



HAL
open science

Barely Random Algorithms and Collective Metrical Task Systems

Romain Cosson, Laurent Massoulié

► **To cite this version:**

Romain Cosson, Laurent Massoulié. Barely Random Algorithms and Collective Metrical Task Systems. Conference on Neural Information Processing Systems, Dec 2024, Vancouver (BC), Canada. hal-04894311

HAL Id: hal-04894311

<https://hal.science/hal-04894311v1>

Submitted on 17 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Barely Random Algorithms and Collective Metrical Task Systems

Romain Cosson

Inria, Paris

romain.cosson@inria.fr

Laurent Massoulié

Inria, Paris

laurent.massoulie@inria.fr

Abstract

We consider metrical task systems on general metric spaces with n points, and show that any fully randomized algorithm can be turned into a randomized algorithm that uses only $2 \log n$ random bits, and achieves the same competitive ratio up to a factor 2. This provides the first order-optimal *barely random* algorithms for metrical task systems, i.e., which use a number of random bits that does not depend on the number of requests addressed to the system. We discuss implications on various aspects of online decision making such as: distributed systems, advice complexity, and transaction costs, suggesting broad applicability. We put forward an equivalent view that we call *collective metrical task systems* where k agents in a metrical task system team up, and suffer the average cost paid by each agent. Our results imply that such a team can be $O(\log^2 n)$ -competitive as soon as $k \geq n^2$. In comparison, a single agent is always $\Omega(n)$ -competitive.

1 Introduction

Recent progress on the competitive analysis of important online problems, such as the k -server problem, metrical service systems, and metrical task systems have relied on the introduction of continuous optimization methods, such as the online primal-dual framework [18, 7, 38], and more recently, the online mirror descent framework [14–16, 5]. By design, these methods assume that the online algorithm is provided with an infinite number of random bits. In this paper, we address the question of whether this requirement is inherent to the online problem or specific to the methods at hand, by focusing on the influential example of metrical task systems [48, 11, 34, 15, 23].

To answer this question, we study the notion of barely random algorithm, introduced by [46] in the early days of competitive analysis of online algorithms: “We call an algorithm that uses a bounded number of random bits regardless of the number of requests *barely random*”. In the case of metrical task systems, we will observe that this notion is particularly fruitful, as it connects various aspects of online decision-making, such as collective systems, switching costs, and advice complexity.

Metrical Task Systems (MTS) is a central problem in online algorithms and online learning [45, 20, 29, 36, 19, 27] which recently attracted the interest of the community working on learning-augmented algorithms [1, 21, 3, 22, 4]. In essence, the problem is similar to the classical learning setting of ‘prediction with expert advice’ [43], with the key difference that there is a cost associated to switching between experts. We now describe the setting of metrical task systems [13] in a way that highlights the role played by the source of randomness.

Metrical Task Systems. The problem is defined on a metric space $\mathcal{X} = (X, d)$ where X is a finite set of cardinality $|X| = n$. The input is a sequence of cost vectors $\mathbf{c}(\cdot) = (\mathbf{c}(t))_{t \in \mathbb{N}} \in (\mathbb{R}_+^X)^\mathbb{N}$ and the output is a sequence of states $\rho(\cdot) = (\rho(t))_{t \in \mathbb{N}} \in X^\mathbb{N}$. The cost associated to each time $t \in \mathbb{N}$ is the sum of the movement cost $d(\rho(t-1), \rho(t))$ and the service cost $c_{\rho(t)}(t)$. The cost over all time

steps therefore writes as follows:

$$\text{Cost}(\rho(\cdot), \mathbf{c}(\cdot)) = \sum_{t \geq 1} d(\rho(t-1), \rho(t)) + c_{\rho(t)}(t).$$

The offline benchmark, denoted by $\text{OPT}(\mathbf{c}(\cdot))$, is defined as the smallest achievable cost over all possible sequences of states, i.e. $\text{OPT}(\mathbf{c}(\cdot)) = \inf_{\rho(\cdot) \in \mathcal{X}^{\mathbb{N}}} \text{Cost}(\rho(\cdot), \mathbf{c}(\cdot))$. An online algorithm \mathcal{A} is a method to define a time-consistent trajectory, i.e., such that the state at some time depends only on current and past information. It may use some source of randomness \mathbf{s} (seed), which is sampled from a known probability distribution $\mathbf{s} \sim \mathcal{D}$. Formally, the sequence of states $\rho_{\mathcal{A}}(\cdot)$ defined by algorithm \mathcal{A} can be computed by $\forall t : \rho_{\mathcal{A}}(t) = \mathcal{A}(\mathbf{c}(1), \dots, \mathbf{c}(t), \mathbf{s})$. The algorithm \mathcal{A} is said to be α -competitive, for some $\alpha \in \mathbb{R}$, if it satisfies $\mathbb{E}_{\mathbf{s} \sim \mathcal{D}}(\text{Cost}(\rho_{\mathcal{A}}(\cdot), \mathbf{c}(\cdot))) \leq \alpha \times \text{OPT}(\mathbf{c}(\cdot)) + \beta$ for some fixed $\beta \in \mathbb{R}$ and for any $\mathbf{c}(\cdot) \in (\mathbb{R}_+^X)^{\mathbb{N}}$. In this paper, we are interested in the way the competitive ratio scales with the source of randomness \mathcal{D} . When $\mathcal{D} = \mathcal{U}(\{0, 1\}^b)$, or equivalently when $\mathcal{D} = \mathcal{U}(\{1, \dots, 2^b\})$, we say that the algorithm is provided with b random bits. More generally, we consider the case where $\mathcal{D} = \mathcal{U}(\{1, \dots, k\})$ for arbitrary $k \in \mathbb{N}$. In one extreme, the case of $k = 1$ is known as the *deterministic* variant of the problem and has been well understood since the inception of metrical task systems. In the other extreme, the case of $k = +\infty$ (more formally, $\mathcal{D} = \mathcal{U}(\{0, 1\}^{\mathbb{N}})$) is known as the *randomized* variant of the problem and was solved more recently. Any other value of $k \in \mathbb{N}$ defines the *k-barely random* variant, which is the focus of this paper. For reasons that will now become apparent, we will also refer to this setting as the *k-collective* variant.

Collective algorithm for metrical task systems. Consider a team of $k \in \mathbb{N}$ agents that are collectively confronted to a metrical task system. We can denote the positions of the agents by $\rho_1(t), \dots, \rho_k(t) \in X^k$ and their trajectory is given by k deterministic online algorithms $\mathcal{A}_1, \dots, \mathcal{A}_k$. The participants collaborate in the sense that the cost paid by the team is the average cost paid by its members. It is immediate to observe that the *k-collective* setting of metrical task systems is exactly the same problem as the *k-barely random* setting defined above. This is because, given a *k-collective* strategy, one can define a *k-barely random* algorithm by using the seed $\mathbf{s} \sim \mathcal{U}(\{1, \dots, k\})$ to sample uniformly at random one strategy to imitate. In the rest of the paper, we will thus use the terms *collective* and *barely random* indistinctively. The collective presentation reflects a variety of real-world situations where a finite team of agents is confronted with an adversarial environment. A simple biological illustration is in foraging, e.g., when a colony of k ants is tasked to collectively gather a large amount of food (modeled by the service costs) from n locations while limiting the collective energy spent (modeled by the movement costs) [33, 37].

1.1 Main contributions

Our main technical result is the following,

Theorem 1.1 (Section 3). *For any metric space \mathcal{X} with n points, for any $\epsilon > 0$, for any integer $k \geq n^2/\epsilon$, if there exists a (fully) randomized MTS algorithm that is α -competitive, then there exists a *k-barely random* MTS algorithm that is $(1 + \epsilon)\alpha$ -competitive.*

The factor 2 announced in the abstract can be readily recovered from this theorem by letting $\epsilon = 1$, which we can assume in the rest of this section for ease of presentation. Theorem 1.1 relies on two techniques: a new equivalence between *barely random* algorithms and *barely fractional* strategies for metrical task systems that is presented in Section 2 ; and a new discrete first-order method that is developed in Section 3.

In light of the $\mathcal{O}(\log^2 n)$ -competitive randomized algorithm of Bubeck et al. [15] for metrical task systems, Theorem 1.1 has the following consequence:

Corollary 1.2 (of Theorem 1.1). *There is a $\mathcal{O}(\log^2 n)$ -competitive algorithm for MTS that requires $\lceil 2 \log n \rceil$ random bits on any n point metric.*

We provide in Section 4 a simple lower bound that almost matches this result. We also provide a tight guarantee in the case of the uniform metric space.

Proposition 1.3 (Section 4). *Any $\mathcal{O}(\log^2 n)$ -competitive algorithm for MTS on an n point metric requires at least $\log n - \mathcal{O}(\log \log n)$ random bits. In the uniform metric space, there is a $\mathcal{O}(\log n)$ -competitive algorithm that only requires $\lceil \log n \rceil$ random bits.*

Finally, we observe that the notion of barely random algorithm is closely connected to the notion of advice complexity. Advice complexity was introduced by [30] to measure the information content of online problems. In our context, one could ask the following question: What amount of information about the cost sequence suffices to improve over the $2n - 1$ lower bound on the deterministic competitive ratio of metrical task systems? Since bits of advice are at least as useful as random bits (see [39] for more details), we have:

Corollary 1.4 (of Theorem 1.1). *For any n point metric, there exists a deterministic MTS algorithm that is $\mathcal{O}(\log^2 n)$ -competitive, using only $\lceil 2 \log n \rceil$ bits of advice.*

This is a clear improvement over the result of [35], which requires advice of size that scales with the length of the cost sequence.

1.2 Other applications

Our method is helpful in any setting where acquiring random bits is costly. In concrete implementations and real-world systems, we also expect that barely fractional configurations (which are the discrete objects manipulated in this paper) are generally much more practical to handle than fully fractional distributions (which are arbitrary real numbers).

Our results also imply that a finite team of deterministic agents can be competitive in a way no single deterministic agent could be. This echoes the study of intelligence in multi-agent systems with concrete applications, such as dynamic power management [3]. Consider, for example, a computer with three power modes (*on*, *sleep*, *off*) with a switching cost of 1 between *on* and *sleep* and another switching cost of, say, 5 between *sleep* and *off*. At each time t the request is either that the computer is used and $c(t) = (1, +\infty, +\infty)$, i.e., the computer must be in state *on*; or the computer is not used and $c(t) = (1, 0.5, 0)$, i.e., the computer would rather be *off* than *sleep*, and *sleep* than *on*. Consider the variant of the problem, where the computer is, in fact, composed of $k = 9$ components (e.g., screen, CPU cores, etc.) that can each individually switch between one of the three modes (*on*, *sleep*, *off*). The energy spent by the computer is the sum of energy spent by its constituents. Since $n = 3$ and $k > n^2$, our results imply that such a system can enjoy the *randomized* competitive ratio, even if the cost sequence $c(\cdot)$ depends on the state of the system (*deterministic* adversary).

We also note that our work has a flavor that is similar to learning-augmented algorithms. Learning-augmented algorithms use predictions coming from a black-box algorithm in the hope of improving over the performance of competitive algorithms when predictions are accurate. In the same spirit, our method in Section 3 uses a fully fractional distribution coming from a black-box algorithm and tracks it via a proximal method.

1.3 Related works

Metrical task systems is a problem introduced by Borodin et al. (1992) [13]. This initial work resolved the deterministic competitive ratio, which is of $2n - 1$ for any metric space with n points; as well as the randomized competitive ratio for the uniform metric space, which is in $\Theta(\log n)$. The randomized competitive ratio for general n point metrics remained open until Bubeck et al. (2021) proposed a $\mathcal{O}(\log^2 n)$ -competitive algorithm and [17] obtained a matching lower-bound. These breakthroughs followed a long line of research (see e.g. [48, 11, 34, 23] and references therein).

Metrical task systems is deeply connected (both in results and methods) to the classical online setting of “learning with expert advice”, as observed by [10]. The differences are (1) there is a (movement) cost associated with switching from one expert to another (2) the offline benchmark is the best moving strategy as opposed to the best-fixed strategy (3) the goal is to obtain a multiplicative guarantee (competitive ratio) rather than an additive guarantee (regret).

The notion of barely random algorithms was introduced by [46] for the list update problem. In this paper, they present an algorithm that is 1.75-competitive for the list update problem, using exactly n random bits, where n is the length of the list. Such algorithms are further studied in the celebrated book of [12]. Barely random algorithms were later investigated for paging [39]. The notion also had some echo in the literature on scheduling (see [32] and references therein) but it was not applied before to metrical task systems or related problems.

Advice complexity was first studied in the context of metrical task systems in [31]. In this paper, the authors study the setting where b bits of advice are provided at each time-step to a deterministic metrical task systems algorithm for b of order $\log n$. The total amount of advice they require is, therefore, of order $B = T \log n$, where T is the number of time steps. One surprising consequence of our results (Corollary 1.4) is that we present a deterministic algorithm that has a competitive ratio of $\mathcal{O}(\log^2 n)$ using a single advice of total size $B = 2 \log n$, i.e. which is independent of the number of time-steps T . We note that advice complexity is connected to learning-augmented algorithms when the advice is untrusted [2].

Collective algorithms originate from the literature on distributed algorithms. For example, in the problem of collective tree exploration, a team of k agents is tasked to go through all edges of some unknown tree as fast as possible [35]. The connections observed in [25] between this problem and randomized algorithms for metrical service systems provided some inspiration for the present paper. We note that the interpretation of a probability distribution as a continuous swarm of infinitesimally small agents is not novel, see e.g. [6], and is directly connected to the so-called fractional formulation of metrical task systems. The novelty of our methods lies in the rigorous discretization of this fractional view, and leads to applications in collective and barely random algorithms. A surprising aspect of our method is that it does not rely on a tree-embedding of the metric space, but seems naturally adapted to the general setting, thanks to the very general properties coming from optimal transport, such as the Birkhoff-von Neuman theorem. The discretization also relies on a non-trivial (though concise) first-order method, which could be of interest for future works.

Finally, Section 3 of this paper is reminiscent of first-order optimization methods. Gradient descent of a function $f(\cdot)$ is also known as the explicit Euler method, where one can write its t -th iterate as the minimizer of the sum of a first-order approximation of $f(\cdot)$ at the $t - 1$ -th iterate and a quadratic cost. A closely related first-order method is the implicit Euler method where the t -th iterate is defined by,

$$\mathbf{x}(t) = \arg \min_{\mathbf{x}} f(\mathbf{x}) + \|\mathbf{x} - \mathbf{x}(t - 1)\|^2,$$

which is also known as the proximal operator [44]. Our Equation (1) is reminiscent of this operator. One key difference is that we replace the squared norm by an optimal transport cost, and that the offline objective function $f(\cdot)$ is replaced by an online potential, like in the work function algorithm [40]. Note that quite recently, the notion of Wasserstein proximal operator has appeared as a central object of study in the context of partial differential equations (see e.g. [47]). A key difference with our Equation (1) is that we consider the Wasserstein-1 metric (aka, the earthmover distance) and not the Wasserstein-2 metric.

1.4 Notations, definitions, and preliminaries

In the following, $\mathcal{X} = (X, d)$ is a finite metric space, i.e. X is a finite set and the distance d is positive, symmetric, and satisfies the triangle inequality. We denote by n the cardinality of X .

We denote by $\mathcal{P}(\mathcal{X})$ the set of all distributions on \mathcal{X} . For any two such distributions $\mathbf{x}, \mathbf{x}' \in \mathcal{P}(\mathcal{X})$, we call the optimal transport cost from \mathbf{x} to \mathbf{x}' and we denote by $\text{OT}(\mathbf{x}, \mathbf{x}')$ the quantity,

$$\text{OT}(\mathbf{x}, \mathbf{x}') = \inf_{\pi \in \Gamma(\mathbf{x}, \mathbf{x}')} \sum_{\rho, \rho'} \pi(\rho, \rho') d(\rho, \rho')$$

where $\Gamma(\mathbf{x}, \mathbf{x}') = \{\pi \in [0, 1]^{\mathcal{X} \times \mathcal{X}} : \forall \rho, \sum_{\rho' \in \mathcal{X}} \pi(\rho, \rho') = x_\rho \text{ and } \forall \rho', \sum_{\rho \in \mathcal{X}} \pi(\rho, \rho') = x'_{\rho'}\}$ is the set of couplings from \mathbf{x} to \mathbf{x}' . Classically, $\Gamma(\mathbf{x}, \mathbf{x}')$ can be viewed as the probability distributions on $\mathcal{X} \times \mathcal{X}$ with their marginals respectively equal to \mathbf{x} and \mathbf{x}' . The optimal transport cost (also known as the Wasserstein distance) defines a distance between probability distributions on \mathcal{X} . In particular, it satisfies positivity, symmetry, and the triangle inequality.

For some point $\rho \in \mathcal{X}$, we denote by $e_\rho \in \mathcal{P}(\mathcal{X})$ the probability distribution on \mathcal{X} that has all of its mass on ρ . Note that e_ρ can alternatively be seen as a vector of $\mathbb{R}^{\mathcal{X}}$. Denoting by \otimes the Kronecker product (aka the outer product) between vectors of $\mathbb{R}^{\mathcal{X}}$, we have, for any $\rho, \rho' \in \mathcal{X}$ that $e_\rho \otimes e_{\rho'}$ is an optimal coupling from e_ρ to $e_{\rho'}$.

For an arbitrary constant $k \in \mathbb{N}$ we denote by $\mathcal{P}_k(\mathcal{X})$ the set of all distributions on \mathcal{X} that only take their values in $\{0, 1/k, \dots, 1\}$. Such distributions will sometimes be called ‘configurations’ to

distinguish them from their continuous counterparts. For any two configurations $\mathbf{x}, \mathbf{x}' \in \mathcal{P}_k(\mathcal{X})$, there exists an associated optimal coupling π that only takes its values in $\{0, 1/k, \dots, 1\}$, i.e. which follows the discrete formulation of optimal transport by Monge. We call such a (discrete) coupling, an optimal transport plan. The existence of this optimal transport plan can be seen as a consequence of the Birkhoff-von Neumann theorem which states that the extreme points of the polytope of doubly stochastic matrices are permutation matrices [9]. Indeed a $k \times k$ doubly stochastic matrix naturally induces a coupling between two k -barely fractional configurations, with values in $\{0, 1/k, \dots, 1\}$ if it is a permutation matrix. The result then follows from the linearity of the objective.

For an integer $k \in \mathbb{N}$, we denote by $\mathcal{U}(\{1, \dots, k\})$ the uniform probability distribution over $\{1, \dots, k\}$. For an arbitrary distribution \mathcal{D} , we denote by $\mathbb{E}_{\mathbf{s} \sim \mathcal{D}}(\cdot)$ the expectation when the variable \mathbf{s} is sampled following \mathcal{D} . All logarithms, denoted by $\log(\cdot)$, are in base 2.

Paper outline Section 2 provides an equivalence between k -collective strategies and k -barely fractional strategies, which is used in Section 3 to prove Theorem 1.1. Section 4 provides additional discussions, including the lower-bound as well as refined guarantees for the uniform metric space. The conclusion highlights some open directions toward making further connections between online algorithms and distributed/collective systems.

2 Barely fractional strategies for metrical task systems

In the introduction, we gave the classical definition of metrical task systems [13], while highlighting the role played by the source of randomness \mathcal{D} . In the literature, a variant called the *fractional* formulation is known to be equivalent to the problem when the algorithm is allowed an infinite number of random bits (i.e. when $\mathcal{D} = \mathcal{U}(\{0, 1\}^{\mathbb{N}})$). We start by recalling this fractional formulation and the equivalence in Section 2.1. We then introduce the notion of *k -barely fractional* strategies and we show their relevance to the aforementioned k -barely random setting in Section 2.2. In contrast with previous results, this equivalence relies on the Birkhoff-von Neumann theorem.

2.1 (Fully) fractional strategies for metrical task systems

Fractional Metrical Task System. In the fractional variant of metrical task systems, at any instant t , a fractional strategy denoted by $\mathbf{x}(\cdot)$ maintains a distribution over the states that only depends on the information available before time t , i.e., $\mathbf{c}(\mathbf{c}(1), \dots, \mathbf{c}(t)) \in \mathcal{P}(\mathcal{X})$, which shall be denoted by $\mathbf{x}(t)$ hereafter. The cost paid by the strategy at time t is defined as the sum of a transport cost $\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$ and a service cost $\sum_{\rho \in \mathcal{X}} x_{\rho}(t) c_{\rho}(t)$. We denote by $\text{Cost}(\mathbf{x}(\cdot), \mathbf{c}(\cdot))$ the total cost associated to the strategy, i.e.

$$\text{Cost}(\mathbf{x}(\cdot), \mathbf{c}(\cdot)) = \sum_{t \geq 1} \text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) + \sum_{\rho \in \mathcal{X}} x_{\rho}(t) c_{\rho}(t).$$

The interest of this fractional formulation comes from the following reduction, which is classical in the literature (see, e.g., [15]).

Proposition 2.1. *For any fractional strategy for metrical task systems, there is a (fully) randomized algorithm that achieves the same cost and reciprocally.*

Proof. Consider a randomized algorithm $\mathcal{A}(\cdot, \mathbf{s})$ with $\mathbf{s} \sim \mathcal{U}(\{0, 1\}^{\mathbb{N}})$ for metrical task systems and define $\mathbf{x}(\cdot)$ at time t by setting $\forall \rho \in \mathcal{X} : x_{\rho}(t) = \mathbb{P}_{\mathbf{s}}(\mathcal{A}(\mathbf{c}(1), \dots, \mathbf{c}(t), \mathbf{s}) = \rho)$. It is clear from this definition that the service cost of the fractional strategy equals the expected service cost of the randomized algorithm. Also, observe that the movement cost of the fractional strategy, which equals $\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$ at time t , must be less than the expected movement cost of the randomized algorithm, which uses a (possibly sub-optimal) coupling between the two consecutive distributions.

Reciprocally, assume that we are given a fractional strategy $\mathbf{x}(\cdot) \in \mathcal{P}(\mathcal{X})^{\mathbb{N}}$. We can design a randomized algorithm $\mathcal{A}(\cdot, \mathbf{s})$ which relies on the random seed $\mathbf{s} \sim \mathcal{D}(\{0, 1\}^{\mathbb{N}})$. Observe that we can make the seemingly stronger (but equivalent, because $\{0, 1\}^{\mathbb{N}}$ is uncountable) assumption that $\mathcal{D} = \mathcal{U}([0, 1]^{\mathbb{N}})$. This reformulation allows to have access to one fresh (independent) real sampled uniformly at random from $[0, 1]$ at each time-step $t \in \mathbb{N}$, which we denote by $s(t) \in [0, 1]$. We assume that at time $t-1$, the algorithm is in some state $\mathcal{A}(\mathbf{c}(1), \dots, \mathbf{c}(t-1), \mathbf{s}) = \rho(t-1)$. At time

t , the algorithm considers the optimal transport plan π associated to $\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$ and samples the state $\rho(t)$ following the probability distribution proportional to $(\pi(\rho(t-1), \rho))_{\rho \in \mathcal{X}}$. This can be done by inversion sampling using only the random sample $s(t) \in \mathcal{U}([0, 1])$. It is clear by induction that the distribution of $\mathcal{A}(\mathbf{c}(1), \dots, \mathbf{c}(t), \mathbf{s}) = \rho(t)$ with this randomized algorithm follows exactly $\mathbf{x}(t)$. Further, its expected movement cost $\mathbb{E}_s(d(\rho(t-1), \rho(t)))$ is exactly equal (by definition) to the transport cost $\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$. \square

2.2 Barely fractional strategies for metrical task systems

Barely Fractional Metrical Task System. We now introduce a discretization of the fractional formulation of metrical task systems. This will define a k -barely fractional strategy $\mathbf{x}(\cdot)$. The definitions are the same as in the fully fractional variant above, except that at all times, the distribution $\mathbf{x}(t) \in \mathcal{P}(\mathcal{X})$ is constrained to belong to the set $\mathcal{P}_k(\mathcal{X})$ of distributions taking values in $\{0, 1/k, \dots, 1\}$ (see the notations section). We now show that this formulation enjoys an equivalence with the barely random variant of metrical task systems, when the source of randomness is limited to $\mathcal{D} = \mathcal{U}(\{1, \dots, k\})$.

Proposition 2.2. *For any k -barely fractional strategy for metrical task system, there is a k -barely random algorithm that achieves the same cost, and reciprocally.*

Proof. We assume that we are given a k -barely random algorithm \mathcal{A} and define as above the fractional strategy $\mathbf{x}(\cdot)$ at time t by $\forall \rho \in \mathcal{X} : x_\rho(t) = \mathbb{P}_s(\mathcal{A}(\mathbf{c}(1), \dots, \mathbf{c}(t), \mathbf{s}) = \rho)$. It remains true (see the argument of Section 2.1) that this fractional strategy has a smaller service and movement cost than the randomized algorithm. We also observe that since $\mathbf{s} \sim \mathcal{D} = \mathcal{U}(\{0, 1/k, \dots, 1\})$, we have $\mathbf{x}(t) \in \mathcal{P}_k(\mathcal{X})$. Therefore, $\mathbf{x}(\cdot)$ is a valid k -barely fractional strategy with a smaller cost than \mathcal{A} .

The reverse reduction uses an ingredient from the theory of optimal transport: the Birkhoff-von Neumann theorem, which states that the optimal transport between $\mathbf{x}, \mathbf{x}' \in \mathcal{P}_k(\mathcal{X})$ always admits an optimal coupling π taking its values in $\{0, 1/k, \dots, 1\}$, that we call an optimal transport plan (see notations and preliminaries section). Given a barely fractional strategy $\mathbf{x}(\cdot)$, we define a k -collective algorithm for which the population distribution will always match $\mathbf{x}(t) \in \mathcal{P}_k(\mathcal{X})$. The result then follows from the equivalence between collective and barely random algorithms discussed in the introduction. We initially distribute the k agents following $\mathbf{x}(0)$. Now, we assume as our induction hypothesis that the team at time $t-1$ is distributed exactly following $\mathbf{x}(t-1)$ and we show that we can deterministically redistribute the members of the team to follow $\mathbf{x}(t)$, while paying a moment cost of $\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$. Using the aforementioned Birkhoff-von Neumann theorem, there exists an optimal coupling between $\mathbf{x}(t-1) \in \mathcal{P}_k(\mathcal{X})$ and $\mathbf{x}(t) \in \mathcal{P}_k(\mathcal{X})$ that takes its values in $\{0, 1/k, \dots, 1\}$. Concretely, from the collective point of view, this means that we can move the agents from distribution $\mathbf{x}(t-1)$ to distribution $\mathbf{x}(t)$ using this coupling without having to split any agent. We use this coupling to deterministically reassign the agents to their new destination (we assume that agents have arbitrary identifiers and the ability to communicate in order to break the possible ties). Clearly, since the coupling is optimal, the average movement cost in the collective strategy will equal the transport cost of the fractional strategy, and since the population distribution of the agents is consistent with the barely fractional configuration, the service costs are equal. \square

Remark 2.3 (Fractional metrical task systems, with fixed transaction costs). *A straightforward illustration of the fractional variant of metrical task systems is in asset management. Consider an investor distributing its stakes on several assets via $\mathbf{x}(t) \in \mathcal{P}(\mathcal{X})$ where \mathcal{X} is a set of financial products such as stocks. The distance between two products is defined by the transaction cost associated with exchanging a unit value between two products (i.e., the liquidity of the products varies). The service cost corresponds to the cost of holding a poorly performing asset. The movement cost is a variable cost that is proportional to the mass that is exchanged at any round. Another type of cost that could naturally arise is a fixed transaction cost that the player should pay for converting any amount of stock ρ into stock ρ' . We propose to set this cost equal to $\tau d(\rho, \rho')$ for some real constant $\tau > 0$. In this setting, it becomes relevant for the player to perform transactions only if they involve a fractional mass greater than τ . A direct consequence of our results (Theorem 1.1 and Proposition 2.2) is that adding fixed transaction cost τ to the fractional variant of metrical task systems has little effect on its competitive ratio, provided that $\tau \leq 1/n^2$.*

3 Potential function method for barely fractional strategies

In this section, we provide the proof of the main result of the paper, Theorem 1.1. The core of our method is in Algorithm 1 which provides a way to transform a (black-box) fully fractional strategy $\mathbf{y}(\cdot)$ in a k -barely fractional strategy $\mathbf{x}(\cdot)$, while only losing a factor 2 in the competitive ratio provided that $k \geq n^2$ (it is then generalized to arbitrary factor $1+\epsilon$ in the proof below). Following the reductions from Section 2, this method also allows the transformation of a fully random algorithm into a k -barely random algorithm with similar competitive guarantees.

Note that the cost of $\mathbf{x}(\cdot)$ must be bounded by a constant times the cost of $\mathbf{y}(\cdot)$ if we hope to preserve the competitiveness of the latter. The informal reason why such a guarantee is difficult to obtain is that the adversary *knows exactly* the discretization method used by the team (by adversary, we mean the designer of the cost sequence $\mathbf{c}(\cdot)$, and by the team, we mean the swarm constituting $\mathbf{x}(\cdot)$). For instance, the naive rounding strategy which would be $\mathbf{x}(t) \in \arg \min_{\mathbf{x} \in \mathcal{P}_k(\mathcal{X})} \text{OT}(\mathbf{y}(t), \mathbf{x})$ does not preserve competitiveness. This is because infinitesimal movements of $\mathbf{y}(\cdot)$ could induce large (order $1/k$) movements of $\mathbf{x}(\cdot)$. A good discretization strategy must, therefore, display a hysteresis phenomenon in the sense that it does not undo a move right after that move was decided. This is precisely the idea behind our strategy in Equation (1) that employs a potential function $D(\cdot, \cdot)$ to ensure that $\mathbf{x}(t)$ remains sufficiently close to $\mathbf{y}(t)$ and an additional term in $\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$ to limit the movement cost of $\mathbf{x}(\cdot)$. We note that the technique presents some similarities with first-order optimization methods, which have been successful in several areas of online decision-making. One key novelty is that it is applied to a discrete space $\mathcal{P}_k(\mathcal{X})$.

Algorithm 1: Barely fractional strategy for $k \geq n^2$

Input: $\mathbf{x}(t-1)$: Previous barely fractional configuration

Input: $\mathbf{y}(t)$: Next fully fractional distribution

Output: $\mathbf{x}(t)$: Next barely fractional configuration

Solve for $\mathbf{x} \in \mathcal{P}_k(\mathcal{X})$:

$$\mathbf{x} \in \arg \min_{\mathbf{x} \in \mathcal{P}_k(\mathcal{X})} D(\mathbf{x}, \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t-1), \mathbf{x}), \quad (1)$$

where $D(\mathbf{x}, \mathbf{y}) = 2\text{OT}(\mathbf{x}/2 + 1/(2n)\mathbb{1}, \mathbf{y})$, breaking ties by maximizing $\text{OT}(\mathbf{x}(t-1), \mathbf{x})$.

Theorem 3.1. Consider an arbitrary n point metric \mathcal{X} , some real $\epsilon > 0$ and an integer $k \geq n^2/\epsilon$. For any fractional MTS strategy $\mathbf{y}(\cdot)$ on \mathcal{X} which is α -competitive, Algorithm 1 with potential defined in (2) provides a k -barely fractional MTS strategy $\mathbf{x}(\cdot)$ on \mathcal{X} which is $(1+\epsilon)\alpha$ -competitive.

Proof. We fix an integer $k \geq n^2/\epsilon$. Consider some α -competitive fractional strategy $\mathbf{y}(\cdot)$. We shall define (in an online manner) a $(1+\epsilon)\alpha$ -competitive k -barely fractional strategy $\mathbf{x}(\cdot)$, by performing an appropriate discretization of $\mathbf{y}(\cdot)$.

For any pair of distributions $\mathbf{x}, \mathbf{y} \in \mathcal{P}(\mathcal{X})$ we call the potential between \mathbf{x} and \mathbf{y} and denote by $D(\mathbf{x}, \mathbf{y})$ the quantity

$$D(\mathbf{x}, \mathbf{y}) = (1+\epsilon)\text{OT}\left(\frac{1}{1+\epsilon}\mathbf{x} + \frac{\epsilon}{n(1+\epsilon)}\mathbb{1}, \mathbf{y}\right), \quad (2)$$

where $\mathbb{1}$ denotes the vector of $\mathbb{R}^{\mathcal{X}}$ with all its values equal to 1. The correctness of this definition stems from the fact that we will always have $\mathbf{x}/(1+\epsilon) + \epsilon/n(1+\epsilon)\mathbb{1} \in \mathcal{P}(\mathcal{X})$, because $\mathbf{x} \in \mathcal{P}(\mathcal{X})$.

Upon the arrival of request $\mathbf{c}(t) \in \mathbb{R}_+^{\mathcal{X}}$, the (fully) fractional distribution $\mathbf{y}(t)$ is computed from a black-box algorithm and the barely fractional configuration $\mathbf{x}(t)$ is defined via Equation (1),

$$\mathbf{x}(t) \in \arg \min_{\mathbf{x} \in \mathcal{P}_k(\mathcal{X})} D(\mathbf{x}, \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t-1), \mathbf{x}),$$

where ties are broken in favor of the configuration \mathbf{x} which maximizes $\text{OT}(\mathbf{x}(t-1), \mathbf{x})$. We now aim to bound the movement and service costs of the aforesaid strategy $\mathbf{x}(\cdot)$ in terms of the movement and service costs of the black-box strategy $\mathbf{y}(\cdot)$.

Movement Costs. We start with the movement cost of $\mathbf{x}(\cdot)$. We decompose the change of the potential $P(t) = D(\mathbf{x}(t), \mathbf{y}(t))$ as follows,

$$\begin{aligned} P(t) - P(t-1) &= D(\mathbf{x}(t), \mathbf{y}(t)) - D(\mathbf{x}(t-1), \mathbf{y}(t-1)) \\ &= D(\mathbf{x}(t), \mathbf{y}(t)) - D(\mathbf{x}(t-1), \mathbf{y}(t)) + D(\mathbf{x}(t-1), \mathbf{y}(t)) - D(\mathbf{x}(t-1), \mathbf{y}(t-1)), \\ &\leq -\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) + (1+\epsilon)\text{OT}(\mathbf{y}(t-1), \mathbf{y}(t)), \end{aligned}$$

where the first inequality $D(\mathbf{x}(t), \mathbf{y}(t)) - D(\mathbf{x}(t-1), \mathbf{y}(t)) \leq -\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$ comes from the optimality of $\mathbf{x}(t)$ in (1) and the other inequality $D(\mathbf{x}(t-1), \mathbf{y}(t)) - D(\mathbf{x}(t-1), \mathbf{y}(t-1)) \leq (1+\epsilon)\text{OT}(\mathbf{y}(t-1), \mathbf{y}(t))$ is a direct application of the triangle inequality for optimal transport. We therefore have the at all times $t \geq 1$,

$$\text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) \leq P(t-1) - P(t) + (1+\epsilon)\text{OT}(\mathbf{y}(t-1), \mathbf{y}(t)). \quad (3)$$

Taking the (telescopic) sum with (3), we obtain the desired bound on the movement cost of $\mathbf{x}(\cdot)$,

$$\begin{aligned} \sum_{t \geq 1} \text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) &\leq \sum_{t \geq 1} P(t-1) - P(t) + (1+\epsilon)\text{OT}(\mathbf{y}(t-1), \mathbf{y}(t)) \\ &\leq P(0) + (1+\epsilon) \sum_{t \geq 1} \text{OT}(\mathbf{y}(t-1), \mathbf{y}(t)), \end{aligned}$$

where $P(0) = D(\mathbf{x}(0), \mathbf{y}(0))$ is a constant bounded by the diameter of \mathcal{X} .

Service costs. We shall now show that at all times $t \geq 1$,

$$\forall \rho \in \mathcal{X} : x_\rho(t) \leq (1+\epsilon)y_\rho(t). \quad (4)$$

This equation clearly implies that the service cost of $\mathbf{x}(\cdot)$ is at most $(1+\epsilon)$ that of $\mathbf{y}(\cdot)$. We shall use the following equation from Lemma 3.2 which states that at any time t , and for any configuration $\mathbf{x}' \in \mathcal{P}_k(\mathcal{X})$ such that $\mathbf{x}' \neq \mathbf{x}(t)$,

$$D(\mathbf{x}(t), \mathbf{y}(t)) < D(\mathbf{x}', \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t), \mathbf{x}'). \quad (5)$$

Assume by contradiction that (4) is not satisfied. For some $\rho \in \mathcal{X}$, we have $\frac{1}{1+\epsilon}x_\rho(t) + \frac{\epsilon}{n(1+\epsilon)} > y_\rho(t) + \frac{\epsilon}{n(1+\epsilon)}$. By applying Lemma 3.3 to the distributions $\mathbf{z} = \frac{1}{1+\epsilon}\mathbf{x}(t) + \frac{\epsilon}{n(1+\epsilon)}\mathbb{1}$ and $\mathbf{y}(t)$, with $\delta = \frac{1}{k(1+\epsilon)} \leq \frac{\epsilon}{n^2(1+\epsilon)}$, we obtain that there exists some $\rho' \in \mathcal{X}$ such that $\mathbf{z}' = \mathbf{z} + \delta(e_{\rho'} - e_\rho)$ and satisfying $\text{OT}(\mathbf{z}, \mathbf{y}(t)) = \text{OT}(\mathbf{z}, \mathbf{z}') + \text{OT}(\mathbf{z}', \mathbf{y}(t))$, where $\text{OT}(\mathbf{z}, \mathbf{z}') = \delta d(\rho, \rho')$. We then consider the configuration $\mathbf{x}' = \mathbf{x}(t) + \frac{1}{k}(e_{\rho'} - e_\rho)$, which will provide the contradiction by forming a violation of (5). First note that $\mathbf{x}' \in \mathcal{P}_k(\mathcal{X})$ because $x_\rho(t) \geq 1/k$ since $x_\rho(t) > (1+\epsilon)y_\rho(t) \geq 0$. Also, by definition $\mathbf{x}' \neq \mathbf{x}(t)$. Finally, $D(\mathbf{x}(t), \mathbf{y}(t)) = (1+\epsilon)\text{OT}(\mathbf{z}, \mathbf{y}(t)) = (1+\epsilon)\text{OT}(\mathbf{z}', \mathbf{y}(t)) + (1+\epsilon)\delta d(\rho, \rho') = D(\mathbf{x}', \mathbf{y}(t)) + \frac{1}{k}d(\rho, \rho') = D(\mathbf{x}', \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t), \mathbf{x}')$, which provides the contradiction.

Overall, the movement and service costs of $\mathbf{x}(\cdot)$ are bounded by $(1+\epsilon)$ those of $\mathbf{y}(\cdot)$ (plus a constant), thus, the α -competitiveness of $\mathbf{y}(\cdot)$ implies the $(1+\epsilon)\alpha$ -competitiveness of $\mathbf{x}(\cdot)$. \square

Lemma 3.2. *At any time t , the configuration $\mathbf{x}(t)$ that is selected by Equation (1) and the associated tie-breaking rule satisfies that for any $\mathbf{x}' \in \mathcal{P}_k(\mathcal{X})$ such that $\mathbf{x}' \neq \mathbf{x}(t)$,*

$$D(\mathbf{x}(t), \mathbf{y}(t)) < D(\mathbf{x}', \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t), \mathbf{x}').$$

Proof. Recall that $\mathbf{x}(t-1)$ is the configuration that immediately preceded $\mathbf{x}(t)$. Consider some arbitrary configuration $\mathbf{x}' \in \mathcal{P}_k(\mathcal{X})$. Since $\mathbf{x}(t)$ achieves the minimum of (1), we have

$$D(\mathbf{x}(t), \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) \leq D(\mathbf{x}', \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t-1), \mathbf{x}').$$

Using the triangle inequality of optimal transport, we also have

$$\text{OT}(\mathbf{x}(t-1), \mathbf{x}') \leq \text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) + \text{OT}(\mathbf{x}(t), \mathbf{x}').$$

The combination of both inequalities proves a non-strict version of Equation (5),

$$D(\mathbf{x}(t), \mathbf{y}(t)) \leq D(\mathbf{x}', \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t), \mathbf{x}'). \quad (6)$$

We now explain why this inequality is strict for any $\mathbf{x}' \neq \mathbf{x}(t)$. We note that equality can be met in (6) only if equality is also met in both of the aforementioned components of the inequality. We assume that is the case for some \mathbf{x}' and will now show that $\mathbf{x}' = \mathbf{x}(t)$. We have that,

$$\begin{cases} D(\mathbf{x}(t), \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) = D(\mathbf{x}', \mathbf{y}(t)) + \text{OT}(\mathbf{x}(t-1), \mathbf{x}'), \\ \text{OT}(\mathbf{x}(t-1), \mathbf{x}') = \text{OT}(\mathbf{x}(t-1), \mathbf{x}(t)) + \text{OT}(\mathbf{x}(t), \mathbf{x}'). \end{cases}$$

By the tie-breaking rule of (1), the first inequality implies that $\text{OT}(\mathbf{x}(t-1), \mathbf{x}') \leq \text{OT}(\mathbf{x}(t-1), \mathbf{x}(t))$ which allows to conclude, using the second equality that $\text{OT}(\mathbf{x}(t), \mathbf{x}') \leq 0$, and thus that $\mathbf{x}' = \mathbf{x}(t)$. This relies on the positivity of the optimal transport distance, which follows from the positivity of the distance in the original metric space \mathcal{X} . \square

Lemma 3.3. *Let $\mathbf{z} \in \mathcal{P}(\mathcal{X})$ and $\mathbf{y} \in \mathcal{P}(\mathcal{X})$ be two distributions on a metric space \mathcal{X} with n points. Assume that for some real $\delta > 0$ and some point $\rho \in \mathcal{X}$ one has $z_\rho \geq y_\rho + n\delta$. Then, there is another distribution $\mathbf{z}' \in \mathcal{P}(\mathcal{X})$, taking the form $\mathbf{z}' = \mathbf{z} + \delta(\mathbf{e}_{\rho'} - \mathbf{e}_\rho)$ for some other point $\rho' \in \mathcal{X}$, which satisfies $\text{OT}(\mathbf{z}, \mathbf{y}) = \text{OT}(\mathbf{z}, \mathbf{z}') + \text{OT}(\mathbf{z}', \mathbf{y})$, where $\text{OT}(\mathbf{z}, \mathbf{z}') = \delta d(\rho, \rho')$.*

Proof. By the triangle inequality of the transport distance, for any three distributions $\mathbf{z}, \mathbf{z}', \mathbf{y} \in \mathcal{P}(\mathcal{X})$, we have $\text{OT}(\mathbf{z}, \mathbf{y}) \leq \text{OT}(\mathbf{z}, \mathbf{z}') + \text{OT}(\mathbf{z}', \mathbf{y})$, therefore it will suffice to show the converse inequality.

We consider an optimal coupling from \mathbf{z} to \mathbf{y} denoted by $\pi = (\pi(\rho_1, \rho_2))_{\rho_1, \rho_2 \in \mathcal{X}^2}$. By definition, $z_\rho - y_\rho = \sum_{\rho_1 \in \mathcal{X}} \pi(\rho, \rho_1) - \pi(\rho_1, \rho) \geq n\delta$. In this sum of n terms, at least one term has to exceed δ . This allows to define one point $\rho' \in \mathcal{X}$ such that $\pi(\rho, \rho') \geq \delta$.

Next we consider the distribution $\mathbf{z}' = \mathbf{z} + \delta(\mathbf{e}_{\rho'} - \mathbf{e}_\rho)$. First, observe that this distribution is well-defined since $z_\rho \geq \delta$. Then, we note that $\pi' = \pi + \delta(\mathbf{e}_{\rho'} \otimes \mathbf{e}_\rho - \mathbf{e}_\rho \otimes \mathbf{e}_{\rho'})$ defines a coupling from \mathbf{z}' to \mathbf{y} and thus,

$$\text{OT}(\mathbf{z}', \mathbf{y}) \leq \sum_{\rho_1, \rho_2} d(\rho_1, \rho_2) \pi'(\rho_1, \rho_2) = \text{OT}(\mathbf{z}, \mathbf{y}) - \text{OT}(\mathbf{z}, \mathbf{z}'),$$

which finishes the proof. \square

4 Discussion, lower-bound and open directions

4.1 Uniform metric space

The uniform metric space is a notoriously simple special case of metrical task systems. It is the metric space where all pairs of nodes are at the same distance $\forall \rho, \rho' \in \mathcal{X}, \rho \neq \rho' \implies d(\rho, \rho') = 1$. The (fully) randomized competitive ratio of metrical task systems on the uniform metric space is known to lie between H_n and $2H_n$ [13], where H_n is the n -th harmonic number.

In this section, we show that $\lceil \log n \rceil$ bits of randomness are sufficient to achieve a constant factor optimal competitive ratio on the uniform metric space. The proof can be seen as a direct discretization of the algorithm of [13], and as an application of the ‘game of balls and urns’ in [26].

Proposition 4.1. *There is a n -barely random and $2H_n + 6$ -competitive algorithm for metrical task systems on the uniform metric space of cardinality n .*

Proof. Following [13], we decompose the time in discrete phases in order to obtain a lower bound on $\text{OPT}(c(\cdot))$. The 0-th phase starts at $t_0 = 0$ and for $i \in \mathbb{N}$, the i -th phase begins at time t_i and ends right before t_{i+1} which is defined as the first instant satisfying that $\forall \rho \in \mathcal{X} : \sum_{t_i \leq t < t_{i+1}} c_\rho(t) \geq 1$. Without loss of generality, we can assume for simplicity that there always exists a state ρ_i such that $\sum_{t_i \leq t < t_{i+1}} c_{\rho_i}(t) = 1$ (see the continuous-time variant of metrical task systems, as in [15], for a justification). We denote by i^* the total number of phases, which only depends on the cost sequence $c(\cdot)$. We observe that $\text{OPT}(c(\cdot)) \geq i^*$, because the cost incurred by any fixed trajectory is at least one during a given phase, and the cost of any mobile trajectory is also at least one.

The idea of [13] is to define a randomized algorithm that incurs a cost of at most $2H_n$ in each phase, and which is thus $2H_n$ -competitive. In a given phase i , for any $t \in [t_i, t_{i+1})$, the strategy consists in assigning $\mathbf{x}(t)$ to be the uniform probability distribution over the states in

$\{\rho \in \mathcal{X} : \sum_{t_i \leq \tau \leq t} c_\rho(\tau) \leq 1\}$, which are the non-saturated states. This strategy suffers a movement cost of at most $1 + 1/n + 1/(n-1) \cdots + 1/2 = H_n$, where H_n denotes the n -th harmonic number. Note that the service cost of this strategy is less or equal to its movement cost, so it is $2H_n$ -competitive. Unfortunately, the strategy is not n -barely fractional, because it takes values in $\{1/n, 1/(n-1), \dots, 1\}$ which are not all multiples of $1/n$.

We consider a simple discretization of this strategy, which does not suffer from the caveat described in Section 3. This discretization relies on the decomposition in phases which is specific to the uniform metric space. It can be seen as an application of the analysis of the ‘game of balls and urns’ which was used to design a collective tree exploration algorithm in [26]. We recall the definition of the game. The game starts with n balls, each located in one of n urns. In discrete rounds, an adversary picks an urn that contains a ball and the player has to move one ball in that urn to any other urn. The game ends when all urns have been chosen at least once by the adversary, and the cost of the player is the number of rounds before the end of the game, re-normalized by $1/n$. The simple strategy for the player that consists of relocating selected balls to the least loaded urn is known to induce a cost of at most $\ln(n) + 3 \leq H_n + 3$. Observe that this game models exactly the movement cost a corresponding n -barely fractional strategy in a given phase (where each ball represents a probability mass of $1/n$ and each urn represents a point of the uniform metric space). Also, since the service cost is (again) bounded by the movement cost in a given phase, the strategy is $2H_n + 6$ -competitive. \square

4.2 Lower-bound on the number of random bits

We now explain why $\Omega(\log n)$ random bits are required to obtain an asymptotically optimal competitive ratio. We show the stronger result that a k -barely random algorithm for metrical task systems is at most $(2n-1)/k$ -competitive. Thus, a $\mathcal{O}(\log^2 n)$ -competitive strategy must have $k \geq cst \times n/\log^2 n$ – corresponding to a lower-bound of $\log k \geq \log n - \mathcal{O}(\log \log n)$ in the number of random bits.

Proposition 4.2. *The competitive ratio of k -barely random algorithms for metrical task systems is at least $(2n-1)/k$.*

Proof. Consider a barely random algorithm $\mathcal{A}(\cdot, s)$, with $s \sim \mathcal{D} = \mathcal{U}(\{1, \dots, k\})$. The deterministic algorithm $\mathcal{A}(\cdot, 1)$ is sampled with probability $1/k$. By the $2n-1$ lower-bound of [13] on the competitive ratio of deterministic algorithms for metrical task systems, there exists a sequence $\mathbf{c}(\cdot) \in \mathbb{R}_+^N$ such that $\text{Cost}(\rho^{\mathcal{A}(\cdot, 1)}(\cdot), \mathbf{c}(\cdot)) \geq (2n-1)\text{OPT}(\mathbf{c}(\cdot))$. Integrating over the seed $s \sim \mathcal{U}(\{1, \dots, k\})$, we get $\mathbb{E}_s(\text{Cost}(\rho^{\mathcal{A}(\cdot, s)}(\cdot), \mathbf{c}(\cdot))) \geq \frac{2n-1}{k}\text{OPT}(\mathbf{c}(\cdot))$. \square

The question of whether an order-optimal randomized competitive ratio can be achieved with $k = O(n)$ instead of $k \geq n^2$ remains open on general metric spaces. This seems plausible for weighted star metrics, which admit a $\mathcal{O}(\log n)$ competitive strategies via an algorithm working in phases [18]. It is also possible that some metric spaces do require $k = \Omega(n^2)$. Demonstrating such a cutoff for barely random algorithms would nicely echo the cutoff between $\Theta(\log n)$ and $\Theta(\log^2 n)$ that seems to appear in the competitive ratio.

4.3 Conclusion and open directions for collective algorithms

The term ‘collective algorithm’ used in this paper is borrowed from the literature on distributed algorithms [35, 25]. We believe that highlighting situations in which a team of agents can be competitive/intelligent in a way that no single agent could be is an interesting research direction. As presented in this paper, such situations echo the well-studied competitive gap between randomized and deterministic algorithms for online problems [8]. But collective algorithms come with considerations that go beyond their analogy with barely-random algorithms. For instance, one can study the competitiveness of collective algorithms under: restricted communications (such as the write-read model of [35] or the local model of [28]), synchronous movements (as discussed in [24]), capacity constraints (where the amount of agents at a position is limited [6, 26]) or even with agents having some degree of selfishness or maliciousness (then involving a price of anarchy [41] for selfish agents, or some degree of robustness to byzantine faults [42] for malicious agents).

Acknowledgments The authors thank the anonymous reviewers for their helpful comments. RC thanks Spyros Angelopoulos, Nikhil Bansal, Moïse Blanchard, Christian Coester and Christoph Dürr for discussions. This work was supported by PRAIRIE ANR-19-P3IA-0001.

References

- [1] Keerti Anand, Rong Ge, Amit Kumar, and Debmalya Panigrahi. A regression approach to learning-augmented online algorithms. *Advances in Neural Information Processing Systems*, 34:30504–30517, 2021.
- [2] Spyros Angelopoulos, Christoph Dürr, Shendan Jin, Shahin Kamali, and Marc Renault. Online computation with untrusted advice. In *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2020.
- [3] Antonios Antoniadis, Christian Coester, Marek Eliás, Adam Polak, and Bertrand Simon. Learning-augmented dynamic power management with multiple states via new ski rental bounds. *Advances in Neural Information Processing Systems*, 34:16714–16726, 2021.
- [4] Antonios Antoniadis, Christian Coester, Marek Eliás, Adam Polak, and Bertrand Simon. Mixing predictions for online metric algorithms. In *International Conference on Machine Learning*, pages 969–983. PMLR, 2023.
- [5] Xingjian Bai, Christian Coester, and Romain Cosson. Unweighted layered graph traversal: Passing a crown via entropy maximization. *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2025.
- [6] Nikhil Bansal and Christian Coester. Online metric allocation and time-varying regularization. In *30th Annual European Symposium on Algorithms (ESA 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.
- [7] Nikhil Bansal, Niv Buchbinder, and Joseph Naor. A primal-dual randomized algorithm for weighted paging. *Journal of the ACM (JACM)*, 59(4):1–24, 2012.
- [8] Shai Ben-David, Allan Borodin, Richard Karp, Gabor Tardos, and Avi Wigderson. On the power of randomization in on-line algorithms. *Algorithmica*, 11:2–14, 1994.
- [9] Garrett Birkhoff. Tres observaciones sobre el algebra lineal. *Univ. Nac. Tucuman, Ser. A*, 5: 147–154, 1946.
- [10] Avrim Blum and Carl Burch. On-line learning and the metrical task system problem. In *Proceedings of the tenth annual conference on Computational learning theory*, pages 45–53, 1997.
- [11] Avrim Blum, Howard Karloff, Yuval Rabani, and Michael Saks. A decomposition theorem for task systems and bounds for randomized server problems. *SIAM Journal on Computing*, 30(5): 1624–1661, 2000.
- [12] Allan Borodin and Ran El-Yaniv. *Online computation and competitive analysis*. cambridge university press, 2005.
- [13] Allan Borodin, Nathan Linial, and Michael E Saks. An optimal on-line algorithm for metrical task system. *Journal of the ACM (JACM)*, 39(4):745–763, 1992.
- [14] Sébastien Bubeck, Michael B Cohen, Yin Tat Lee, James R Lee, and Aleksander Madry. K-server via multiscale entropic regularization. In *Proceedings of the 50th annual ACM SIGACT symposium on theory of computing*, pages 3–16, 2018.
- [15] 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.
- [16] Sébastien Bubeck, Christian Coester, and Yuval Rabani. Shortest paths without a map, but with an entropic regularizer. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1102–1113. IEEE, 2022.

- [17] 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*, pages 581–594, 2023.
- [18] Niv Buchbinder, Joseph Seffi Naor, et al. The design of competitive online algorithms via a primal–dual approach. *Foundations and Trends® in Theoretical Computer Science*, 3(2–3): 93–263, 2009.
- [19] Niv Buchbinder, Shahar Chen, Joseph Seffi Naor, and Ohad Shamir. Unified algorithms for online learning and competitive analysis. In *Conference on Learning Theory*, pages 5–1. JMLR Workshop and Conference Proceedings, 2012.
- [20] Lin Chen, Qian Yu, Hannah Lawrence, and Amin Karbasi. Minimax regret of switching-constrained online convex optimization: No phase transition. *Advances in Neural Information Processing Systems*, 33:3477–3486, 2020.
- [21] Jakub Chłędowski, Adam Polak, Bartosz Szabucki, and Konrad Tomasz Żoła. Robust learning-augmented caching: An experimental study. In *International Conference on Machine Learning*, pages 1920–1930. PMLR, 2021.
- [22] Nicolas Christianson, Junxuan Shen, and Adam Wierman. Optimal robustness-consistency tradeoffs for learning-augmented metrical task systems. In *International Conference on Artificial Intelligence and Statistics*, pages 9377–9399. PMLR, 2023.
- [23] Christian Coester and James R Lee. Pure entropic regularization for metrical task systems. In *Conference on Learning Theory*, pages 835–848. PMLR, 2019.
- [24] Romain Cosson. Breaking the $k/\log k$ barrier in collective tree exploration via tree-mining. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4264–4282. SIAM, 2024.
- [25] Romain Cosson and Laurent Massoulié. Collective tree exploration via potential function method. In *15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2024.
- [26] Romain Cosson, Laurent Massoulié, and Laurent Viennot. Efficient collaborative tree exploration with breadth-first depth-next. In *37th International Symposium on Distributed Computing (DISC 2023)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2023.
- [27] Amit Daniely and Yishay Mansour. Competitive ratio vs regret minimization: achieving the best of both worlds. In *Algorithmic Learning Theory*, pages 333–368. PMLR, 2019.
- [28] Dariusz Dereniowski, Yann Disser, Adrian Kosowski, Dominik Pajak, and Przemysław Uznański. Fast collaborative graph exploration. *Information and Computation*, 243:37–49, 2015.
- [29] Michael Dinitz, Sungjin Im, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Algorithms with prediction portfolios. *Advances in neural information processing systems*, 35: 20273–20286, 2022.
- [30] Stefan Dobrev, Rastislav Kráľovič, and Dana Pardubská. How much information about the future is needed? In *International Conference on Current Trends in Theory and Practice of Computer Science*, pages 247–258. Springer, 2008.
- [31] Yuval Emek, Pierre Fraigniaud, Amos Korman, and Adi Rosén. Online computation with advice. *Theoretical Computer Science*, 412(24):2642–2656, 2011.
- [32] Leah Epstein, John Noga, Steve Seiden, Jiří Sgall, and Gerhard Woeginger. Randomized online scheduling on two uniform machines. *Journal of Scheduling*, 4(2):71–92, 2001.
- [33] Ofer Feinerman, Amos Korman, Zvi Lotker, and Jean-Sébastien Sereni. Collaborative search on the plane without communication. In *Proceedings of the 2012 ACM symposium on Principles of distributed computing*, pages 77–86, 2012.

- [34] Amos Fiat and Manor Mendel. Better algorithms for unfair metrical task systems and applications. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 725–734, 2000.
- [35] Pierre Fraigniaud, Leszek Gasieniec, Dariusz R Kowalski, and Andrzej Pelc. Collective tree exploration. *Networks: An International Journal*, 48(3):166–177, 2006.
- [36] Gautam Goel, Yiheng Lin, Haoyuan Sun, and Adam Wierman. Beyond online balanced descent: An optimal algorithm for smoothed online optimization. *Advances in Neural Information Processing Systems*, 32, 2019.
- [37] Briec Guinard and Amos Korman. Intermittent inverse-square lévy walks are optimal for finding targets of all sizes. *Science advances*, 7(15):eabe8211, 2021.
- [38] Enikő Kevi and Kim Tháng Nguyen. Primal-dual algorithms with predictions for online bounded allocation and ad-auctions problems. In *International Conference on Algorithmic Learning Theory*, pages 891–908. PMLR, 2023.
- [39] Dennis Komm and Richard Královič. Advice complexity and barely random algorithms. *RAIRO-Theoretical Informatics and Applications*, 45(2):249–267, 2011.
- [40] Elias Koutsoupias. The k-server problem. *Computer Science Review*, 3(2):105–118, 2009.
- [41] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. In *Annual symposium on theoretical aspects of computer science*, pages 404–413. Springer, 1999.
- [42] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, 1982.
- [43] Nick Littlestone and Manfred K Warmuth. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994.
- [44] Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239, 2014.
- [45] Shyam Sundhar Ramesh, Pier Giuseppe Sessa, Andreas Krause, and Ilija Bogunovic. Movement penalized bayesian optimization with application to wind energy systems. *Advances in Neural Information Processing Systems*, 35:27036–27048, 2022.
- [46] Nick Reingold, Jeffery Westbrook, and Daniel D Sleator. Randomized competitive algorithms for the list update problem. *Algorithmica*, 11(1):15–32, 1994.
- [47] Filippo Santambrogio. {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 7:87–154, 2017.
- [48] Steve Seiden. Unfair problems and randomized algorithms for metrical task systems. *Information and Computation*, 148(2):219–240, 1999.