization. *IEEE Transactions on Parallel and Distributed Systems*, 21(11):1573–1585, 2010.

[46] Frances Yao, Alan Demers, and Scott Shenker. A scheduling model for reduced CPU energy. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 374–382. IEEE, 1995.

[47] Huanan Zhang, Xiuli Chao, and Cong Shi. Technical note—perishable inventory systems: Convexity results for base-stock policies and learning algorithms under censored demand. *Operations Research*, 66(5):1276–1286, 2018.

[48] Wentao Zhang and Wentao Mo. Real-time hard peak age-of-information safety with no-regret learning, 2026. Preprint, arXiv:2607.27626.

[49] Peng Zhao, Yu-Hu Yan, Yu-Xiang Wang, and Zhi-Hua Zhou. Non-stationary online learning with memory and non-stochastic control. *Journal of Machine Learning Research*, 24(206):1–70, 2023.

[50] Hongyu Zhou and Vasileios Tzoumas. Safe non-stochastic control of linear dynamical systems, 2023. arXiv:2308.12395. Extended version of the paper presented at the 2023 IEEE Conference on Decision and Control.

[51] Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. In *Advances in Neural Information Processing Systems*, volume 20, 2007.