Markov Metrical Task Systems

Christian Coester*
Department of Computer Science
University of Oxford
United Kingdom
christian.coester@cs.ox.ac.uk

Alexa Tudose*
Department of Computer Science
University of Oxford
United Kingdom
maria-alexa.tudose@cs.ox.ac.uk

Abstract

We study Metrical Task Systems, a broad class of problems in online algorithms, under a model where input requests are generated by a stochastic process. We first consider the case of requests drawn IID from an unknown distribution, and then extend the model to requests generated by a finite-state Markov chain, where the algorithm observes both the current request and the current state of the chain before making its next move. This formulation captures practical settings in which requests are generated by an evolving environment. In both settings, we show how to find in polynomial time an algorithm whose expected cost per step is arbitrarily close to the cost of the best possible online algorithm. Our theoretical results are complemented by a brief empirical evaluation of our algorithms.

1 Introduction

Metrical Task Systems (MTS), introduced by Borodin et al. [BLS92], is a central problem in online decision-making. In MTS, an algorithm controls a server moving in a metric space $\mathcal{M} = (M, d)$, starting from an initial state $s_0 \in M$. At each time step $t \geq 1$, a request (task) is revealed in the form of a cost function $c_t \colon M \to \mathbb{R}_{\geq 0} \cup \{\infty\}$, specifying the cost of serving the request from each state. In response, the algorithm selects a state $s_t \in M$, pays cost $d(s_{t-1}, s_t)$ to move the server to s_t, and cost $c_t(s_t)$ to serve the request. The goal is to design an algorithm which minimizes the total cost incurred. Despite its simple definition, MTS captures a broad class of online problems as special cases, including caching [ST85], k-server [MMS88], layered graph traversal [FFK$^+$98], and convex body chasing [FL93].

Traditionally, MTS has mostly been studied in the *adversarial* model, where the request sequence is allowed to be arbitrary, and can even be chosen adversarially. The performance of an online algorithm is then evaluated based on its *competitive ratio*, defined as the worst case ratio between algorithm's cost and the best solution in hindsight [ST85]. This model is attractive because it makes no assumptions on the input, but this flexibility comes at the cost of pessimism. Even though strong impossibility results are known in the adversarial model, practical applications often do not resemble worst-case instances, as the requests are not arbitrary but instead follow an underlying pattern.

In this work, we study MTS under *stochastic requests*. We first consider the case where requests are drawn independently from an unknown distribution. We then consider the richer case where requests are generated by a finite-state Markov chain, and the online algorithm observes both the current request and the current Markov state before making its next move. This formulation captures many practical settings in which requests are generated by an evolving environment. For example, in caching, the MTS state represents the cache contents and Markov requests capture temporal locality in page accesses [KPR00, LPR99, PV25]. In mobility applications such as bike-share, the MTS state is a spatial allocation of resources, while Markov requests model temporal patterns such as commuting, nightlife, weekend demand or weather.

*Authors listed in alphabetical order.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

This stochastic formulation changes the nature of the problem. In the adversarial setting, the central question is how an online algorithm compares to the hindsight optimum on the worst possible instance. In our setting, the goal is to characterize and compute an optimal policy for a fixed environment. The resulting problem is related to average-cost Markov decision processes, as outlined in section 2.

Formally, in the *IID requests* model, we assume that each service cost function c_t is drawn independently from an unknown distribution $\mathcal{D}$, and we assume access to a sampling oracle for $\mathcal{D}$. In the *Markov requests* model, we assume that for a known finite set $\mathcal{X}$, for each $x \in \mathcal{X}$ there exists a distribution $\mathcal{D}_x$ over $\mathcal{X} \times [0, \infty]^M$ such that, conditioning on $x_{t-1} = x$, the pair (x_t, c_t) is sampled from $\mathcal{D}_x$. Given an initial $x_0 \in \mathcal{X}$, this induces a Markov chain $(x_t)_{t \geq 0}$, which we assume to be irreducible. The distributions $(\mathcal{D}_x)_{x \in \mathcal{X}}$ are not known in advance. Note that the IID case is a special case where $|\mathcal{X}| = 1$. We consider three ways of obtaining training data. With *conditional oracles*, the learner chooses a state x and receives an independent pair $(x', c) \sim \mathcal{D}_x$. With a *stationary oracle*, each call independently returns a triple (x, x', c) by first drawing $x \sim p$, where p is the stationary distribution of the Markov chain, and then $(x', c) \sim \mathcal{D}_x$. With an *observed trajectory*, we are given a realization of Markov states $x_0, x_1, \ldots, x_N$ and cost functions $c_1, \ldots, c_N$.

1.1 Our results

In the cases of both IID requests and Markov requests, we show how to find in polynomial time a policy whose expected cost is arbitrarily close to the cost of the best possible *online* algorithm.

We denote by g^* the average cost per step of the optimal online policy. Also, we denote by n the number of points in the underlying metric space and by D its diameter.

Theorem 1 (IID requests)**.** Let $\varepsilon \in (0, D)$ and $\delta \in (0, 1)$. By using $O\left(\frac{D^2}{\varepsilon^2}\left(n \log \frac{D}{\varepsilon} + \log \frac{n}{\delta}\right)\right)$ independent samples drawn from $\mathcal{D}$, with probability at least $1 - \delta$ we find in polynomial time a policy with average cost at most $g^* + \varepsilon$ per step.

Theorem 2. (Markov requests with oracle access). Let $\varepsilon \in (0, D)$, $\delta \in (0, 1)$. With either a conditional oracle or a stationary oracle, by using $O\left(\frac{D^2|\mathcal{X}|^2}{\varepsilon^2}\left(n \log \frac{D}{\varepsilon} + \log \frac{n|\mathcal{X}|}{\delta}\right)\right)$ oracle calls, with probability at least $1 - \delta$ we find in polynomial time a policy with average cost at most $g^* + \varepsilon$ per step.

Theorem 3. (Markov requests with observed trajectory). Suppose the Markov chain $(x_t)_{t \geq 0}$ is aperiodic, and let $\varepsilon \in (0, D)$, $\delta \in (0, 1)$. By observing a trajectory of length $N = \widetilde{O}\left(\frac{D^2|\mathcal{X}|^2 n}{\varepsilon^2} + \tau \frac{D|\mathcal{X}|}{\varepsilon}\right)$, with probability at least $1 - \delta$ we find in polynomial time a policy with average cost at most $g^* + \varepsilon$ per step. Here, τ denotes the mixing time of the underlying Markov chain $(x_t)_t$, and $\widetilde{O}$ suppresses logarithmic factors in $n, |\mathcal{X}|, D, 1/\varepsilon, 1/\delta$, and $1 + \tau$.

1.2 Related work

Adversarial MTS. In the classical adversarial request model, an online algorithm is judged based on its competitive ratio – worst case ratio of algorithm's cost and the cost of the best solution in hindsight. It is well understood what competitive ratios are achievable for MTS in this model: For any n-point metric space, the best competitive ratio of deterministic algorithms is precisely $2n - 1$ [BLS92], while for randomized algorithms it is in $\Omega(\log n) \cap O(\log^2 n)$ (with both the lower and upper bounds being tight for some metrics) [BCLL21, CL19, BCR23]. Better bounds are achievable for special cases of MTS: For example, the k-server problem on an n-point metric is an MTS in an $\binom{n}{k}$-point metric, but admits a deterministic algorithm with competitive ratio k [CKZ26].

Beyond worst-case analysis. The adversarial model has been widely criticized as being "too crude and unrealistic" [KP00], because it often produces overly pessimistic guarantees and can obscure meaningful differences between algorithms that behave quite differently in practice. To overcome this criticism, several research directions have emerged. A prominent one is that of *learning-augmented algorithms*, where online algorithms are given access to (possibly inaccurate) predictions which are meant to expose common patterns in the data and lead to improved performance. There has been extensive work in this setting on MTS [ACE+23b, CSW23, ACE+23a, CE25] and special cases of it, such as k-server [LMS25, CSW23], caching (also known as paging) [LV21, Roh20, Wei20,

ACE$^+$23b, JPS22, BCK$^+$22, ABE$^+$23], convex function chasing [CHW22, LCS$^+$24], and dynamic power management [ACE$^+$21]. Another setting in which MTS has been studied is *smoothed analysis*, where the performance of an algorithm is evaluated on random perturbations of worst-case instances [SS05, CU25].

Another natural approach, which we pursue in this paper, is to **model the input as a stochastic process**, rather than assuming it is generated adversarially. IID models have already been studied in the context of various online problems, including matching [GGPW19], Steiner tree/forest [GGLS08], set cover [GGL$^+$13, GKL24], and special cases of MTS such as k-server [DEH$^+$17, CL06] and list update [Coe26]. Out of all prior work, Markov paging, introduced by [KPR00], is the closest in spirit to our setting. In their setup, requests are generated by a (known) Markov chain on the page universe, and the goal is to minimize the expected cache miss rate. [KPR00] note that the optimal online policy can be found precisely, but in exponential time,[2] and show that an $O(1)$-approximation of the optimal policy can be found in polynomial time. The best polynomial-time approximation ratio was subsequently improved to 4 [LPR99] and recently to 2 [PV25].

Markov decision processes and fixed-point methods. Average-reward Markov decision processes (MDPs) [How60, Bla62] are a classical framework for infinite-horizon decision-making under uncertainty. An MDP transitions between multiple states, where the next state is drawn from a probability distribution that depends on the current state and the chosen action. Each state-action pair provides a certain reward, and the goal is to design a policy which chooses actions in order to maximize the average reward per step over an infinite time-horizon. Classical theory [Put94, Ber95, SB18] characterizes optimal policies through the Bellman optimality equation, which can be solved for example via policy iteration or value iteration. A key technical difference from discounted MDPs is that the average-reward Bellman operator is non-expansive but not necessarily contracting, and therefore ordinary value iteration need not converge. Halpern iteration [Hal67], also known in this context as anchored value iteration, provides a method to overcome this issue by computing approximate fixed points of non-expansive maps. Motivated by reinforcement learning applications, where the environment is not known a priori and must be learned through interaction, a significant line of work, such as [ZX23, ZC24, LBC25, ZC25], studies the sample complexity of computing near-optimal policies under a generative model introduced by [KS98], in which transition probabilities are unknown but independent samples of next states can be obtained for any given state-action pairs.

2 Preliminaries

MDP formulation. MTS in the Markov requests model can be cast as an MDP where each state is represented by a tuple (s, x, c), with $s \in M$, $x \in \mathcal{X}$, and $c\colon M \to \mathbb{R}_{\geq 0} \cup \{\infty\}$, and where each available action corresponds to a server location in M. Since our model allows infinitely many possible cost functions c, the state space of the MDP in this reduction is infinite, and therefore previous algorithms for finding a nearly optimal policy for finite-state MDPs do not apply directly. Our algorithms instead exploit the particular structure of stochastic MTS.

When considering the most general class of online policies, namely *non-stationary* policies, a policy makes a decision based on everything it has observed so far (i.e., its own previous decisions $s_0, \ldots, s_{t-1}$, cost functions $c_1, \ldots, c_t$, and states $x_0, \ldots, x_t$ of the chain generating the requests). Such a policy can be deterministic, mapping a history to the next state, or randomized, mapping the history to a probability distribution over states. If a policy makes a decision by only taking into account s_{t-1}, c_t, and x_t, then the policy is called *stationary*. To define the average cost per step for a policy π, we follow the definition in the MDP literature [Put94]. Let the path induced by π be $(a_t)_{t\geq 0}$. Then the cost of π is defined as

$$g(\pi) = \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, a_t) + c_t(a_t)\right].$$

Overview of techniques. Our analysis has three main ingredients. First, we characterize the structure of optimal policies through optimality equations. These equations draw on classical

[2] Even though a nearly optimal policy can be found in polynomial time for stochastic MTS, this does not imply polynomial time in the natural parameters for paging, as the reduction from paging to MTS incurs an exponential blowup in the state space.

optimality equations [Put94], but in our models they involve averaging the Bellman operator over the randomness in the request process. Second, we follow a standard approach [Put94] and show that a small Bellman residual error leads to a nearly-optimal online policy. Third, we show how to obtain an arbitrarily small Bellman residual error by employing Halpern iteration [Hal67], a method used for finding fixed points in non-expansive maps. Since the request distribution is unknown, we run the iterations using an empirical mean of the Bellman operator applied on samples, and control the discrepancy between the empirical and true operators by concentration bounds.

Paper organization. We first consider the special case of IID requests in section 3. In section 4, we show how our results extend to the case of Markov requests. In section 5, we provide a short experimental evaluation of our proposed algorithms.

3 IID requests

This section is concerned with the case of *IID requests*, where we assume that each service cost function c_t is drawn independently from an unknown distribution $\mathcal{D}$. We assume that g^*, the optimal expected cost per step, is finite, which implies that $\mathbb{E}_{c\sim\mathcal{D}}[\min_s c(s)] < \infty$.

For a service cost function c, the corresponding Bellman operator $B^c\colon \mathbb{R}^M \to \mathbb{R}^M$ is defined by

$$(B^c u)(s) := \min_{s'\in M}\{d(s,s') + c(s') + u(s')\}. \tag{1}$$

For ease of notation, we will henceforth write $B^c u(s)$ to denote $(B^c u)(s)$.

For $u\colon M \to \mathbb{R}$, define the stationary deterministic policy π_u by

$$\pi_u(s,c) := \operatorname*{argmin}_{s'\in M}\{d(s,s') + c(s') + u(s')\}. \tag{2}$$

That is, when the current state is $s \in M$ and the service cost function c is received, the policy π_u moves to $\pi_u(s,c)$. Intuitively, we think of $u(s)$ as an approximation for the cost to be paid to serve the remaining requests if starting from s. Note that shifting u by an additive constant still induces the same policy.

We prove that if u^* satisfies the property below, then the policy π_{u^*} is optimal.

Claim 4 (Structure of optimal policy)**.** Suppose $u^*\colon M \to \mathbb{R}$ and $g^* \in \mathbb{R}$ satisfy

$$\mathbb{E}_{c\sim\mathcal{D}}[B^c u^*(s)] - u^*(s) = g^* \quad \text{for all } s \in M. \tag{3}$$

Then the policy $\pi^* := \pi_{u^*}$ is an optimal policy, even among non-stationary randomized policies, and achieves average cost $g(\pi^*) = g^*$ per step.

Proof. Let $a_0, a_1, \ldots, a_T$ be random variables describing the path taken by some non-stationary, possibly randomized policy π for the first T time steps. Here, a_t depends on $a_0, \ldots, a_{t-1}$ and $c_1, \ldots, c_t$. The cost of this policy up to time T is

$$\mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, a_t) + c_t(a_t)\right].$$

By the definition of π^*, and by denoting $o_t = \pi^*(a_{t-1}, c_t)$ (i.e. what the policy π^* would have done at time t if starting from a_{t-1}), we get

$$d(a_{t-1}, a_t) + c_t(a_t) \geq d(a_{t-1}, o_t) + c_t(o_t) + u^*(o_t) - u^*(a_t). \tag{4}$$

Plugging this in, we get that the expected cost of policy π for the first T time steps is at least

$$\mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, o_t) + c_t(o_t) + u^*(o_t) - u^*(a_t)\right].$$

By the definition of o_t and π^*, this is equal to

$$\mathbb{E}\left[\sum_{t=1}^{T} B^{c_t} u^*(a_{t-1}) - u^*(a_t)\right].$$

Since $\mathbb{E}[B^{c_t}u^*(a_{t-1}) \mid a_{t-1}] = u^*(a_{t-1}) + g^*$ by (3), we get that the expression above is equal to

$$\sum_{t=1}^{T} \mathbb{E}\left[\mathbb{E}\left[B^{c_t}u^*(a_{t-1}) \mid a_{t-1}\right]\right] - \mathbb{E}[u^*(a_t)] = \sum_{t=1}^{T} \mathbb{E}\left[u^*(a_{t-1}) + g^* - u^*(a_t)\right]$$
$$= \mathbb{E}[u^*(a_0) - u^*(a_T)] + g^*T.$$

By dividing by T and letting $T \to \infty$, the first term vanishes and we obtain $g(\pi) \geq g^*$.

Repeating essentially the same calculations with a_t defined recursively by $a_t := \pi^*(a_{t-1}, c_t)$, we get that the average cost per step for policy π^* is $g(\pi^*) = g^*$. This is because $o_t = a_t$, and therefore the only inequality involved, namely (4), turns into equality. □

By defining the operator $\bar{B}$ such that $\bar{B}u := \mathbb{E}_{c\sim\mathcal{D}}[B^c u]$, (3) can be interpreted as u^* being a fixed point of $\bar{B}$, up to an additive constant g^*. We next show the existence of such a pair (u^*, g^*).

Claim 5 (Existence of an optimal policy)**.** There exist u^* and g^* satisfying (3), with $\|u^*\|_\infty \leq D$.

Proof. Let $H := \{u\colon M \to [0, D]\}$, and let $T\colon H \to H$ be defined by $Tu := \bar{B}u - \min(\bar{B}u)$. Note that $Tu \in H$ for any u because $B^c u$ is 1-Lipschitz with respect to the underlying metric space, which implies that Tu is 1-Lipschitz as well. Since H is convex and compact, and T is continuous, by Brouwer's fixed-point theorem we know that T has a fixed point $u^* \in H$. By setting $g^* := \min(\bar{B}u^*)$, we get $\bar{B}u^* \equiv u^* + g^*$, as required in (3). □

Note that g^* is unique, but u^* is not necessarily unique. Next, we bound the error of a policy π_u.

Claim 6 (Bounding the policy error)**.** Let π_u be the policy induced by u, and let $g(\pi_u)$ be its cost. Then

$$g(\pi_u) - g^* \leq \|\bar{B}u - u\|_{\text{sp}},$$

where $\|\cdot\|_{\text{sp}}$ denotes the span seminorm, defined by

$$\|u\|_{\text{sp}} := \max(u) - \min(u).^3$$

Proof. Let a be the trajectory induced by π_u, recursively defined by $a_t := \pi_u(a_{t-1}, c_t)$. The expected cost of π_u for the first T time steps is

$$\mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, a_t) + c_t(a_t)\right] = \mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, a_t) + c_t(a_t) + u(a_t) - u(a_t)\right]$$
$$= \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t}u(a_{t-1}) - u(a_t)\right]$$
$$= \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t}u(a_{t-1}) - u(a_{t-1})\right] + \mathbb{E}[u(a_0) - u(a_T)]$$
$$= \mathbb{E}\left[\sum_{t=1}^{T} \bar{B}u(a_{t-1}) - u(a_{t-1})\right] + \mathbb{E}[u(a_0) - u(a_T)]$$
$$\leq T\max(\bar{B}u - u) + \mathbb{E}[u(a_0) - u(a_T)].$$

Here, in the second equality we used the definitions of $B^{c_t}u$ and a_t, and in the penultimate step we used the definition of $\bar{B}$. Dividing by T and letting $T \to \infty$, we obtain

$$g(\pi_u) \leq \max(\bar{B}u - u). \tag{5}$$

[3]Throughout the paper, we use $\max(u)$ and $\min(u)$ as a shorthand for $\max_{s\in M} u(s)$ and $\min_{s\in M} u(s)$, respectively.

Let o be the trajectory induced by π^*, recursively defined by $o_t := \pi^*(o_{t-1}, c_t)$. Similarly to the above, we show that the expected cost of policy π^* for the first T time steps is

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} d(o_{t-1}, o_t) + c_t(o_t)\right] &= \mathbb{E}\left[\sum_{t=1}^{T} d(o_{t-1}, o_t) + c_t(o_t) + u(o_t) - u(o_t)\right] \\
&\geq \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t} u(o_{t-1}) - u(o_t)\right] \\
&= \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t} u(o_{t-1}) - u(o_{t-1})\right] + \mathbb{E}[u(o_0) - u(o_T)] \\
&= \mathbb{E}\left[\sum_{t=1}^{T} \bar{B}u(o_{t-1}) - u(o_{t-1})\right] + \mathbb{E}[u(o_0) - u(o_T)] \\
&\geq T \min(\bar{B}u - u) + \mathbb{E}[u(o_0) - u(o_T)].
\end{aligned}$$

Dividing by T and letting $T \to \infty$, we obtain

$$g^* = g(\pi^*) \geq \min(\bar{B}u - u). \tag{6}$$

Putting (5) and (6) together, we obtain

$$\min(\bar{B}u - u) \leq g^* \leq g(\pi_u) \leq \max(\bar{B}u - u),$$

from which we conclude that

$$g(\pi_u) - g^* \leq \|\bar{B}u - u\|_{\text{sp}}. \qquad \square$$

Equipped with Claim 6, we now seek to find u for which $\|\bar{B}u - u\|_{\text{sp}} \leq \varepsilon$. To this end, we make use of Halpern iteration and its convergence guarantees.

Theorem 7. (Halpern residual bound). Let $F\colon Z \to Z$ be a non-expansive map defined on space Z with associated seminorm $\|\cdot\|$. Let $C \subseteq Z$ be a convex set, and let $\operatorname{diam}(C) := \sup_{z,z' \in C} \|z - z'\|$. Let $(\alpha_k)_{k \geq 1}$ be a step size sequence with $\alpha_k \in (0, 1)$. The Halpern iteration [Hal67] with anchor at 0 and starting point $z_0 := 0$ produces a sequence $(z_k)_{k \geq 0}$ defined by

$$z_k := (1 - \alpha_k) F z_{k-1}, \qquad k \geq 1. \tag{7}$$

Suppose that $0 \in C$, and $z \in C$ implies $Fz \in C$. Then, for the choice

$$\alpha_k = \frac{2}{k+2}, \qquad k \geq 1,$$

we have

$$\|z_k - F z_k\| \leq \frac{4 \cdot \operatorname{diam}(C)}{k+1}.$$

To the best of our knowledge, [SS17] were the first to prove this bound for general normed spaces, as previous work had focused on Hilbert spaces and smooth Banach spaces. For completeness, we include a proof of Theorem 7 in Appendix B.

In order to apply Halpern iteration, we show that the Bellman operator is non-expansive.

Claim 8. B^c is non-expansive with respect to $\|\cdot\|_{\text{sp}}$, for any $c\colon M \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ with $\min(c) < \infty$.

Proof. Let $u, v \in \mathbb{R}^M$. Fix $s \in M$ and let $r \in M$ such that $B^c u(s) = d(s, r) + c(r) + u(r)$.

Then $B^c v(s) \leq d(s, r) + c(r) + v(r)$, and therefore

$$B^c v(s) - B^c u(s) \leq v(r) - u(r) \leq \max(v - u).$$

Similarly,

$$B^c v(s) - B^c u(s) \geq \min(v - u),$$

and we can conclude that

$$\|B^c v - B^c u\|_{\text{sp}} \leq \|v - u\|_{\text{sp}}. \qquad \square$$

If $\mathcal{D}$ were known, and if we could compute $\bar{B}u$ for any u, then we could find an approximately optimal online policy by running Halpern iteration for $\bar{B}$. But since we assume a weaker, generative model, we do not know $\mathcal{D}$. Instead, we sample N cost functions $c_1, \ldots, c_N$ from $\mathcal{D}$, where N will be determined later. We then define an empirical counterpart to $\bar{B}$, by

$$\widetilde{B}u := \frac{1}{N}\sum_{i=1}^{N} B^{c_i}u.$$

We determine how many samples N are necessary to ensure that $\widetilde{B}$ is a good approximation of $\bar{B}$.

Claim 9. (Sample complexity). Let $C := \{u \in \mathbb{R}^M : \|u\|_{\text{sp}} \le D\}$. Let $\delta \in (0,1)$ and $\varepsilon' > 0$. Suppose that

$$N \ge \frac{8D^2}{\varepsilon'^2}\left(n \log\left(\left\lceil \frac{16D}{\varepsilon'}\right\rceil + 1\right) + \log\frac{2n}{\delta}\right).$$

Then, with probability at least $1-\delta$ it holds simultaneously for all $u \in C$ that $\|\widetilde{B}u - \bar{B}u\|_{\text{sp}} \le \varepsilon'$.

Claim 9 follows from Claim 15, proved in Appendix A for the more general setting of Markov requests.

Putting everything together, we are now ready to prove Theorem 1.

Proof of Theorem 1. We apply Theorem 7 using $\widetilde{B}$. Note that $\widetilde{B}$ is non-expansive with respect to $\|\cdot\|_{\text{sp}}$, because it is an average of the maps $B^{c_1}, \ldots, B^{c_N}$, which are non-expansive by Claim 8.

Let $C := \{u \in \mathbb{R}^M : \|u\|_{\text{sp}} \le D\}$. Since $B^c u$ is 1-Lipschitz with respect to the underlying metric, we have $\|B^c u\|_{\text{sp}} \le D$ for every u, so $\widetilde{B}$ maps C to itself. Moreover, $\text{diam}(C) = 2D$.

Run the Halpern iteration for $k := \lceil \frac{16D}{\varepsilon} \rceil$ steps. By Theorem 7, the empirical residual error satisfies

$$\|z_k - \widetilde{B}z_k\|_{\text{sp}} \le \frac{4 \cdot \text{diam}(C)}{k+1} \le \frac{\varepsilon}{2}. \tag{8}$$

Combining (8) with Claim 9 applied with $\varepsilon' = \varepsilon/2$, and by using the triangle inequality, we get that

$$\|z_k - \bar{B}z_k\|_{\text{sp}} \le \varepsilon$$

holds with probability at least $1-\delta$. Finally, by Claim 6, we obtain that the policy π_{z_k} induced by z_k satisfies $g(\pi_{z_k}) \le g^* + \varepsilon$ with probability at least $1-\delta$. □

4 Markov requests

We consider requests generated by a Markov chain $(x_t)_{t\ge 0}$ on a known finite state space $\mathcal{X}$. Conditional on $x_{t-1} = x$, the pair (x_t, c_t) is drawn from $\mathcal{D}_x$, independently of the preceding history. The chain is irreducible, with stationary distribution p, and the online algorithm observes both x_t and c_t before choosing its move. The distributions $(\mathcal{D}_x)_{x\in\mathcal{X}}$ and p are unknown. We assume that g^*, the optimal expected cost per step, is finite. By irreducibility, this implies that $\mathbb{E}_{(x',c)\sim\mathcal{D}_x}[\min_{s\in M} c(s)] < \infty$ for every $x \in \mathcal{X}$, ensuring that the Bellman expectations below are finite.

Instead of identifying a stationary policy by a function $u : M \to \mathbb{R}$ as in the IID case, we now need to consider a separate function $u_x : M \to \mathbb{R}$ for each $x \in \mathcal{X}$. We think of $u_x(s)$ as an approximation for the relative cost to be paid to serve the remaining requests if the server is located at $s \in M$ and the chain is in state $x \in \mathcal{X}$. Each such u induces a stationary deterministic policy π_u, defined by

$$\pi_u(s,c,x) := \operatorname*{argmin}_{s'\in M}\{d(s,s') + c(s') + u_x(s')\}. \tag{9}$$

That is, when the MTS state is $s \in M$, the next revealed cost function is c, and the next revealed Markov chain state is x, the policy π_u moves to $\pi_u(s,c,x)$. Note that adding a constant to any u_x does not change the induced policy.

We define

$$(\overline{B}u)_x := \mathbb{E}_{(x',c)\sim\mathcal{D}_x}[B^c u_{x'}],$$

where B^c is defined in (1). The following claims extend the corresponding results of Section 3, and their proofs can be found in Appendix A.1.

Claim 10. (Structure of an optimal policy). Suppose $u^* \in (\mathbb{R}^M)^{\mathcal{X}}$ and $\lambda^* \in \mathbb{R}^{\mathcal{X}}$ satisfy

$$(\overline{B}u^*)_x(s) - u^*_x(s) = \lambda^*_x \qquad \text{for every } x \in \mathcal{X}, s \in M. \tag{10}$$

Then π_{u^*} is optimal, even among non-stationary randomized policies, and $g^* := g(\pi_{u^*}) = \sum_x p_x \lambda^*_x$.

Claim 11. (Existence of an optimal policy). There exist u^* and λ^* satisfying (10).

For all three observation models that we consider, we denote the observations by $(x_i, c_i, x'_i)_{i=1}^N$. The way in which x_i is obtained differs between models: with conditional oracles, x_i can be chosen by the learner; with a stationary oracle, x_i is drawn from the stationary distribution p independently from previous history; in the observed trajectory model, $x_{i+1} = x'_i$. In every model, conditional on x_i and the preceding observations, the pair (x'_i, c_i) is drawn from $\mathcal{D}_{x_i}$.

Let $I_x := \{(x'_i, c_i) : x_i = x\}$ be the multiset of observations with source state x, and let $N_x := |I_x|$. Define

$$(\widetilde{B}u)_x := \begin{cases} \dfrac{1}{N_x} \displaystyle\sum_{(x',c) \in I_x} B^c u_{x'}, & N_x > 0, \\ 0, & N_x = 0. \end{cases} \tag{11}$$

Claim 12. (Non-expansiveness). $\widetilde{B}$ is non-expansive under $\|\cdot\|$, where $\|v\| := \max_{x \in \mathcal{X}} \|v_x\|_{\text{sp}}$.

The following bound weights estimation errors by the stationary probabilities, allowing states that occur less frequently to be estimated less accurately.

Claim 13. (Bounding the policy error). For every $u \in (\mathbb{R}^M)^{\mathcal{X}}$, conditional on observations $(I_x)_{x \in \mathcal{X}}$,

$$g(\pi_u) - g^* \le \sum_{x \in \mathcal{X}} p_x \|(\overline{B}u)_x - u_x\|_{\text{sp}} \le \|\widetilde{B}u - u\| + \sum_{x \in \mathcal{X}} p_x \|(\overline{B}u)_x - (\widetilde{B}u)_x\|_{\text{sp}}. \tag{12}$$

Lemma 14. Let $C := \{u \in (\mathbb{R}^M)^{\mathcal{X}} \colon \|u_x\|_{\text{sp}} \le D \text{ for all } x \in \mathcal{X}\}$. If

$$\sum_x p_x \|(\overline{B}u)_x - (\widetilde{B}u)_x\|_{\text{sp}} \le \varepsilon/2 \text{ for all } u \in C,$$

the policy π_{z_k} obtained after running $k = O(D/\varepsilon)$ Halpern iterations has cost at most $g^* + \varepsilon$.

Proof. Since $\|B^c u_{x'}\|_{\text{sp}} \le D$ for all $(x', c) \in I_x$, $\widetilde{B}$ maps C to itself. To bound $\|(\widetilde{B}u) - u\|$, we run Halpern iteration (7) using $\widetilde{B}$ and set $u := z_k$ for $k := \lceil 16D/\varepsilon \rceil$. Since C is convex, contains 0, and has diameter at most $2D$ with respect to $\|\cdot\|$, Theorem 7 and Claim 12 give $z_k \in C$ and

$$\|\widetilde{B}z_k - z_k\| \le \frac{8D}{k+1} \le \frac{\varepsilon}{2}.$$

Combining our hypothesis applied to $u = z_k$ with Claim 13 concludes the proof. □

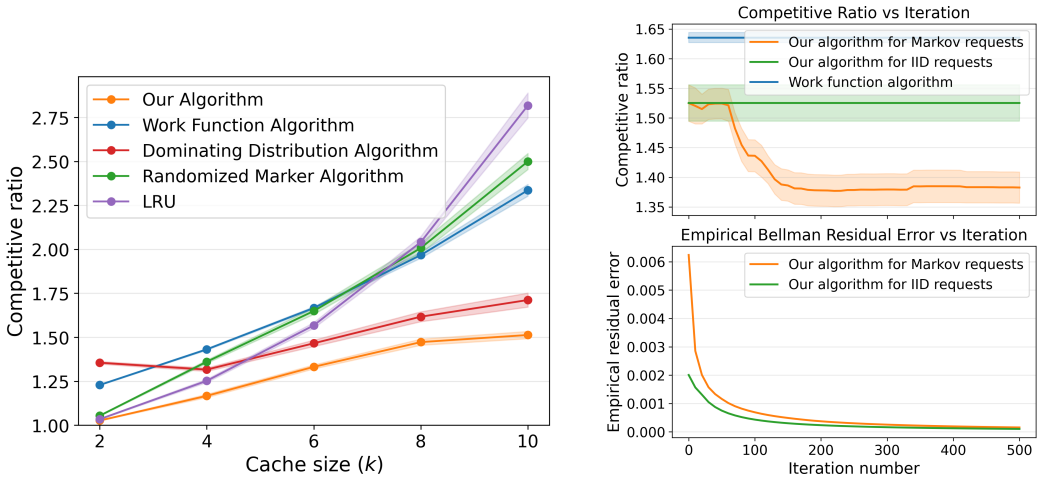
To complete the proofs of Theorems 2 and 3, it suffices to show that, with probability at least $1 - \delta$, we have $\sum_x p_x \|(\overline{B}u)_x - (\widetilde{B}u)_x\|_{\text{sp}} \le \varepsilon/2$ for all $u \in C$. Since the bound holds simultaneously for all $u \in C$, it applies to z_k even though this iterate depends on the observations $(x_i, c_i, x'_i)_{i=1}^N$. We prove the bound in Appendix A.2.

5 Experiments

5.1 Synthetic caching

We evaluate our algorithm on an important special case of MTS, namely the caching problem. In this problem, there are n pages and k cache slots. Each request consists of a page. If the requested page is not in the cache, a cache miss occurs and unit cost must be paid. In this case, the page must be brought into the cache, and a different page might need to be evicted to free up space.

We conduct a synthetic experiment in which the requests are generated according to a Markov chain on the pages. The pages are arranged in a circle, and the probability that j is requested next, conditioned



(a) Experiment 1: Markov caching. Comparison of the competitive ratios of algorithms, as the cache size varies.

(b) Experiment 2: citibike k-server. The competitive ratios and empirical Bellman residual errors, as a function of the number of Halpern iterations.

Figure 1: Results of our experiments. Shades denote standard error of the mean, computed using numpy.std() and dividing by the square root of the number of runs.

on the current request being i, decays exponentially with $i \cdot \text{dist}(i, j)$. This construction yields a family of conditional distributions of the next request ranging from uniform (for $i = 0$) to highly concentrated (for large i). We compare our algorithm against: 1) the Dominating Distribution Algorithm (DDA), which is guaranteed to be within a factor of 2 of the optimal online policy [PV25]; 2) the Work Function Algorithm (WFA), (one of) the best possible classical online deterministic algorithms, obtaining a competitive ratio of k [CKZ26]; 3) the Least Recently Used (LRU) Algorithm, which upon a cache miss evicts the page which was requested least recently, also achieving a competitive ratio of k [ST85] and often obtaining a very good performance in practice; 4) the Randomized Marker Algorithm, which achieves the optimal competitive ratio of $O(\log k)$ among randomized algorithms [FKL$^+$91]. DDA is given access to the transition matrix of the underlying Markov chain, while our algorithm operates without such knowledge and instead learns from samples. We set the number of pages to $n = 12$ and vary the size k of the cache. We run 20 tests for each value of k, each test consisting of 10.000 requests. In each test, our algorithm uses 500 samples for each Markov state and runs Halpern iteration for 100 steps.

Fig. 1a shows the empirical competitive ratios obtained by the algorithms. The competitive ratio is defined as the ratio between the cost of an online algorithm and the cost of the optimal offline algorithm.[4] We compute the latter quantity by dynamic programming. We observe that our algorithm consistently has the best performance. The factors of variability in this experiment are the randomly generated problem instance and the internal randomness used by DDA and the Randomized Marker Algorithm.

5.2 k-server on a real-world dataset

We additionally evaluate our algorithms on a real-world dataset from a popular bike-sharing platform citibike[cit25]. We consider the k-server problem [MMS88], a central problem in online algorithms and a special case of MTS: There are k servers which must be moved in a metric space of size n to serve incoming requests. Each request is a point in the metric space, and must be served by moving one of the servers there. The goal is to minimize the total distance traveled by the servers. We consider the k-server problem with $k = n - 2$, which implies that at any moment there are precisely 2 locations not covered by the servers. This is also known as the 2-evader problem [KP96].

[4]We use the optimal offline algorithm as a benchmark in all of our experiments because the experiments with real-world data are not generated by an actual Markov chain, so there is no optimal online policy to compare against.

We use the citibike dataset as follows: We set $n = 10$ and take the top 10 most popular starting stations with Euclidean distance to be our metric space. We then map each trip starting in one of these stations to a request. We model temporal patterns in the requests by using a Markov chain with state space $\{(w, h) \mid w \in \{0, 1\}, h \in \{0, \ldots, 23\}\}$, where w indicates whether the request occurred on a weekday or weekend, and h indicates the hour of the day. Here, w and h are observed context labels used to construct an approximate Markov model. We run our algorithm for Markov requests (where we create for each request an observed Markov state as described above), our algorithm for IID requests, and the Work Function Algorithm (WFA). The latter obtains the optimal competitive ratio of $n - 2$ among classical deterministic online algorithms for the 2-evader problem [KP96]. We create 12 instances, corresponding to the months of 2025, and evaluate the performance of our algorithms using leave-one-out cross validation. We run 500 Halpern iterations and evaluate our policies after every 10 iterations.

The results are shown in Fig. 1b. We observe that our algorithm which assumes IID requests already outperforms WFA. Our algorithm for Markov requests achieves further improvements, despite relying on a relatively simple Markov chain model. We also report the empirical residual error of the Halpern iterates, namely $\|\widetilde{B}z_k - z_k\|$, which approaches 0 as we find an approximate fixed point of $\widetilde{B}$. This experiment suggests that algorithms designed for the Markov setting (which is often an imperfect model of reality) perform well beyond this idealized abstraction.

The experiments were run on a personal computer with 32 GB RAM and completed in a few hours.

6 Limitations and conclusion

We develop a general approach for efficiently learning nearly optimal online policies for MTS under IID and Markov requests. This includes applications such as systems with metric switching costs between operating modes, problems involving the location of a single movable object (i.e. process migration), and dynamic power management.

However, our guarantees do not yield sample complexity or running time polynomial in the natural parameters of every special case of MTS. For example, in the reduction from k-server, each MTS state represents an entire server configuration, so the number of states grows exponentially in k and can make our approach inefficient. This limitation parallels competitive analysis, where exploiting the structure of particular problems can give substantially stronger guarantees than a direct reduction to general MTS.

Perhaps the most fascinating open problem motivated by our work is whether sample complexities polynomial in natural problem parameters of MTS subproblems with special structure can be obtained. A concrete question is whether k-server admits a polynomial time algorithm for computing an $O(1)$-approximation of the optimal policy under Markov requests.

Acknowledgments and disclosures

ChatGPT 6 Astra and predecessor models were used to assist with standard concentration and estimation arguments in Appendix A.2, as well as with sentence-level rephrasing. The authors independently checked all AI-assisted material and substantially revised it before inclusion in the manuscript. All main ideas in the paper are due to the authors.

Christian Coester is funded by the European Union (ERC, CCOO, 101165139). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

[ABE+23] Antonios Antoniadis, Joan Boyar, Marek Eliás, Lene Monrad Favrholdt, Ruben Hoeksma, Kim S. Larsen, Adam Polak, and Bertrand Simon. Paging with succinct predictions. In *International Conference on Machine Learning, ICML*, pages 952–968, 2023.

[ACE+21] Antonios Antoniadis, Christian Coester, Marek Eliás, Adam Polak, and Bertrand Simon. Learning-augmented dynamic power management with multiple states via new ski rental bounds. In *Advances in Neural Information Processing Systems 34, NeurIPS*, pages 16714–16726, 2021.

[ACE+23a] Antonios Antoniadis, Christian Coester, Marek Eliás, Adam Polak, and Bertrand Simon. Mixing predictions for online metric algorithms. In *International Conference on Machine Learning, ICML*, pages 969–983, 2023.

[ACE+23b] Antonios Antoniadis, Christian Coester, Marek Eliás, Adam Polak, and Bertrand Simon. Online metric algorithms with untrusted predictions. *ACM Trans. Algorithms*, 19(2):19:1–19:34, 2023.

[BCK+22] Nikhil Bansal, Christian Coester, Ravi Kumar, Manish Purohit, and Erik Vee. Learning-augmented weighted paging. In *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022*, pages 67–89, 2022.

[BCLL21] Sébastien Bubeck, Michael B. Cohen, James R. Lee, and Yin Tat Lee. Metrical task systems on trees via mirror descent and unfair gluing. *SIAM Journal on Computing*, 50(3):909–923, 2021.

[BCR23] Sébastien Bubeck, Christian Coester, and Yuval Rabani. The randomized k-server conjecture is false! In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023*, pages 581–594. ACM, 2023.

[Ber95] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 1995.

[Bla62] David Blackwell. Discrete dynamic programming. *The Annals of Mathematical Statistics*, 33(2):719–726, 1962.

[BLS92] Allan Borodin, Nathan Linial, and Michael E. Saks. An optimal on-line algorithm for metrical task system. *J. ACM*, 39(4):745–763, 1992.

[CE25] Matei Gabriel Cosa and Marek Eliás. Learning-augmented algorithms for MTS with bandit access to multiple predictors. In *Forty-second International Conference on Machine Learning, ICML*, 2025.

[CHW22] Nicolas Christianson, Tinashe Handina, and Adam Wierman. Chasing convex bodies and functions with black-box advice. In *Conference on Learning Theory, COLT*, pages 867–908, 2022.

[cit25] Citi bike system data. `https://citibikenyc.com/system-data`, 2025. Accessed: 2026-05.

[CKZ26] Christian Coester, Elias Koutsoupias, and Marek Zbysiński. The k-server conjecture is true. *arXiv preprint arXiv:2609.15979*, 2026.

[CL06] Randy Cogill and Sanjay Lall. On decentralized policies for the stochastic k-server problem. *arXiv preprint arXiv:math/0605188*, 2006.

[CL19] Christian Coester and James R. Lee. Pure entropic regularization for metrical task systems. In *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 835–848. PMLR, 2019.

[CLLM12] Kai-Min Chung, Henry Lam, Zhenming Liu, and Michael Mitzenmacher. Chernoff–Hoeffding Bounds for Markov Chains: Generalized and Simplified. In *29th International Symposium on Theoretical Aspects of Computer Science (STACS 2012)*, volume 14 of *Leibniz International Proceedings in Informatics*, pages 124–135, 2012.

[Coe26] Christian Coester. Transposition is nearly optimal for IID list update. *CoRR*, abs/2603.10244, 2026.

[CSW23] Nicolas Christianson, Junxuan Shen, and Adam Wierman. Optimal robustness-consistency tradeoffs for learning-augmented metrical task systems. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 9377–9399. PMLR, 2023.

[CU25] Christian Coester and Jack Umenberger. Smoothed analysis of online metric problems. In *33rd Annual European Symposium on Algorithms, ESA 2025*, LIPIcs, pages 115:1–115:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025.

[DEH+17] Sina Dehghani, Soheil Ehsani, MohammadTaghi Hajiaghayi, Vahid Liaghat, and Saeed Seddighin. Stochastic k-server: How should uber work? *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, 80:126:1–126:14, 2017.

[FFK+98] Amos Fiat, Dean P. Foster, Howard J. Karloff, Yuval Rabani, Yiftach Ravid, and Sundar Vishwanathan. Competitive algorithms for layered graph traversal. *SIAM J. Comput.*, 28(2):447–462, 1998.

[FKL+91] Amos Fiat, Richard M Karp, Michael Luby, Lyle A McGeoch, Daniel D Sleator, and Neal E Young. Competitive paging algorithms. *Journal of Algorithms*, 12(4):685–699, 1991.

[FL93] J. Friedman and N. Linial. On convex body chasing. *Discrete and computational geometry*, 9(3):293–322, 1993.

[GGL+13] Fabrizio Grandoni, Anupam Gupta, Stefano Leonardi, Pauli Miettinen, Piotr Sankowski, and Mohit Singh. Set covering with our eyes closed. *SIAM J. Comput.*, 42(3):808–830, 2013.

[GGLS08] Naveen Garg, Anupam Gupta, Stefano Leonardi, and Piotr Sankowski. Stochastic analyses for online combinatorial optimization problems. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2008*, pages 942–951. SIAM, 2008.

[GGPW19] Anupam Gupta, Guru Guruganesh, Binghui Peng, and David Wajc. Stochastic Online Metric Matching. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 67:1–67:14, Dagstuhl, Germany, 2019. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[GKL24] Anupam Gupta, Gregory Kehne, and Roie Levin. Set covering with our eyes wide shut. In *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024*, pages 4530–4553. SIAM, 2024.

[Hal67] Benjamin Halpern. Fixed points of nonexpanding maps. *Bulletin of the American Mathematical Society*, 73(6):957–961, 1967.

[How60] Ronald A. Howard. *Dynamic Programming and Markov Processes*. John Wiley & Sons, New York, 1960.

[JPS22] Zhihao Jiang, Debmalya Panigrahi, and Kevin Sun. Online algorithms for weighted paging with predictions. *ACM Trans. Algorithms*, 18(4):39:1–39:27, 2022.

[KP96] Elias Koutsoupias and Christos Papadimitriou. The 2-evader problem. *Inf. Process. Lett.*, 57(5):249–252, March 1996.

[KP00] Elias Koutsoupias and Christos H. Papadimitriou. Beyond competitive analysis. *SIAM Journal on Computing*, 30(1):300–317, 2000.

[KPR00] Anna R. Karlin, Steven J. Phillips, and Prabhakar Raghavan. Markov paging. *SIAM J. Comput.*, 30(3):906–922, 2000.

[KS98] Michael Kearns and Satinder Singh. Finite-sample convergence rates for q-learning and indirect algorithms. In *Advances in Neural Information Processing Systems*, volume 11. MIT Press, 1998.

[LBC25] Jongmin Lee, Mario Bravo, and Roberto Cominetti. Near-optimal sample complexity for mdps via anchoring. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025.

[LCS+24] Adam Lechowicz, Nicolas Christianson, Bo Sun, Noman Bashir, Mohammad Hajiesmaili, Adam Wierman, and Prashant J. Shenoy. Chasing convex functions with long-term constraints. In *Forty-first International Conference on Machine Learning, ICML*, 2024.

[LMS25] Alexander Lindermayr, Nicole Megow, and Bertrand Simon. Boosting double coverage for k-server via imperfect predictions. *Algorithmica*, 87(11):1477–1517, 2025.

[LP17] David A. Levin and Yuval Peres. *Markov Chains and Mixing Times*. American Mathematical Society, 2 edition, 2017. With contributions by Elizabeth L. Wilmer.

[LPR99] Carsten Lund, Steven Phillips, and Nick Reingold. Paging against a distribution and ip networking. *J. Comput. Syst. Sci.*, 58(1):222–231, February 1999.

[LV21] Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. *J. ACM*, 68(4):24:1–24:25, 2021.

[MMS88] Mark Manasse, Lyle McGeoch, and Daniel Sleator. Competitive algorithms for on-line problems. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, pages 322–333, New York, NY, USA, 1988. Association for Computing Machinery.

[Put94] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics. Wiley, 1994.

[PV25] Chirag Pabbaraju and Ali Vakilian. New and improved bounds for markov paging. In *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025*, LIPIcs, pages 123:1–123:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025.

[Roh20] Dhruv Rohatgi. Near-optimal bounds for online caching with machine learned advice. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1834–1845, 2020.

[SB18] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning - an introduction, 2nd Edition*. MIT Press, 2018.

[SS05] Guido Schäfer and Naveen Sivadasan. Topology matters: Smoothed competitiveness of metrical task systems. *Theor. Comput. Sci.*, 341(1-3):216–246, 2005.

[SS17] Shoham Sabach and Shimrit Shtern. A first order method for solving convex bilevel optimization problems. *SIAM Journal on Optimization*, 27(2):640–660, 2017.

[ST85] Daniel D. Sleator and Robert E. Tarjan. Amortized efficiency of list update and paging rules. *Commun. ACM*, 28(2):202–208, February 1985.

[Wei20] Alexander Wei. Better and simpler learning-augmented online caching. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, pages 60:1–60:17, 2020.

[ZC24] Matthew Zurek and Yudong Chen. Span-based optimal sample complexity for weakly communicating and general average reward mdps. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024*, 2024.

[ZC25] Matthew Zurek and Yudong Chen. The plug-in approach for average-reward and discounted mdps: Optimal sample complexity analysis. In *Proceedings of The 36th International Conference on Algorithmic Learning Theory*, volume 272 of *Proceedings of Machine Learning Research*, pages 1386–1387. PMLR, 2025.

[ZX23] Zihan Zhang and Qiaomin Xie. Sharper model-free reinforcement learning for average-reward markov decision processes. In *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pages 5476–5477. PMLR, 12–15 Jul 2023.

A Omitted proofs from section 4

A.1 Structural properties and policy error bounds

Claim 10. (Structure of an optimal policy). Suppose $u^* \in (\mathbb{R}^M)^{\mathcal{X}}$ and $\lambda^* \in \mathbb{R}^{\mathcal{X}}$ satisfy

$$(\overline{B}u^*)_x(s) - u^*_x(s) = \lambda^*_x \qquad \text{for every } x \in \mathcal{X}, s \in M. \tag{10}$$

Then π_{u^*} is optimal, even among non-stationary randomized policies, and $g^* := g(\pi_{u^*}) = \sum_x p_x \lambda^*_x$.

Proof. The proof is similar to the proof of Claim 4. Let $a_0, a_1, \ldots, a_T$ be random variables describing the path taken by some non-stationary, possibly randomized policy π for the first T time steps. Here, a_t depends on $a_0, \ldots, a_{t-1}$, $c_1, \ldots, c_t$, and $x_0, \ldots, x_t$. The cost of this policy up to time T is

$$\mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, a_t) + c_t(a_t)\right].$$

By the definition of π_{u^*}, and by denoting $o_t = \pi_{u^*}(a_{t-1}, c_t, x_t)$ (i.e. what the policy π_{u^*} would have done at time t if starting from a_{t-1}), we get

$$d(a_{t-1}, a_t) + c_t(a_t) \geq d(a_{t-1}, o_t) + c_t(o_t) + u^*_{x_t}(o_t) - u^*_{x_t}(a_t). \tag{13}$$

Plugging this in, we get that the expected cost of policy π for the first T time steps is at least

$$\mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, o_t) + c_t(o_t) + u^*_{x_t}(o_t) - u^*_{x_t}(a_t)\right].$$

By the definition of o_t, this is equal to

$$\mathbb{E}\left[\sum_{t=1}^{T} B^{c_t} u^*_{x_t}(a_{t-1}) - u^*_{x_t}(a_t)\right].$$

Since $\mathbb{E}[B^{c_t} u^*_{x_t}(a_{t-1}) \mid a_{t-1}, x_{t-1}] = u^*_{x_{t-1}}(a_{t-1}) + \lambda^*_{x_{t-1}}$ by (10), we get that the expression above is equal to

$$\begin{aligned}\sum_{t=1}^{T} \mathbb{E}\left[\mathbb{E}\left[B^{c_t} u^*_{x_t}(a_{t-1}) \mid a_{t-1}, x_{t-1}\right]\right] - \mathbb{E}[u^*_{x_t}(a_t)] &= \sum_{t=1}^{T} \mathbb{E}\left[u^*_{x_{t-1}}(a_{t-1}) + \lambda^*_{x_{t-1}}\right] - \mathbb{E}\left[u^*_{x_t}(a_t)\right] \\ &= \mathbb{E}[u^*_{x_0}(a_0) - u^*_{x_T}(a_T)] + \mathbb{E}\left[\sum_{t=0}^{T-1} \lambda^*_{x_t}\right].\end{aligned}$$

Note that the first term in the expression above is bounded. Dividing by T and letting $T \to \infty$, the first term vanishes and the second term converges to $\sum_x p_x \lambda^*_x$. Therefore, we obtain $g(\pi) \geq \sum_x p_x \lambda^*_x$.

Repeating essentially the same calculations with a_t defined recursively by $a_t := \pi_{u^*}(a_{t-1}, c_t, x_t)$, we get that the average cost per step for policy π_{u^*} is $g(\pi_{u^*}) = \sum_{x \in \mathcal{X}} p_x \lambda^*_x$. This is because $o_t = a_t$, and hence inequality (13) turns into equality, and all other equations were already equalities. □

Claim 11. (Existence of an optimal policy). There exist u^* and λ^* satisfying (10).

Proof. Let $H := ([0, D]^M)^{\mathcal{X}}$, and define the auxiliary operator $T : H \to H$ by

$$(Tu)_x := (\bar{B}u)_x - \min((\bar{B}u)_x).$$

Note that T is a normalized version of $\bar{B}$, where normalization takes place along each component $x \in \mathcal{X}$ separately. Since H is convex and compact, and T is continuous, by Brouwer's fixed-point theorem we know that T has a fixed point $u^* \in H$. Setting $\lambda^*_x := \min((\bar{B}u^*)_x)$ completes the proof. □

Claim 12. (Non-expansiveness). $\widetilde{B}$ is non-expansive under $\|\cdot\|$, where $\|v\| := \max_{x \in \mathcal{X}} \|v_x\|_{\text{sp}}$.

Proof. Let $u, v \in (\mathbb{R}^M)^{\mathcal{X}}$. For $x \in \mathcal{X}$ with $N_x > 0$, by using the definition of $\widetilde{B}$ and applying triangle inequality,

$$\begin{aligned}\|(\widetilde{B}u)_x - (\widetilde{B}v)_x\|_{\text{sp}} &= \frac{1}{N_x}\left\|\sum_{(x',c)\in I_x}(B^c u_{x'} - B^c v_{x'})\right\|_{\text{sp}} \\ &\le \frac{1}{N_x}\sum_{(x',c)\in I_x}\|B^c u_{x'} - B^c v_{x'}\|_{\text{sp}}.\end{aligned}$$

Since B^c is non-expansive by Claim 8, each term is at most $\|u_{x'} - v_{x'}\|_{\text{sp}} \le \max_{z\in\mathcal{X}}\|u_z - v_z\|_{\text{sp}}$, so

$$\|(\widetilde{B}u)_x - (\widetilde{B}v)_x\|_{\text{sp}} \le \max_{z\in\mathcal{X}}\|u_z - v_z\|_{\text{sp}}.$$

For $x \in \mathcal{X}$ with $N_x = 0$, the same inequality holds trivially because $(\widetilde{B}u)_x = (\widetilde{B}v)_x = 0$. Therefore,

$$\|\widetilde{B}u - \widetilde{B}v\| = \max_{x\in\mathcal{X}}\|(\widetilde{B}u)_x - (\widetilde{B}v)_x\|_{\text{sp}} \le \max_{z\in\mathcal{X}}\|u_z - v_z\|_{\text{sp}} = \|u - v\|. \qquad \square$$

Claim 13. (Bounding the policy error). For every $u \in (\mathbb{R}^M)^{\mathcal{X}}$, conditional on observations $(I_x)_{x\in\mathcal{X}}$,

$$g(\pi_u) - g^* \le \sum_{x\in\mathcal{X}} p_x\|(\overline{B}u)_x - u_x\|_{\text{sp}} \le \|\widetilde{B}u - u\| + \sum_{x\in\mathcal{X}} p_x\|(\overline{B}u)_x - (\widetilde{B}u)_x\|_{\text{sp}}. \tag{12}$$

Proof. For ease of notation, let $v_x := (\bar{B}u)_x - u_x$. Let a be the trajectory induced by π_u, recursively defined by $a_t := \pi_u(a_{t-1}, c_t, x_t)$. The expected cost of π_u for the first T time steps is

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, a_t) + c_t(a_t)\right] &= \mathbb{E}\left[\sum_{t=1}^{T} d(a_{t-1}, a_t) + c_t(a_t) + u_{x_t}(a_t) - u_{x_t}(a_t)\right] \\ &= \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t}u_{x_t}(a_{t-1}) - u_{x_t}(a_t)\right] \\ &= \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t}u_{x_t}(a_{t-1}) - u_{x_{t-1}}(a_{t-1})\right] + \mathbb{E}[u_{x_0}(a_0) - u_{x_T}(a_T)] \\ &= \mathbb{E}\left[\sum_{t=1}^{T} (\bar{B}u)_{x_{t-1}}(a_{t-1}) - u_{x_{t-1}}(a_{t-1})\right] + \mathbb{E}[u_{x_0}(a_0) - u_{x_T}(a_T)] \\ &\le \mathbb{E}\left[\sum_{t=1}^{T} \max(v_{x_{t-1}})\right] + \mathbb{E}[u_{x_0}(a_0) - u_{x_T}(a_T)]\end{aligned}$$

Here, in the second equality we used the definitions of $B^{c_t}u$ and a_t, in the penultimate step we used the definition of $\bar{B}$, and in the last step we used the definition of v. Dividing by T and letting $T \to \infty$, the second term vanishes, and we obtain

$$g(\pi_u) \le \sum_{x\in\mathcal{X}} p_x \max(v_x). \tag{14}$$

Let o be the trajectory induced by an optimal policy π^*, recursively defined by $o_t := \pi^*(o_{t-1}, c_t, x_t)$. Similarly to the above, we show that the expected cost of policy π^* for the first T time steps is

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} d(o_{t-1}, o_t) + c_t(o_t)\right] &= \mathbb{E}\left[\sum_{t=1}^{T} d(o_{t-1}, o_t) + c_t(o_t) + u_{x_t}(o_t) - u_{x_t}(o_t)\right] \\
&\geq \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t} u_{x_t}(o_{t-1}) - u_{x_t}(o_t)\right] \\
&= \mathbb{E}\left[\sum_{t=1}^{T} B^{c_t} u_{x_t}(o_{t-1}) - u_{x_{t-1}}(o_{t-1})\right] + \mathbb{E}[u_{x_0}(o_0) - u_{x_T}(o_T)] \\
&= \mathbb{E}\left[\sum_{t=1}^{T} (\bar{B}u)_{x_{t-1}}(o_{t-1}) - u_{x_{t-1}}(o_{t-1})\right] + \mathbb{E}[u_{x_0}(o_0) - u_{x_T}(o_T)] \\
&\geq \mathbb{E}\left[\sum_{t=1}^{T} \min(v_{x_{t-1}})\right] + \mathbb{E}[u_{x_0}(o_0) - u_{x_T}(o_T)].
\end{aligned}$$

Dividing by T and letting $T \to \infty$, we obtain

$$g(\pi^*) \geq \sum_{x \in \mathcal{X}} p_x \min(v_x). \tag{15}$$

Putting (14) and (15) together, we obtain

$$\sum_{x \in \mathcal{X}} p_x \min(v_x) \leq g(\pi^*) \leq g(\pi_u) \leq \sum_{x \in \mathcal{X}} p_x \max(v_x).$$

By rearranging, applying triangle inequality, and using the definition of $\|\cdot\|$,

$$\begin{aligned}
g(\pi_u) - g^* \leq \sum_{x \in \mathcal{X}} p_x \|v_x\|_{\text{sp}} &= \sum_{x \in \mathcal{X}} p_x \|(\overline{B}u)_x - u_x\|_{\text{sp}} \\
&\leq \sum_{x \in \mathcal{X}} p_x \|(\widetilde{B}u)_x - u_x\|_{\text{sp}} + \sum_{x \in \mathcal{X}} p_x \|(\overline{B}u)_x - (\widetilde{B}u)_x\|_{\text{sp}} \\
&\leq \|\widetilde{B}u - u\| + \sum_{x \in \mathcal{X}} p_x \|(\overline{B}u)_x - (\widetilde{B}u)_x\|_{\text{sp}}. \qquad \square
\end{aligned}$$

A.2 Uniform estimation of the Bellman operator

We now control the error caused by replacing the expected Bellman operator with its empirical counterpart. For the remainder of this appendix, write

$$e_x(u) := \|(\overline{B}u)_x - (\widetilde{B}u)_x\|_{\text{sp}}.$$

By Lemma 14, it suffices to show that, with probability at least $1 - \delta$,

$$\sum_{x \in \mathcal{X}} p_x e_x(u) \leq \frac{\varepsilon}{2} \qquad \text{for every } u \in C.$$

Uniformity over C ensures that this bound applies to the data-dependent Halpern iterates, while weighting by p_x allows less accurate estimates at rarely visited states. We first prove the bound for the two oracle models, developing estimates that we will subsequently reuse for learning from a single trajectory.

A.2.1 Learning from oracle access

Recall the two types of oracles we consider: With *conditional oracles*, the learner chooses a state x and receives an independent pair $(x', c) \sim \mathcal{D}_x$. With a *stationary oracle*, each call independently returns a triple (x, x', c) by first drawing $x \sim p$, where p is the stationary distribution of the Markov chain, and then $(x', c) \sim \mathcal{D}_x$.

In both cases, we denote by N the total number of oracle calls, and by $(x_i, x_i', c_i)_{i=1}^N$ the list of triples obtained. Let

$$I_x := \{(x_i', c_i) : x_i = x\}, \qquad N_x := |I_x|.$$

Also, let $m := |\mathcal{X}|$, and set

$$\Gamma := nm \log\left(\left\lceil \frac{64D}{\varepsilon} \right\rceil + 1\right) + \log \frac{4nm}{\delta}.$$

Claim 15 (Uniform estimation for oracle samples)**.** Suppose that the training data are obtained either by taking a fixed number of independent samples from each conditional oracle, or by making a fixed number of independent stationary-oracle calls. With probability at least $1 - \delta/2$, simultaneously for every $u \in C$ and every x with $N_x > 0$,

$$e_x(u) \le D\sqrt{\frac{2\Gamma}{N_x}} + \frac{\varepsilon}{8}. \tag{16}$$

Proof. Let $H := ([0, D]^M)^{\mathcal{X}}$, and note that it suffices to prove the claim uniformly over $u \in H$. For $c\colon M \to \mathbb{R}_{\ge 0}$, $x' \in \mathcal{X}$, and $u \in H$, define

$$\Phi_{x',c}(u) := B^c u_{x'} - \min(B^c u_{x'}).$$

Each $B^c u_{x'}$ is 1-Lipschitz on M and hence has span at most D, so $0 \le \Phi_{x',c}(u)(s) \le D$.

Let $\eta := \varepsilon/64$, and let $\mathcal{N}$ be an η-net of H with respect to $\|\cdot\|_\infty$. This means that for each $u \in H$, there exists $w \in \mathcal{N}$ such that $|u - w|_\infty \le \eta$. We can choose $\mathcal{N}$ such that

$$|\mathcal{N}| \le \left(\left\lceil \frac{D}{\eta} \right\rceil + 1\right)^{nm} = \left(\left\lceil \frac{64D}{\varepsilon} \right\rceil + 1\right)^{nm}.$$

For the conditional oracles, the $x_1, \dots, x_N$ are fixed in advance by the learner. For the stationary oracle, condition on $x_1, \dots, x_N$. In either case, the counts N_x are fixed and, for each x, the pairs in I_x are independent samples from $\mathcal{D}_x$. For fixed $w \in \mathcal{N}$, $s \in M$, and $x \in \mathcal{X}$ with $N_x > 0$, Hoeffding's inequality gives, conditionally on $x_1, \dots, x_N$,

$$\mathbb{P}\left(\left|\frac{1}{N_x}\sum_{(x',c)\in I_x} \Phi_{x',c}(w)(s) - \mathbb{E}_{(x',c)\sim\mathcal{D}_x}[\Phi_{x',c}(w)(s)]\right| > D\sqrt{\frac{\Gamma}{2N_x}}\right) \le 2e^{-\Gamma}.$$

A union bound over such w, x, s gives a conditional failure probability at most

$$2|\mathcal{N}|mne^{-\Gamma} \le \frac{\delta}{2}.$$

This holds for every realization of $x_1, \dots, x_N$, so the same bound holds unconditionally.

On this event, fix $u \in H$ and choose $w \in \mathcal{N}$ with $\|u - w\|_\infty \le \eta$. We have

$$\|\Phi_{x',c}(u) - \Phi_{x',c}(w)\|_\infty \le 2\|u - w\|_\infty \le 2\eta, \tag{17}$$

and therefore

$$\left\|\frac{1}{N_x}\sum_{(x',c)\in I_x} \Phi_{x',c}(u) - \mathbb{E}_{(x',c)\sim\mathcal{D}_x}[\Phi_{x',c}(u)]\right\|_\infty \le D\sqrt{\frac{\Gamma}{2N_x}} + 4\eta.$$

Since $\|v\|_{\mathrm{sp}} \le 2\|v\|_\infty$, it follows that

$$\left\|\frac{1}{N_x}\sum_{(x',c)\in I_x} \Phi_{x',c}(u) - \mathbb{E}_{(x',c)\sim\mathcal{D}_x}[\Phi_{x',c}(u)]\right\|_{\mathrm{sp}} \le D\sqrt{\frac{2\Gamma}{N_x}} + 8\eta = D\sqrt{\frac{2\Gamma}{N_x}} + \frac{\varepsilon}{8}. \tag{18}$$

Note that $\frac{1}{N_x}\sum_{(x',c)\in I_x} \Phi_{x',c}(u)$ and $\mathbb{E}_{(x',c)\sim\mathcal{D}_x}[\Phi_{x',c}(u)]$ are equivalent to $(\widetilde{B}u)_x$ and $(\overline{B}u)_x$, respectively, up to an additive constant, and hence taking the span seminorm gives exactly $e_x(u)$. $\square$

The following lemma provides sufficient conditions for bounding $\sum_x p_x e_x(u)$. It does not make any assumption on how the observations $(I_x)_{x\in\mathcal{X}}$ are obtained.

Lemma 16 (Sufficient visit counts)**.** Let $A > 0$. Suppose that, simultaneously for every $u \in C$ and every x with $N_x > 0$,

$$e_x(u) \le D\sqrt{\frac{2A}{N_x}} + \frac{\varepsilon}{8}.$$

Set $\rho := \varepsilon/(16Dm)$, and suppose that

$$N_x \ge Np_x/4 \qquad \text{whenever } p_x > \rho. \tag{19}$$

If

$$N \ge \frac{128D^2mA}{\varepsilon^2}, \tag{20}$$

then $\sum_x p_x e_x(u) \le \varepsilon/2$ for all $u \in C$.

Proof. Fix $u \in C$. Note that $e_x(u) \le 2D$ for all $x \in \mathcal{X}$, because both $(\overline{B}u)_x$ and $(\widetilde{B}u)_x$ have span at most D. Therefore, the states with $p_x \le \rho$ contribute at most

$$\sum_{x:p_x\le\rho} p_x e_x(u) \le 2Dm\rho = \frac{\varepsilon}{8}.$$

For the remaining states, the estimation bound and (19) imply

$$\begin{aligned}\sum_{x:p_x>\rho} p_x e_x(u) &\le D\sqrt{2A}\sum_{x:p_x>\rho}\frac{p_x}{\sqrt{N_x}} + \frac{\varepsilon}{8}\sum_{x:p_x>\rho} p_x \\ &\le D\sqrt{\frac{8A}{N}}\sum_{x:p_x>\rho}\sqrt{p_x} + \frac{\varepsilon}{8} \\ &\le D\sqrt{\frac{8mA}{N}} + \frac{\varepsilon}{8},\end{aligned}$$

where the last step uses Cauchy–Schwarz and $\sum_x p_x = 1$. Together with (20), this gives

$$\sum_x p_x e_x(u) \le \frac{\varepsilon}{8} + \frac{\varepsilon}{8} + D\sqrt{\frac{8mA}{N}} \le \frac{\varepsilon}{2}. \qquad \square$$

We are now ready to prove Theorem 2, restated below.

Theorem 2. (Markov requests with oracle access). Let $\varepsilon \in (0, D)$, $\delta \in (0, 1)$. With either a conditional oracle or a stationary oracle, by using $O\left(\frac{D^2|\mathcal{X}|^2}{\varepsilon^2}\left(n\log\frac{D}{\varepsilon} + \log\frac{n|\mathcal{X}|}{\delta}\right)\right)$ oracle calls, with probability at least $1-\delta$ we find in polynomial time a policy with average cost at most $g^* + \varepsilon$ per step.

Proof. We treat the two types of oracles separately.

Conditional oracles. We make $N_x := O\left(\frac{D^2|\mathcal{X}|}{\varepsilon^2}\left(n\log\frac{D}{\varepsilon} + \log\frac{n|\mathcal{X}|}{\delta}\right)\right)$ calls to the conditional oracle for each state $x \in \mathcal{X}$. By Claim 15 and our choice of N_x, with probability at least $1 - \delta/2$,

$$e_x(u) \le D\sqrt{\frac{2\Gamma}{N_x}} + \frac{\varepsilon}{8} \le \frac{\varepsilon}{4} \qquad \text{for every } u \in C \text{ and } x \in \mathcal{X}.$$

Combining this with Lemma 14 concludes the proof.

Stationary oracle. Here N_x is binomial with parameters N and p_x, so the Chernoff bound yields

$$\mathbb{P}(N_x < Np_x/2) \le e^{-Np_x/8}.$$

Set $\rho := \varepsilon/(16Dm)$. A union bound shows that, with probability at least $1 - \delta/2$, $N_x \ge Np_x/2$ for every state with $p_x > \rho$, provided

$$N \ge \frac{8}{\rho}\log\frac{2m}{\delta} = \frac{128Dm}{\varepsilon}\log\frac{2m}{\delta}. \tag{21}$$

Our choice of N satisfies this because $D/\varepsilon \ge 1$ and $\Gamma \ge \log(2m/\delta)$. On this count event, (19) holds. Moreover, the definition of N ensures (20) with $A = \Gamma$. Combining Lemma 16 and Lemma 14 concludes the proof. $\square$

A.2.2 Learning from an observed trajectory

We consider the trajectory $x_0, (x_1, c_1), \ldots, (x_N, c_N)$ from an arbitrary initial state. Thus N_x counts visits to x at times $0, \ldots, N-1$.

Define

$$\Gamma_N := \Gamma + \log N = nm \log\left(\left\lceil \frac{64D}{\varepsilon} \right\rceil + 1\right) + \log \frac{4nmN}{\delta}. \tag{22}$$

Lemma 17 (Uniform estimation along a trajectory)**.** For N transitions from any initial Markov state, with probability at least $1 - \delta/2$, simultaneously for every $u \in C$ and every x with $N_x > 0$,

$$e_x(u) \le D\sqrt{\frac{2\Gamma_N}{N_x}} + \frac{\varepsilon}{8}.$$

Proof. We mostly reuse the analysis from Claim 15, with one caveat: conditioned on $x_1, \ldots, x_N$, the observations I_x are no longer independent samples from $\mathcal{D}_x$.

To handle this, we proceed as follows. For each $x \in \mathcal{X}$, generate an infinite independent sequence $Y_{x,1}, Y_{x,2}, \ldots \sim \mathcal{D}_x$, independently across states. When the current state is x, reveal the next unused element of this sequence. Its conditional distribution given the past is $\mathcal{D}_x$, so this generates the required Markov request process. The observed samples from x are exactly $Y_{x,1}, \ldots, Y_{x,N_x}$.

Apply the Hoeffding bound from the proof of Claim 15 to every prefix length $1 \le \ell \le N$ of the sequence $Y_{x,1}, Y_{x,2}, \ldots$, with Γ_N in place of Γ. For fixed $w \in \mathcal{N}$, x, s, and ℓ, the deviation threshold is $D\sqrt{\Gamma_N/(2\ell)}$ and the failure probability is at most $2e^{-\Gamma_N}$. The additional union bound over ℓ costs a factor N, which is exactly absorbed by $\Gamma_N = \Gamma + \log N$:

$$2|\mathcal{N}|mnNe^{-\Gamma_N} \le \frac{\delta}{2}.$$

The net-extension and translation arguments giving (16) apply unchanged. On a single event of probability at least $1 - \delta/2$, for every $u \in C$, $x \in \mathcal{X}$, and $1 \le \ell \le N$,

$$\left\| \frac{1}{\ell} \sum_{j=1}^{\ell} \Phi_{Y_{x,j}}(u) - \mathbb{E}_{Y \sim \mathcal{D}_x}[\Phi_Y(u)] \right\|_{\mathrm{sp}} \le D\sqrt{\frac{2\Gamma_N}{\ell}} + \frac{\varepsilon}{8}. \tag{23}$$

For each $x \in \mathcal{X}$ with $N_x > 0$, take $\ell = N_x$. The estimate holds simultaneously for every ℓ, so substituting the random N_x is valid without conditioning on it. The left-hand side equals $e_x(u)$, as in (16). □

Let

$$d(t) := \max_x \|P^t(x, \cdot) - p\|_{\mathrm{TV}},$$

where P denotes the transition matrix of the Markov chain and $\|\cdot\|_{\mathrm{TV}}$ denotes the total variation distance, defined by

$$\|\mu - \nu\|_{\mathrm{TV}} := \frac{1}{2} \sum_{x \in \mathcal{X}} |\mu_x - \nu_x|.$$

Then, the mixing time of the chain is defined by

$$\tau := \min\left\{ t \ge 1 : d(t) \le \frac{1}{8} \right\}. \tag{24}$$

We next establish the necessary visit-count bound.

Lemma 18 (Visit counts)**.** There is an absolute constant $K_0 > 0$ such that, for every $\rho \in (0, 1]$, $\delta \in (0, 1)$, and integer N satisfying

$$N \ge K_0 \frac{\tau}{\rho} \log \frac{8m}{\delta}, \tag{25}$$

with probability at least $1 - \delta/2$,

$$N_x = |\{0 \le t < N : x_t = x\}| \ge \frac{Np_x}{4} \qquad \text{for every } x \text{ with } p_x \ge \rho.$$

Proof. The standard mixing bound [LP17] gives $d(t) \le 2^{-\lfloor t/\tau \rfloor}$. Set $b := \tau\lceil \log_2(4/\delta)\rceil$, so that $d(b) \le \delta/4$, and let $R := N - b$. By maximal coupling [LP17, Proposition 4.7], $(x_b, \dots, x_{N-1})$ can be coupled to a stationary trajectory $(\widehat{x}_0, \dots, \widehat{x}_{R-1})$ so that

$$\mathbb{P}\left((x_b, \dots, x_{N-1}) = (\widehat{x}_0, \dots, \widehat{x}_{R-1})\right) \ge 1 - d(b) \ge 1 - \delta/4.$$

On coupling success,

$$N_x = \sum_{t=0}^{N-1} \mathbf{1}_{\{x_t = x\}} \ge \sum_{t=b}^{N-1} \mathbf{1}_{\{x_t = x\}} = \sum_{t=0}^{R-1} \mathbf{1}_{\{\widehat{x}_t = x\}}.$$

For a fixed state x, apply the lower-tail bound of Chung–Lam–Liu–Mitzenmacher [CLLM12, Theorem 3]. Since the auxiliary chain starts in stationarity, this gives

$$\mathbb{P}\left(\sum_{t=0}^{R-1} \mathbf{1}_{\{\widehat{x}_t = x\}} \le \frac{Rp_x}{2}\right) \le c_0 \exp\left(-\frac{Rp_x}{300\tau}\right),$$

where c_0 is an absolute constant.

We will choose N to ensure that $R \ge N/2$, and hence $Rp_x/2 \ge Np_x/4$. By union bound,

$$\mathbb{P}\left(\exists x : p_x \ge \rho,\ N_x < \frac{Np_x}{4}\right) \le \frac{\delta}{4} + c_0 m \exp\left(-\frac{N\rho}{600\tau}\right) \le \frac{\delta}{2}.$$

Setting $N \ge \max\{\lceil 600\frac{\tau}{\rho} \log \frac{4mc_0}{\delta}\rceil, 2b\}$ completes the proof. □

We are now ready to prove Theorem 3, restated below.

Theorem 3. (Markov requests with observed trajectory). Suppose the Markov chain $(x_t)_{t\ge 0}$ is aperiodic, and let $\varepsilon \in (0, D)$, $\delta \in (0, 1)$. By observing a trajectory of length $N = \widetilde{O}\Big(\frac{D^2|\mathcal{X}|^2 n}{\varepsilon^2} + \tau\frac{D|\mathcal{X}|}{\varepsilon}\Big)$, with probability at least $1 - \delta$ we find in polynomial time a policy with average cost at most $g^* + \varepsilon$ per step. Here, τ denotes the mixing time of the underlying Markov chain $(x_t)_t$, and $\widetilde{O}$ suppresses logarithmic factors in $n, |\mathcal{X}|, D, 1/\varepsilon, 1/\delta$, and $1 + \tau$.

Proof. Set

$$N := \left\lceil C_0 \left(\frac{D^2 m}{\varepsilon^2}\Gamma + \frac{\tau D m}{\varepsilon} \log \frac{8m}{\delta}\right)\right\rceil, \tag{26}$$

for a sufficiently large absolute constant C_0.

We verify the prerequisites needed to apply Lemma 16. The first term in the definition ensures that, for large enough C_0,

$$N \ge 128 \frac{D^2 m}{\varepsilon^2}(\Gamma + \log N),$$

which is precisely (20) with $A = \Gamma_N$.

By setting $\rho^{-1} := 16Dm/\varepsilon$, the second term in (26) ensures

$$N \ge K_0 \frac{\tau}{\rho} \log \frac{8m}{\delta},$$

which satisfies (25), for sufficiently large C_0.

Lemma 18 gives, with probability at least $1 - \delta/2$, $N_x \ge Np_x/4$ for all x with $p_x \ge \rho$. Lemma 17 gives the required uniform estimation bound, with $A = \Gamma_N$, with probability at least $1 - \delta/2$. Taking a union bound and applying Lemma 16 combined with Lemma 14 yields the conclusion. □

B Proof of Halpern residual bound (Theorem 7)

This section contains a proof of Theorem 7, restated here for convenience. The proof is very similar to the one given in [SS17].

Theorem 7. (Halpern residual bound). Let $F\colon Z \to Z$ be a non-expansive map defined on space Z with associated seminorm $\|\cdot\|$. Let $C \subseteq Z$ be a convex set, and let $\mathrm{diam}(C) := \sup_{z,z' \in C} \|z - z'\|$. Let $(\alpha_k)_{k\geq 1}$ be a step size sequence with $\alpha_k \in (0,1)$. The Halpern iteration [Hal67] with anchor at 0 and starting point $z_0 := 0$ produces a sequence $(z_k)_{k\geq 0}$ defined by

$$z_k := (1-\alpha_k)Fz_{k-1}, \qquad k \geq 1. \tag{7}$$

Suppose that $0 \in C$, and $z \in C$ implies $Fz \in C$. Then, for the choice

$$\alpha_k = \frac{2}{k+2}, \qquad k \geq 1,$$

we have

$$\|z_k - Fz_k\| \leq \frac{4 \cdot \mathrm{diam}(C)}{k+1}.$$

Proof. To bound $\|z_k - Fz_k\|$, we decompose it into

$$\|z_k - Fz_k\| \leq \|z_k - z_{k+1}\| + \|z_{k+1} - Fz_k\|. \tag{27}$$

The second term is easy to bound:

$$\begin{aligned} \|z_{k+1} - Fz_k\| &= \|(1-\alpha_{k+1})Fz_k - Fz_k\| \\ &= \alpha_{k+1}\|Fz_k\| \\ &\leq \alpha_{k+1}\mathrm{diam}(C) \\ &= \frac{2}{k+3}\mathrm{diam}(C). \end{aligned} \tag{28}$$

To bound the first term, we proceed as follows:

$$\begin{aligned} \|z_k - z_{k+1}\| &= \|(1-\alpha_k)Fz_{k-1} - (1-\alpha_{k+1})Fz_k\| \\ &\leq (1-\alpha_k)\|Fz_{k-1} - Fz_k\| + (\alpha_k - \alpha_{k+1})\|Fz_k\|. \end{aligned}$$

Since $Fz_k \in C$, we have $\|Fz_k\| \leq \mathrm{diam}(C)$. Since F is non-expansive,

$$\|Fz_{k-1} - Fz_k\| \leq \|z_{k-1} - z_k\|.$$

Putting these together, and denoting $\|z_k - z_{k+1}\|$ by l_k, we get

$$l_k \leq (1-\alpha_k)l_{k-1} + (\alpha_k - \alpha_{k+1})\mathrm{diam}(C).$$

Unfolding this, and noting that $l_0 \leq \mathrm{diam}(C)$, we get

$$l_k \leq \sum_{i=2}^{k+1} \left((1-\alpha_k)\dots(1-\alpha_i)\right)(\alpha_{i-1} - \alpha_i)\mathrm{diam}(C) + (1-\alpha_k)\dots(1-\alpha_1)\mathrm{diam}(C).$$

Substituting $\alpha_i = \frac{2}{i+2}$, we get that

$$(1-\alpha_k)\dots(1-\alpha_i) = \frac{k}{k+2}\dots\frac{i}{i+2} = \frac{i(i+1)}{(k+1)(k+2)}$$

and

$$\alpha_{i-1} - \alpha_i = \frac{2}{(i+1)(i+2)}.$$

Therefore,

$$l_k \leq \left(\sum_{i=2}^{k+1} \frac{i(i+1)}{(k+1)(k+2)} \cdot \frac{2}{(i+1)(i+2)} + \frac{2}{(k+1)(k+2)}\right)\mathrm{diam}(C)$$

We can bound each term by $\frac{2}{(k+1)(k+2)}$ to obtain

$$l_k \leq (k+2)\frac{2}{(k+1)(k+2)}\mathrm{diam}(C) = \frac{2}{k+1}\mathrm{diam}(C).$$

Using this together with (27) and (28), we get the desired result. □

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our theoretical results are stated informally in the abstract and formally in section 1.1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss potential limitations in section 6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Our proofs can be found in sections 3 and 4, and appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the details of our experimental setup in section 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have uploaded our code as part of the supplementary material.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide all the necessary details in section 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the standard error of the mean in our experimental results in section 5, and explain how it was computed. We also report the factors of variability.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide this information at the end of section 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Our work is theoretical, and while the techniques could inform practical decision-making systems, we do not foresee any specific direct societal impact that warrants discussion.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use the publicly available 2025 Citi Bike trip-history data [cit25], provided by Lyft Bikes and Scooters, LLC under the Citi Bike Data Sharing Policy. Our use of the data complies with these terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: We describe the usage of LLMs in the Acknowledgments and disclosures section.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.