I N F O R M A T I K

Topology Matters: Smoothed Competitiveness of Metrical Task Systems

Guido Schäfer Naveen Sivadasan

MPI–I–2003–1–016 December 2003

FORSCHUNGSBERICHT RESEARCH REPORT

M A X - P L A N C K - I N S T I T U T
F Ü R
I N F O R M A T I K

Stuhlsatzenhausweg 85 66123 Saarbrücken Germany

Authors' Addresses

Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85, 66123 Saarbrücken, Germany
Email: {schaefer,ns}@mpi-sb.mpg.de

Publication Notes

An extended abstract of this paper was accepted for publication in the proceedings of the 21st Symposium on Theoretical Aspects of Computer Science (STACS 2004).

Acknowledgements

The first author was supported by DFG graduated studies program "Quality Guarantees for Computer Systems" (GK 623), Saarbrücken. This work was partially supported by the Future and Emerging Technologies programme of the EU under contract number IST-1999-14186 (ALCOM-FT).

Abstract

We consider online problems that can be modeled as *metrical task systems*: An online algorithm resides in a graph G of n nodes and may move in this graph at a cost equal to the distance. The algorithm has to service a sequence of *tasks* that arrive online; each task specifies for each node a *request cost* that is incurred if the algorithm services the task in this particular node. The objective is to minimize the total request cost plus the total travel cost. Several important online problems can be modeled as metrical task systems. Borodin, Linial and Saks [6] presented a deterministic *work function algorithm* (WFA) for metrical task systems having a tight competitive ratio of $2n - 1$.

In this chapter, we present a *smoothed competitive analysis* of WFA. Given an adversarial task sequence, we smoothen the request costs by means of a symmetric additive smoothing model and analyze the competitive ratio of WFA on the smoothed task sequence. Our analysis reveals that the smoothed competitive ratio of WFA is much better than $O(n)$ and that it depends on several topological parameters of the underlying graph G, such as the minimum edge length $U_{\min}$, the maximum degree D, and the edge diameter *diam*. Assuming that the ratio between the maximum and the minimum edge length of G is bounded by a constant, the smoothed competitive ratio of WFA becomes $O(diam(U_{\min}/\sigma + \log(D)))$ and $O(\sqrt{n \cdot (U_{\min}/\sigma + \log(D))})$, where σ denotes the standard deviation of the smoothing distribution. For example, already for perturbations with $\sigma = \Theta(U_{\min})$ the competitive ratio reduces to $O(\log n)$ on a clique and to $O(\sqrt{n})$ on a line. We also prove that for a large class of graphs these bounds are asymptotically tight. Furthermore, we provide two lower bounds for any arbitrary graph. We obtain a better bound of $O(\beta \cdot (U_{\min}/\sigma + \log(D)))$ on the smoothed competitive ratio of WFA if each adversarial task contains at most β non-zero entries. Our analysis holds for various probability distributions, including the uniform and the normal distribution.

We also provide the first average case analysis of WFA. We prove that WFA has $O(\log(D))$ expected competitive ratio if the request costs are chosen randomly from an arbitrary non-increasing distribution with standard deviation $\sigma = \Theta(U_{\min})$.

1 Introduction

Metrical Task Systems. Borodin, Linial and Saks [6] introduced a general framework to model online problems, called *metrical task systems*. Many online problems can be formulated as metrical task systems; for example, the paging problem, the static list accessing problem and the k-server problem. Due to its generality, the competitive ratio of an algorithm for metrical task systems is usually weak compared to the one of an online algorithm that is designed for a particular problem, such as the k-server problem.

Metrical task systems are formulated as follows. We are given an undirected and connected graph $G := (V, E)$, with node set V and edge set E, and a positive length function $\lambda : E \to \mathbb{R}^+$ on the edges of G. We extend λ to a metric δ on G. Let $\delta : V \times V \to \mathbb{R}^+$ be a distance function such that $\delta(u, v)$ denotes the shortest path distance (with respect to λ) between any two nodes u and v in G. A *task* τ is an n-vector $(r(v_1), \ldots, r(v_n))$ of *request costs*. The cost to process task τ in node v_i is $r(v_i) \in \mathbb{R}^+ \cup \{\infty\}$. The online algorithm starts from a given initial position $s_0 \in V$ and has to service a sequence $\mathcal{S} := \langle \tau_1, \ldots, \tau_r \rangle$ of tasks, arriving one at a time. If the online algorithm resides after task τ_{t-1} in node u, the cost to service task τ_t in node v is $\delta(u, v) + r_t(v)$; $\delta(u, v)$ is the *transition cost* and $r_t(v)$ is the *processing cost*. The objective is to minimize the total transition plus processing cost.

Competitive Analysis. A measurement for the quality of an online algorithm is its *competitive ratio* [10]. For a cost minimization problem the competitive ratio is defined as the supremum over all input instances of the ratio between the cost of the online algorithm and the cost of an optimal offline algorithm.

Borodin, Linial and Saks [6] gave a deterministic online algorithm, known as the *work function algorithm* (WFA), for metrical task systems. WFA has a competitive ratio of $2n - 1$, which is optimal. Borodin, Linial and Saks [6] and Manasse, McGeoch and Sleator [8] proved that *every* deterministic online algorithm for metrical task systems has competitive ratio at least $2n - 1$. That is, the performance of WFA (and any other deterministic online algorithm) can be made arbitrarily bad. On the other hand, the competitive ratio often is an over-pessimistic estimation of the performance of an online algorithm, since instances that force the online algorithm in its worst case behavior are highly artificial.

Smoothed Competitive Analysis. Spielman and Teng [11] proposed a new complexity measure, called *smoothed complexity*, which is a hybrid between average case and worst case complexity. The basic idea is to randomly perturb, or *smoothen*, the input instances and to analyze the performance of the algorithm on the perturbed instances. The smoothed complexity of an algorithm is defined as the supremum over all input instances of the expected running time on the perturbed instances. Intuitively, the smoothed complexity of an algorithm is small if the worst case instances are isolated peaks in the instance/time space.

Based on the idea underlying *smoothed analysis* [11], Becchetti et al. [3] recently proposed *smoothed competitive analysis* as an alternative to worst case competitive analysis of online algorithms. The idea is to randomly perturb, or *smoothen*, an adversarial input instance $\check{\mathcal{S}}$ and to analyze the performance of the algorithm on the perturbed instances.

Upper Bounds		
random tasks	$O\big(\frac{\sigma}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)\big)$	
arbitrary tasks	$O\big(\frac{Diam}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)\big)$ and	$O\left(\sqrt{n\cdot\frac{U_{\max}}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)}\right)$
β-elementary tasks	$O\big(\beta\cdot\frac{U_{\max}}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)\big)$	

Table 1: Upper bounds on the smoothed competitive ratio of WFA.

Let $\text{ALG}[\mathcal{S}]$ and $\text{OPT}[\mathcal{S}]$, respectively, be the cost of the online and the optimal offline algorithm on a smoothed instance $\mathcal{S}$ obtained from $\check{\mathcal{S}}$. The *smoothed competitive ratio* c of ALG with respect to a smoothing distribution f is defined as

$$c := \sup_{\check{\mathcal{S}}} \mathbf{E}_{\mathcal{S}\xleftarrow{f}\check{\mathcal{S}}}\left[\frac{\text{ALG}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\right].$$

Our Contribution. We use the notion of smoothed competitiveness to characterize the asymptotic performance of WFA. We smoothen the request costs of each task according to an additive symmetric smoothing model. Each cost entry is smoothed by adding a random number chosen from a probability distribution f, whose expectation coincides with the original cost entry. Our analysis holds for various probability distributions, including the uniform and the normal distribution. We use σ to refer to the standard deviation of f. Our analysis reveals that the smoothed competitive ratio of WFA is much better than its worst case competitive ratio suggests and that it depends on certain *topological parameters* of the underlying graph.

Definition of Topological Parameters: Throughout this chapter, we assume that the underlying graph G has n nodes, minimum edge length $U_{\min}$, maximum edge length $U_{\max}$, and maximum degree D. Furthermore, we use *Diam* to refer to the *diameter* of G, i.e., the maximum length of a shortest path between any two nodes. Similarly, a graph has *edge* diameter *diam* if any two nodes are connected by a path of at most *diam* edges. Observe that $diam U_{\min} \leq Diam \leq diam U_{\max}$. We emphasize that these topological parameters are defined with respect to G and its length function λ—not with respect to the resulting metric.

We prove several upper bounds; see Table 1.

1. We show that if the request costs are chosen randomly from a distribution f, which is non-increasing in $[0,\infty)$, the expected competitive ratio of WFA is

$$O\big(1+\tfrac{\sigma}{U_{\min}}\cdot\log(D)\big).$$

   In particular, WFA has an expected competitive ratio of $O(\log(D))$ if $\sigma=\Theta(U_{\min})$. For example, we obtain a competitive ratio of $O(\log(n))$ on a clique and of $O(1)$ on a binary tree.

Lower Bounds			
arbitrary tasks			
– existential	$\Omega\big(\frac{Diam}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)\big)$	and	$\Omega\Big(\sqrt{n\cdot\frac{U_{\max}}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)}\Big)$
– universal	$\Omega\big(\frac{U_{\min}}{\sigma}+\frac{U_{\min}}{U_{\max}}\log(D)\big)$	and	$\Omega\Big(\sqrt{diam\cdot\frac{U_{\min}}{U_{\max}}\big(\frac{U_{\min}}{\sigma}+1\big)}\Big)$
β-elementary tasks	$\Omega\big(\beta\cdot\big(\frac{U_{\min}}{\sigma}+1\big)\big)$ (existential)		

Table 2: Lower bounds on the smoothed competitive ratio of any deterministic online algorithm.

2. We prove two upper bounds on the smoothed competitive ratio of WFA:

$$O\Big(\frac{Diam}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)\Big) \quad \text{and} \quad O\Big(\sqrt{n\cdot\frac{U_{\max}}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)}\Big).$$

For example, if $\sigma=\Theta(U_{\min})$ and $U_{\max}/U_{\min}=\Theta(1)$, WFA has smoothed competitive ratio $O(\log(n))$ on any graph with constant edge diameter and $O(\sqrt{n})$ on any graph with constant maximum degree. Note that we obtain an $O(\log(n))$ bound on a complete binary tree.

3. We obtain a better upper bound on the smoothed competitive ratio of WFA if the adversarial task sequence only consists of *β-elementary tasks*. A task is β-elementary if it has at most β non-zero entries. (We will use the term *elementary task* to refer to a 1-elementary task.) We prove a smoothed competitive ratio of

$$O\big(\beta\cdot\frac{U_{\max}}{U_{\min}}\big(\frac{U_{\min}}{\sigma}+\log(D)\big)\big).$$

For example, if $\sigma=\Theta(U_{\min})$ and $U_{\max}/U_{\min}=\Theta(1)$, WFA has smoothed competitive ratio $O(\beta\log(D))$ for β-elementary tasks.

We also present lower bounds; see Table 2. All our lower bounds hold for *any* deterministic online algorithm and if the request costs are smoothed according to the additive symmetric smoothing model. We distinguish between *existential* and *universal* lower bounds. An existential lower bound, say $\Omega(f(n))$, means that there *exists* a class of graphs such that *every* deterministic algorithm has smoothed competitive ratio $\Omega(f(n))$ on these graphs. On the other hand, a universal lower bound $\Omega(f(n))$ states that for *any arbitrary* graph, *every* deterministic algorithm has smoothed competitive ratio $\Omega(f(n))$. Clearly, for metrical task systems, the best lower bound we can hope to obtain is $\Omega(n)$. Therefore, if we state a lower bound of $\Omega(f(n))$, we actually mean $\Omega(\min\{n,f(n)\})$.

4. For a large range of values for *Diam* and D, we present existential lower bounds that are asymptotically tight to the upper bounds stated in 2. This means (a) that the stated smoothed competitive ratio of WFA is asymptotically tight, and (b) that WFA is asymptotically optimal under the additive smoothing model—no other deterministic algorithm can achieve a better smoothed competitive ratio.

5. We also prove two universal lower bounds on the smoothed competitive ratio:

$$\Omega\left(\tfrac{U_{\min}}{\sigma} + \tfrac{U_{\min}}{U_{\max}} \log(D)\right) \quad \text{and} \quad \Omega\left(\min\left\{ diam, \sqrt{diam \cdot \tfrac{U_{\min}}{U_{\max}} \left(\tfrac{U_{\min}}{\sigma} + 1\right)} \right\}\right).$$

   Assume that $U_{\max}/U_{\min} = \Theta(1)$. Then, the first bound matches the first upper bound stated in 2 if the edge diameter *diam* is constant, e.g., for a clique. The second bound matches the second upper bound in 2 if $diam = \Omega(n)$ and the maximum degree D is constant, e.g., for a line.

6. For β-elementary tasks, we prove an existential lower bound of

$$\Omega\left(\beta \cdot \left(\tfrac{U_{\min}}{\sigma} + 1\right)\right).$$

   This implies that the bound in 3 is tight up to a factor of $(U_{\max}/U_{\min}) \log(D)$.

Constrained Balls into Bins Game. Our analysis crucially relies on a lower bound on the cost of an optimal offline algorithm. We therefore study the growth of the work function values on a sequence of random requests. It turns out that the increase in the work function values can be modeled by a version of a balls into bins game with dependencies between the heights of the bins, which are specified by a constraint graph. We call this game the *constrained balls into bins game.* The dependencies between the heights of the bins make it difficult to analyze this stochastic process. We believe that the constrained balls into bins game is also interesting independently of the context of this work.

Related Work. Several other attempts were made in the past to overcome the over-pessimistic estimation of the performance of an online algorithm by its competitive ratio. One idea was to enhance the capability of the online algorithm by allowing a limited lookahead [1, 2]. Another idea was to restrict the power of the adversary. For example, Borodin et al. [5] used an access graph model to restrict the input sequences in online paging problems to specific patterns. The diffuse adversary model by Koutsoupias and Papadimitriou [7] is another attempt to refine the notion of competitiveness. In this model, the actual distribution of the input is chosen by an adversary from a known class of possible distributions. We strongly believe that smoothed competitive analysis is a natural alternative to adequately characterize the performance guarantee of an online algorithm.

Organization of this Paper. In Section 2, we first review the work function algorithm and state some of its properties. In Section 3, we define the smoothing model that we use. The lower bound on the cost of an optimal offline algorithm and the related balls into bins game are presented in Section 4. Then, in Section 5 and Section 6, we prove the upper bounds on the smoothed competitive ratio of WFA. After that, in Section 8 we present an upper bound on the expected competitive ratio of WFA, and in Section 9 we develop the bound for β-elementary tasks. Finally, in Section 10 we prove existential and universal lower bounds. We give some concluding remarks in Section 11.

2 Work Function Algorithm

Let $\mathcal{S} = \langle \tau_1, \ldots, \tau_\ell \rangle$ be a request sequence and let $s_0 \in V$ denote the initial position. Let $\mathcal{S}_t$ denote the subsequence of the first t tasks of $\mathcal{S}$. For each t, $0 \leq t \leq \ell$, we define a function $w_t : V \to \mathbb{R}$ such that for each node $u \in V$, $w_t(u)$ is the minimum offline cost to process $\mathcal{S}_t$ starting in s_0 and ending in u. The function w_t is called the *work function* at time t with respect to $\mathcal{S}$ and s_0.

Let OPT denote an optimal offline algorithm. Clearly, the optimal offline cost OPT$[\mathcal{S}]$ on $\mathcal{S}$ is equal to the minimum work function value at time ℓ, i.e., OPT$[\mathcal{S}] = \min_{u \in V}\{w_\ell(u)\}$. We can compute $w_t(u)$ for each $u \in V$ by dynamic programming:

$$w_0(u) := \delta(s_0, u), \quad \text{and} \quad w_t(u) := \min_{v \in V}\{w_{t-1}(v) + r_t(v) + \delta(v, u)\}. \tag{1}$$

We next describe the online work function algorithm; see also [6, 4]. Intuitively, a good strategy for an online algorithm to process task τ_t is to move to a node where OPT would reside if τ_t would be the final task. However, the competitive ratio of an algorithm that solely sticks to this policy can become arbitrarily bad. A slight modification gives a $2n - 1$ competitive algorithm: Instead of blindly (no matter at what cost) traveling to the node of minimum work function value, we additionally take the transition cost into account. Essentially, this is the idea underlying the work function algorithm.

Work Function Algorithm (WFA): Let $s_0, \ldots, s_{t-1}$ denote the sequence of nodes visited by WFA to process $\mathcal{S}_{t-1}$. Then, to process task τ_t, WFA moves to a node s_t that minimizes $w_t(v) + \delta(s_{t-1}, v)$ for all $v \in V$. There is always a choice for s_t such that in addition $w_t(s_t) = w_{t-1}(s_t) + r_t(s_t)$. More formally,

$$s_t := \arg\min_{v \in V}\{w_t(v) + \delta(s_{t-1}, v)\} \quad \text{such that} \quad w_t(s_t) = w_{t-1}(s_t) + r_t(s_t). \tag{2}$$

In the sequel, we use WFA and OPT, respectively, to denote the work function and the optimal offline algorithm. For a given sequence $\mathcal{S} = \langle \tau_1, \ldots, \tau_\ell \rangle$ of tasks, WFA$[\mathcal{S}]$ and OPT$[\mathcal{S}]$ refer to the cost incurred by WFA and OPT on $\mathcal{S}$, respectively. By $s_0, \ldots, s_\ell$ we denote the sequence of nodes visited by WFA.

We continue by observing a few properties of work functions and of the online algorithm WFA (see Appendix A for the corresponding proofs).

Fact 1. *For any node u and any time t, $w_t(u) \geq w_{t-1}(u)$.*

Fact 2. *For any node u and any time t, $w_t(u) \leq w_{t-1}(u) + r_t(u)$.*

Fact 3. *For any two nodes u and v and any time t, $|w_t(u) - w_t(v)| \leq \delta(u, v)$.*

Fact 4. *At any time t, $w_t(s_t) = w_t(s_{t-1}) - \delta(s_{t-1}, s_t)$.*

Fact 5. *At any time t, $r_t(s_t) + \delta(s_{t-1}, s_t) = w_t(s_{t-1}) - w_{t-1}(s_t)$.*

3 Smoothing Models

Let the *adversarial task sequence* be given by $\check{\mathcal{S}} := \langle \check{\tau}_1, \ldots, \check{\tau}_r \rangle$. We smoothen each task vector $\check{\tau}_t := (\check{r}_t(v_1), \ldots, \check{r}_t(v_n))$ by perturbing each *original cost* entry $\check{r}_t(v_j)$ according to some probability distribution f as follows

$$r_t(v_j) := \max\{0, \check{r}_t(v_j) + \varepsilon(v_j)\}, \quad \text{where } \varepsilon(v_j) \leftarrow f.$$

That is, to each original cost entry we add a random number which is chosen from f. The obtained *smoothed* task is denoted by $\tau_t := (r_t(v_1), \ldots, r_t(v_n))$. We use μ and σ, respectively, to denote the expectation and the standard deviation of f. We assume that f is symmetric around $\mu := 0$. We take the maximum of zero and the smoothing outcome in order to assure that the smoothed costs are non-negative. Thus, the probability for an original zero cost entry to remain zero is amplified to $\frac{1}{2}$.

A major criticism to the additive model is that zero cost entries are destroyed. However, as we will argue in the next subsection, one can easily verify that the lower bound proof of $2n-1$ on the competitive ratio of any deterministic algorithm for metrical task systems goes through for any smoothing model that does not destroy zeros.

Our analysis holds for a large class of probability distributions, which we call *permissible*. We say f is permissible if (i) f is symmetric around $\mu = 0$, and (ii) f is non-increasing in $[0, \infty)$. For example, the uniform and the normal distribution are permissible. The concentration of f around μ is given by its standard deviation σ. Since the stated upper bounds on the smoothed competitive ratio of WFA do not further improve by choosing σ much larger than $U_{\min}$, we assume that $\sigma \leq 2U_{\min}$. Moreover, we use c_f to denote a constant depending on f such that for a random ε chosen from f, $\mathbf{P}[\varepsilon \geq \sigma/c_f] \geq \frac{1}{4}$.

All our results hold against an *adaptive adversary.* An adaptive adversary reveals the task sequence over time, thereby taking decisions made by the online algorithm in the past into account.

3.1 Lower Bound for Zero-Retaining Smoothing Models

The proof of the $2n - 1$ lower bound on the competitive ratio of any deterministic algorithm, see [6, 8, 4], is based only on the use of elementary tasks and the fact that the cost of the online algorithm is monotone increasing with the length of the input sequence. Assume we consider a zero-retaining smoothing model, i.e., a model in which zero cost entries are invariant to the smoothing. In such a model, elementary tasks are smoothed to elementary tasks. In particular this means that the above two properties still hold. Therefore, the lower bound proof also goes through for sequences that are smoothed according to any zero-retaining smoothing model.

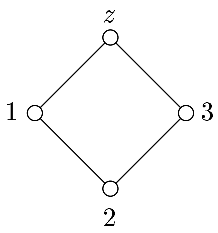
Theorem 1. *Every deterministic online algorithm* ALG *for metrical task systems has a smoothed competitive ratio of at least* $2n - 1$ *under a zero-retaining smoothing model.*

4 A Lower Bound on the Optimal Offline Cost

In this section, we establish a lower bound on the cost incurred by an optimal offline algorithm OPT when run on tasks smoothed according to the additive smoothing model. For the purpose of proving the lower bound, we first investigate an interesting version of a balls into bins experiment, which we call the *constrained balls into bins game.*

4.1 Constrained Balls into Bins Game

We are given n bins $B_1, \ldots, B_n$. In each round, we place a ball independently in each bin B_i with probability p; with probability $1 - p$ no ball is placed in B_i. We define the *height* $h_t(i)$ of a bin B_i as the number of balls in B_i after round t. We have dependencies



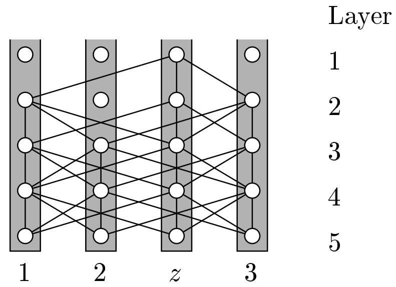


Figure 1: Illustration of the "unfolding" for $Q = 1$ and $h = 5$. Left: constraint graph G_c. Right: layered dependency graph $\mathcal{D}_h$.

between the heights of different bins that are specified by an (undirected) *constraint graph* $G_c := (V_c, E_c)$. The node set V_c of G_c contains n nodes $u_1, \ldots, u_n$, where each node u_i corresponds to a bin B_i. All edges in E_c have uniform edge lengths equal to Q. Let D be the maximum degree of a vertex in G_c. Throughout the experiment, we maintain the following invariant.

Invariant: The difference in height between two bins B_i and B_j is at most the shortest path distance between u_i and u_j in G_c.

If the placement of a ball into a bin B_i would violate this invariant, the ball is *rejected*; otherwise we say that the ball is *accepted*. Observe that if two bins B_i and B_j do not violate the invariant in round t, then, in round $t+1$, B_i and B_j might cause a violation only if one bin, say B_i, receives a ball, and the other, B_j, does not receive a ball; if both receive a ball, or both do not receive a ball, the invariant remains true.

Theorem 2. *Fix any bin B_z. Let R_z be the number of rounds needed until the height of B_z becomes $h \geq \log(n)$. Then,* $\mathbf{P}[R_z > c_3 h\,(1 + \log(D)/Q)] \leq 1/n^4$.

We remark that constraint graphs with $Q = 1$ exist, e.g., a clique on n nodes, such that the expected number of rounds needed for the height of a bin to become h is $\Omega(h \log(n))$. Moreover, for any given maximum degree D, one can create graph instances with $Q = 1$ such that the expected number of rounds is $\Omega(h \log(D))$.

We next describe how one can model the growth of the height of B_z by an alternative, but essentially equivalent, game on a *layered dependency graph*. A layered dependency graph $\mathcal{D}_h$ consists of h layers, $V_1, \ldots, V_h$, and edges are present only between adjacent layers. The idea is to "unfold" the constraint graph G_c into a layered dependency graph $\mathcal{D}_h$.

We first describe the construction for $Q = 1$: Each layer of $\mathcal{D}_h$ corresponds to a subset of nodes in G_c. Layer 1 consists of z only, the node corresponding to bin B_z. Assume we have constructed layers $V_1, \ldots, V_i$, $i < h$. Then, V_{i+1} is constructed from V_i by adding all nodes, $\Gamma_{G_c}(V_i)$, that are adjacent to V_i in G_c, i.e., $V_{i+1} := V_i \cup \Gamma_{G_c}(V_i)$. For every pair $(u, v) \in V_i \times V_{i+1}$, we add an edge (u, v) to $\mathcal{D}_h$ if $(u, v) \in E_c$, or $u = v$. See Figure 1 for an example.

Now, the following game on $\mathcal{D}_h$ is equivalent to the balls and bins game. Each node in $\mathcal{D}_h$ is in one of three states, namely UNFINISHED, READY or FINISHED. Initially, all nodes in layer h are READY and all other nodes are UNFINISHED. In each round, all READY nodes independently toss a coin; each coin turns up *head* with probability p and *tail* with probability $1-p$. A READY node changes its state to FINISHED if the outcome of its coin toss is *head*. At the end of each round, an UNFINISHED node in layer j changes its state to READY, if all its neighbors in layer $j+1$ are FINISHED.

Note that the nodes in layer V_j are FINISHED if and only if the corresponding bins B_i, $i \in V_j$, have height at least j. Consequently, the number of rounds needed until the root node z in $\mathcal{D}_h$ becomes FINISHED is equal to the number of rounds needed for the height of B_z to become h.

We use a similar construction if $Q > 1$. For simplicity, let h be a multiple of Q and define $h' = h/Q$. We construct a dependency graph $\mathcal{D}_{h'}$ with h' layers as described above (replace h by h' in the description above). Then, we transform $\mathcal{D}_{h'}$ into a layered graph $\mathcal{D}_h$ with h layers as follows. Let v be a node in layer j of $\mathcal{D}_{h'}$. We replace v by a path $(v_1, \ldots, v_k)$, where $k = |Q|$. Node v_1 is connected to all neighbors of v in layer $j-1$ and node v_k is connected to all neighbors of v in layer $j+1$. This replacement makes sure that the number of rounds needed until the root node becomes FINISHED in $\mathcal{D}_h$ dominates the number of rounds needed for the height of B_z to become h.

Proof of Theorem 2 . Let $\mathcal{D}_h$ be a layered dependency graph constructed from G_c as described above. As argued above, the event $(R_z \leq t)$ is stochastically dominated by the event that the root node becomes FINISHED within t rounds in $\mathcal{D}_h$. Consider the event that the root node z does not become FINISHED after t rounds. Then, there exists a *bad* path $P := (v_1, \ldots, v_h)$ from $z = v_1$ to some node v_h in the bottom layer h such that no node v_i of P was delayed by nodes other than $v_{i+1}, \ldots, v_h$. Put differently, P was delayed independently of any other path. Consider the outcome of the coin flips only for the nodes along P. If P is bad then the number of coin flips, denoted by X, that turned up *head* within t rounds is at most $h-1$. Let $\alpha(t)$ denote the probability that P is bad, i.e., $\alpha(t) := \mathbf{P}[X \leq h-1]$. Clearly, $\mathbf{E}[X] = tp$.

Observe that in $\mathcal{D}_h$ (i) at most h' layers contain nodes of degree larger than 2, and (ii) these nodes have at most $D+1$ neighbors in the next larger layer. That is, the number of possible paths from z to any node v in layer h is bounded by $(D+1)^{h'}$.

Thus, $\mathbf{P}[R_z > t] \leq \alpha(t)(D+1)^{h'}$. We want to choose t such that this probability is at most $1/n^4$. If we choose $t \geq (32/p)(h + h'\log(D))$ and use Chernoff's bound (see [9]) on X, we obtain for $h \geq \log(n)$

$$\alpha(t) = \mathbf{P}[X \leq h-1] \leq \mathbf{P}[X \leq pt/2] \leq e^{-pt/8} \leq \frac{1}{n^4(D+1)^{h'}}.$$

□

4.2 Lower Bound

We are now in a position to prove that an optimal offline algorithm incurs with high probability a cost of at least $n\gamma U_{\min}$ on a sequence of $\Theta(n\gamma\,(U_{\min}/\sigma + \log(D)))$ tasks.

Lemma 1. *Let $\check{S}$ be an adversarial sequence of $\ell := \lceil c_2 n\gamma(U_{\min}/\sigma + \log(D))\rceil$ tasks, for a fixed constant c_2 and some $\gamma \geq 1$. Then, $\mathbf{P}[\mathrm{OPT}[\mathcal{S}] < n\gamma U_{\min}] \leq 1/n^3$.*

We will use the Lemma 1 several times as follows.

Corollary 1. *Let $\check{\mathcal{S}}$ be an adversarial sequence of $\ell := \lceil c_2 n\gamma(U_{\min}/\sigma + \log(D))\rceil$ tasks, for a fixed constant c_2 and an some $\gamma \geq 1$. Then, the smoothed competitive ratio of* WFA *is at most*

$$\mathbf{E}\left[\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\right] \leq \frac{\mathbf{E}[\text{WFA}[\mathcal{S}]]}{n\gamma U_{\min}} + o(1).$$

Proof. Let $\mathcal{S}$ be a random variable denoting a smoothed sequence obtained from $\check{\mathcal{S}}$. We define $\mathcal{E}$ as the event that OPT incurs a cost of at least $n\gamma U_{\min}$ on $\mathcal{S}$. From the discussion above, we know that $\mathbf{P}[\neg\mathcal{E}] \leq 1/n^3$. Thus,

$$\begin{aligned}\mathbf{E}\left[\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\right] &= \mathbf{E}\left[\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\,\middle|\,\mathcal{E}\right]\mathbf{P}[\mathcal{E}] + \mathbf{E}\left[\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\,\middle|\,\neg\mathcal{E}\right]\mathbf{P}[\neg\mathcal{E}]\\ &\leq \frac{\mathbf{E}[\text{WFA}[\mathcal{S}]\,|\,\mathcal{E}]\,\mathbf{P}[\mathcal{E}]}{n\gamma U_{\min}} + \frac{2n-1}{n^3} \leq \frac{\mathbf{E}[\text{WFA}[\mathcal{S}]]}{n\gamma U_{\min}} + o(1),\end{aligned}$$

where the second inequality follows from the definition of $\mathcal{E}$ and the fact that the (worst case) competitive ratio of WFA is $2n-1$. □

Proof of Lemma 1 . The cost of OPT on a smoothed sequence $\mathcal{S}$ of length ℓ is $\text{OPT}[\mathcal{S}] = \min_{u\in V}\{w_\ell(u)\}$. Therefore, it suffices to prove that with probability at least $1 - 1/n^3$, $w_\ell(u) \geq n\gamma U_{\min}$ for each $u \in V$. Observe that we can assume that the initial work function values are all set to zero, since this can only reduce the cost of OPT.

We relate the growth of the work function values to the balls and bins experiment. For each node v_i of G we have a corresponding bin B_i. The constraint graph G_c is obtained from G by setting all edge lengths to $Q := \lfloor U_{\min}/\Delta\rfloor$, where $\Delta := \min\{U_{\min}, \sigma/c_f\}$. The placement of a ball in B_i in round t corresponds to the event $(r_t(v_i) \geq \sigma/c_f)$. Since our smoothing distribution satisfies $\mathbf{P}[\varepsilon \geq \sigma/c_f] \geq \frac{1}{4}$, we have that for any v_i and any t the smoothed request cost $r_t(v_i)$ is at least σ/c_f with probability at least $\frac{1}{4}$, irrespectively of its original cost entry and independently of the other request costs. Therefore, in each round t we place a ball into each bin with probability $p = \frac{1}{4}$.

By Lemma 2 given below, the number of rounds needed until a bin B_i has height h stochastically dominates the time t needed until $w_t(v_i) \geq h\Delta$. Applying Theorem 2, we obtain that for any bin B_i, after $\ell \geq c_3 h(1 + \log(D)/Q)$ rounds, $\mathbf{P}[h_\ell(i) < h] \leq 1/n^4$. Consequently, after ℓ rounds with probability at least $1 - 1/n^3$ all bins have height at least h. Choosing $h := 2n\gamma Q$, this implies that at time ℓ with probability at least $1 - 1/n^3$, $w_\ell(v_i) \geq 2n\gamma Q\Delta \geq n\gamma U_{\min}$ for all v_i of G. Finally, we make sure that $\ell := c_2 n\gamma(U/\sigma + \log(D)) \geq c_3 h(1 + \log(D)/Q)$ by fixing $c_2 := 4c_3\lceil c_f\rceil$. □

Lemma 2. *Consider any node v_i and its corresponding bin B_i. Let $h_t(i)$ denote the number of balls in bin B_i after t rounds. Then, for any $t \geq 0$, $w_t(v_i) \geq h_t(i)\,\Delta$.*

Proof. We proof the lemma by induction on the number of rounds t. For $t = 0$, the lemma clearly holds. (We can assume that the initial work function values are all zero.) Assume that the induction hypothesis holds after t rounds. In round $t+1$, if no ball is accepted in any bin then clearly the hypothesis remains true. Consider the case where at least one ball is accepted by some bin B_i. By the induction hypothesis, we have

$w_t(v_i) \geq h_t(i)\Delta$. Let v_k be the node that determines the work function value $w_{t+1}(v_i)$, i.e.,

$$w_{t+1}(v_i) = w_t(v_k) + r_{t+1}(v_k) + \delta(v_i, v_k).$$

Assume that $v_k = v_i$. Then, the work function value of v_i increases by the request cost $r_{t+1}(v_i)$, and since a ball was accepted in B_i, $r_{t+1}(v_i) \geq \Delta$. Thus, we have $w_{t+1}(v_i) \geq w_t(v_i) + \Delta \geq (h_t(i) + 1)\Delta = h_{t+1}(i)\Delta$, and we are done.

Next, assume that $v_k \neq v_i$. Let d be the shortest path distance between v_i and v_k in the constraint graph. Since in round $t+1$ a ball was accepted in B_i, B_i and B_k do not violate the invariant. Therefore,

$$h_t(i) - h_t(k) \leq d - 1 + [\text{ball accepted in } B_k \text{ in round } t+1],$$

where "[*statement*]" is 1 if *statement* is true, and 0 otherwise. By multiplying both sides with Δ and rearranging terms, we obtain

$$(h_t(k) + d)\Delta \geq (h_t(i) + 1 - [\text{ball accepted in } B_k \text{ in round } t+1])\Delta.$$

Observe that $d\Delta \leq \delta(v_i, v_k)$ by the definition of d and the edge lengths Q of the constraint graph. Moreover, $r_{t+1}(v_k) \geq [\text{ball accepted in } B_k \text{ in round } t+1]\Delta$. Thus,

$$\begin{aligned} w_{t+1}(v_i) &= w_t(v_k) + r_{t+1}(v_k) + \delta(v_i, v_k) \\ &\geq h_t(k)\Delta + [\text{ball accepted in } B_k \text{ in round } t+1]\Delta + d\Delta \\ &\geq (h_t(i) + 1)\Delta = h_{t+1}(i)\Delta. \end{aligned}$$

□

5 First Upper Bound

We can use the lower bound obtained in the last section to derive our first upper bound on the smoothed competitive ratio of WFA. We prove the following deterministic bound on the cost of WFA.

Lemma 3. *Let $\mathcal{K}$ be any request sequence of length ℓ. Then,* WFA$[\mathcal{K}] \leq$ OPT$[\mathcal{K}] + Diam \cdot \ell$.

Proof. Let $s_0, \ldots, s_\ell$ denote the sequence of nodes visited by WFA. For any t, the cost incurred by WFA to process task t is $C(t) := r_t(s_t) + \delta(s_{t-1}, s_t)$. By Fact 5, we obtain $C(t) = w_t(s_{t-1}) - w_{t-1}(s_t)$. Hence,

$$\begin{aligned} \text{WFA}[\mathcal{K}] = \sum_{t=1}^{\ell} C(t) \;&=\; w_\ell(s_{\ell-1}) - w_0(s_1) + \sum_{t=1}^{\ell-1} w_t(s_{t-1}) - w_t(s_{t+1}) \\ &\leq w_\ell(s_{\ell-1}) + (\ell - 1) \cdot Diam \;\leq\; \min_{v \in V}\{w_\ell(v)\} + \ell \cdot Diam, \end{aligned}$$

where the last two inequalities follow from Fact 3. Since OPT$[\mathcal{K}] \geq \min_{v \in V} w_\ell(v)$, the lemma follows. □

Theorem 3. *The smoothed competitive ratio of* WFA *is*

$$O\left(\frac{Diam}{\sigma} + \frac{Diam}{U_{\min}} \cdot \log(D)\right).$$

Proof. Consider an adversarial task sequence $\check{\mathcal{S}}$ of length $\ell := \lceil c_2 n\gamma\,(U_{\min}/\sigma + \log(D))\rceil$ for some $\gamma \geq 1$. Let $\mathcal{S}$ be a random variable denoting a smoothed sequence obtained from $\check{\mathcal{S}}$. Due to the proof of Corrollary 1 it suffices to bound $\mathbf{E}[\textsc{wfa}[\mathcal{S}]/\textsc{opt}[\mathcal{S}]\,|\,\mathcal{E}]$. Using Lemma 3, we have for any sequence $\mathcal{K}$ of ℓ tasks, $\textsc{wfa}[\mathcal{K}] \leq \textsc{opt}[\mathcal{K}] + \mathit{Diam} \cdot \ell$. Thus,

$$\mathbf{E}\left[\frac{\textsc{wfa}[\mathcal{S}]}{\textsc{opt}[\mathcal{S}]}\,\middle|\,\mathcal{E}\right] \leq \mathbf{E}\left[\frac{\textsc{opt}[\mathcal{S}] + \mathit{Diam}\cdot\ell}{\textsc{opt}[\mathcal{S}]}\,\middle|\,\mathcal{E}\right] \;\leq\; 1 + \frac{\mathit{Diam}\cdot\ell}{n\gamma U_{\min}}$$
$$= O\left(\frac{\mathit{Diam}}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(D)\right)\right),$$

where the last equality follows from the definition of ℓ. □

6 Second Upper Bound

We prove a second upper bound on the smoothed competitive ratio of WFA. The idea is as follows. We derive two upper bounds on the smoothed competitive ratio of WFA. The first one is a deterministic bound, and the second one uses the probabilistic lower bound on OPT. We then combine these two bounds using the following fact. The proof of Fact 6 can be found in Appendix A.

Fact 6. *Let A, B, and X_i, $1 \leq i \leq m$, be positive quantities. We have*

$$\min\left\{\frac{A\sum_{i=1}^m X_i}{\sum_{i=1}^m X_i^2}, \frac{B\sum_{i=1}^m X_i}{m}\right\} \leq \sqrt{A\,B}.$$

Consider any deterministic task sequence $\mathcal{K}$ of length ℓ. Let $s_0, s_1, \ldots, s_\ell$ denote the sequence of nodes visited by WFA. Define $C(t) := r_t(s_t) + \delta(s_{t-1}, s_t)$ as the service cost plus the transition cost incurred by WFA in round t.

With respect to $\mathcal{K}$ we define T as the set of rounds, where the increase of the work function value of s_{t-1} is at least one half of the transition cost, i.e., $t \in T$ if and only if $w_t(s_{t-1}) - w_{t-1}(s_{t-1}) \geq \delta(s_{t-1}, s_t)/2$. Due to Fact 4 we have $w_t(s_{t-1}) = w_t(s_t) + \delta(s_{t-1}, s_t)$. Therefore, the above definition is equivalent to

$$T := \left\{t : w_t(s_t) - w_{t-1}(s_{t-1}) \geq -\frac{1}{2}\delta(s_{t-1}, s_t)\right\}. \tag{3}$$

We use $\bar{T}$ to denote the complement of T.

We first prove that the total cost of WFA on $\mathcal{K}$ is bounded by a constant times the total cost contributed by rounds in T.

Lemma 4. *Let $\mathcal{K}$ be a sufficiently long task sequence such that* $\textsc{wfa}[\mathcal{K}] \geq 6\mathit{Diam}$. *Then,* $\textsc{wfa}[\mathcal{K}] \leq 8\sum_{t\in T} C(t)$.

Proof. We have $w_\ell(s_\ell) - w_0(s_0) \geq -\mathit{Diam}$, since $w_0(s_0) \leq w_\ell(s_0)$ and due to Fact 3. Thus,

$$\sum_{t=1}^{\ell}(w_t(s_t) - w_{t-1}(s_{t-1})) \geq -\mathit{Diam}.$$

Let R^- be the set of rounds where $w_t(s_t) - w_{t-1}(s_{t-1}) < 0$, and let R^+ be the set of rounds where $w_t(s_t) - w_{t-1}(s_{t-1}) \geq 0$. The above inequality can be rewritten as

$$\sum_{t \in R^-} (w_{t-1}(s_{t-1}) - w_t(s_t)) \leq Diam + \sum_{t \in R^+} (w_t(s_t) - w_{t-1}(s_{t-1})).$$

Since $\bar{T} \subseteq R^-$ and each term on the left hand side is non-negative, we have

$$\sum_{t \in \bar{T}} (w_{t-1}(s_{t-1}) - w_t(s_t)) \leq Diam + \sum_{t \in R^+} (w_t(s_t) - w_{t-1}(s_{t-1})). \tag{4}$$

For any $t \in \bar{T}$, we have $C(t) < 3\,(w_{t-1}(s_{t-1}) - w_t(s_t))$. This can be seen as follows. We have $w_{t-1}(s_t) \geq w_{t-1}(s_{t-1}) - \delta(s_{t-1}, s_t)$ (by Fact 3) and $r_t(s_t) = w_t(s_t) - w_{t-1}(s_t)$ (by (2)). Therefore, $r_t(s_t) \leq \delta(s_{t-1}, s_t) - w_{t-1}(s_{t-1}) + w_t(s_t)$. Moreover, since $t \in \bar{T}$ and by the definition (3) of T, $\delta(s_{t-1}, s_t) < 2(w_{t-1}(s_{t-1}) - w_t(s_t))$. Hence, $C(t) = r_t(s_t) + \delta(s_{t-1}, s_t) < 3\,(w_{t-1}(s_{t-1}) - w_t(s_t))$.

Furthermore, for any t, we have $w_t(s_t) - w_{t-1}(s_{t-1}) \leq C(t)$. This follows from $w_t(s_t) = w_{t-1}(s_t) + r_t(s_t)$ (by (2)) and $w_{t-1}(s_t) - w_{t-1}(s_{t-1}) \leq \delta(s_{t-1}, s_t)$ (by Fact 3). Since $R^+ \subseteq T$, we conclude

$$\sum_{t \in R^+} (w_t(s_t) - w_{t-1}(s_{t-1}) \leq \sum_{t \in R^+} C(t) \leq \sum_{t \in T} C(t).$$

Therefore, (4) implies

$$\frac{1}{3} \sum_{t \in \bar{T}} C(t) \leq Diam + \sum_{t \in T} C(t).$$

Exploiting the fact that $\textsc{wfa}[\mathcal{K}] = \sum_{t \in \bar{T}} C(t) + \sum_{t \in T} C(t)$ and $\textsc{wfa}[\mathcal{K}] \geq 6Diam$, we obtain $\textsc{wfa}[\mathcal{K}] \leq 8 \sum_{t \in T} C(t)$. ☐

We partition T into T^1 and T^2, where $T^1 := \{t \in T : w_t(s_t) - w_{t-1}(s_t) \leq 4U_{\max} diam\}$, and $T^2 = T \setminus T^1$. For any round t, we define $W_t := \sum_{i=1}^n w_t(v_i)$ and $\Delta W_t := W_t - W_{t-1}$.

Lemma 5. *Fix a round t and consider any node u such that $w_t(u) - w_{t-1}(u) \geq H$. If $H \leq 4U_{\max} diam$ then $\Delta W_t \geq H^2/(10U_{\max})$; otherwise, $\Delta W_t \geq nH/2$.*

Proof. Let $H \leq 4U_{\max} diam$. Define $d := \lfloor H/(8U_{\max}) \rfloor$. For $d = 0$, the claim clearly holds. Assume $d > 0$. Consider a shortest path $P := (u_0, u_1, \ldots, u_d)$ of edge length d starting from $u_0 := u$. Since $d \leq \lfloor diam/2 \rfloor$, there always exists a shortest path of length d. (Consider a breadth-first search tree rooted at u_0; the depth of this tree is at least $\lceil diam/2 \rceil$.) By Fact 3, we have for each i, $0 \leq i \leq d$, $w_t(u_i) \geq w_t(u_0) - iU_{\max}$ and $w_{t-1}(u_i) \leq w_{t-1}(u_0) + iU_{\max}$.

Therefore,

$$\begin{aligned}\sum_{i=0}^{d} (w_t(u_i) - w_{t-1}(u_i)) &\geq \sum_{i=0}^{d} (w_t(u_0) - w_{t-1}(u_0)) - 2U_{\max} \sum_{i=1}^{d} i \\ &\geq (d+1)H - (d+1)dU_{\max} \geq (d+1)(H - dU_{\max}) \geq \frac{H^2}{10U_{\max}},\end{aligned}$$

where the last inequality holds since $d \leq H/(8U_{\max}) \leq d+1$.

Let $H > 4U_{\max} diam$. Since for any node v_i, $w_{t-1}(v_i) \leq w_{t-1}(u) + U_{\max} diam$ and $w_t(v_i) \geq w_t(u) - U_{\max} diam$, we have

$$\sum_{i=1}^{n} (w_t(v_i) - w_{t-1}(v_i)) \geq nH - 2nU_{\max} diam \geq nH/2.$$

□

Lemma 6. *Let $\mathcal{K}$ be a sufficiently long task sequence such that $\text{OPT}[\mathcal{K}] \geq 2Diam$. There exists a constant b such that*

$$\text{OPT}[\mathcal{K}] \geq \frac{1}{bn} \left(\frac{1}{U_{\max}} \sum_{t \in T^1} C(t)^2 + n \sum_{t \in T^2} C(t) \right).$$

Proof. For every node v_i, $w_\ell(v_i) \leq \min_{u \in V}\{w_\ell(u)\} + Diam$ (by Fact 3). Moreover, $\text{OPT}[\mathcal{K}] \geq \min_{u \in V}\{w_\ell(u)\}$. We obtain

$$\sum_{i=1}^{n} w_\ell(v_i) \leq n\text{OPT}[\mathcal{K}] + nDiam, \quad \text{or, equivalently,} \quad \text{OPT}[\mathcal{K}] \geq \frac{1}{n} \left(\sum_{i=1}^{n} w_\ell(v_i) - nDiam \right).$$

Since $\text{OPT}[\mathcal{K}] \geq 2Diam$, the latter reduces to

$$\text{OPT}[\mathcal{K}] \geq \frac{2}{3n} \sum_{i=1}^{n} w_\ell(v_i). \tag{5}$$

Claim 1. *For any $t \in T^1$, $\Delta W_t \geq C(t)^2/(160U_{\max})$.*

Proof. By (2) we have $r_t(s_t) = w_t(s_t) - w_{t-1}(s_t)$. Below, we will show that

$$\Delta W_t \geq \left(\delta(s_{t-1}, s_t)^2 + r_t(s_t)^2\right) / (80U_{\max}). \tag{6}$$

Since $C(t)^2 = (\delta(s_{t-1}, s_t) + r_t(s_t))^2 \leq 2(\delta(s_{t-1}, s_t)^2 + r_t(s_t)^2)$, we conclude that $\Delta W_t \geq C(t)^2/(160U_{\max})$. Now, all that remains to be shown is (6). We distinguish two cases.

Let $\delta(s_{t-1}, s_t) \geq r_t(s_t)$. By the definition of T, we have $w_t(s_{t-1}) - w_{t-1}(s_{t-1}) \geq \delta(s_{t-1}, s_t)/2$. Using Lemma 5 with $H := \delta(s_{t-1}, s_t)/2$, we obtain

$$\Delta W_t \geq \delta(s_{t-1}, s_t)^2/(40U_{\max}) \geq \left(\delta(s_{t-1}, s_t)^2 + r_t(s_t)^2\right) / (80U_{\max}).$$

Let $\delta(s_{t-1}, s_t) < r_t(s_t)$. Since $w_t(s_t) - w_{t-1}(s_t) = r_t(s_t)$ and $r_t(s_t) \leq 4U_{\max} diam$ by the definition of T_1, using Lemma 5 with $H := r_t(s_t)$, we obtain

$$\Delta W_t \geq r_t(s_t)^2/(10U_{\max}) \geq (\delta(s_{t-1}, s_t)^2 + r_t(s_t)^2)/(20U_{\max}).$$

□

Claim 2. *For any $t \in T^2$, $\Delta W_t \geq 4nC(t)/10$.*

Proof. Since $t \in T^2$ and by (2), $r_t(s_t)/4 > diam U_{\max} \geq \delta(s_{t-1}, s_t)$. Thus, $C(t) = r_t(s_t) + \delta(s_{t-1}, s_t) < 5r_t(s_t)/4$. Furthermore, by (2) we have $r_t(s_t) = w_t(s_t) - w_{t-1}(s_t)$. Applying Lemma 5 with $H := r_t(s_t)$, we obtain $\Delta W_t \geq nr_t(s_t)/2 \geq 4nC(t)/10$. □

Claim 1 and Claim 2 together imply that

$$\sum_{i=1}^{n} w_\ell(v_i) \geq \sum_{t=1}^{\ell} \Delta W_t \geq \sum_{t \in T} \Delta W_t \geq \frac{1}{160 U_{\max}} \sum_{t \in T^1} C(t)^2 + \frac{4n}{10} \sum_{t \in T^2} C(t).$$

The proof now follows for an appropriate constant b from (5). □

Theorem 4. *The smoothed competitive ratio of* WFA *is*

$$O\left(\sqrt{n \cdot \frac{U_{\max}}{U_{\min}} \left(\frac{U_{\min}}{\sigma} + \log(D)\right)}\right).$$

Proof. Consider an adversarial task sequence $\check{\mathcal{S}}$ of length $\ell := \lceil c_2 n\gamma(U_{\min}/\sigma + \log(D))\rceil$, for an appropriate γ, and let $\mathcal{S}$ be a random variable denoting a smoothed sequence obtained from $\check{\mathcal{S}}$. Due to the proof of Corrollary 1 it suffices to bound $\mathbf{E}[\text{WFA}[\mathcal{S}]/\text{OPT}[\mathcal{S}] \,|\, \mathcal{E}]$, where $\mathcal{E}$ is the event $(\text{OPT}[\mathcal{S}] \geq n\gamma U_{\min})$. Consider a smoothing outcome $\mathcal{S}$ such that the event $\mathcal{E}$ holds. We fix γ sufficiently large such that $\text{OPT}[\mathcal{S}] \geq 6Diam$. Observe that $\text{WFA}[\mathcal{S}] \geq \text{OPT}[\mathcal{S}] \geq 6Diam$.

First, assume $\sum_{t\in T^1} C(t) < \sum_{t \in T^2} C(t)$. Then, due to Lemma 4 and Lemma 6,

$$\text{WFA}[\mathcal{S}] \leq 16 \sum_{t \in T^2} C(t) \quad \text{and} \quad \text{OPT}[\mathcal{S}] \geq \frac{1}{b} \sum_{t \in T^2} C(t).$$

Hence, $\mathbf{E}[\text{WFA}[\mathcal{S}]/\text{OPT}[\mathcal{S}] \,|\, \mathcal{E}] = O(1)$.

Next, assume $\sum_{t\in T^1} C(t) \geq \sum_{t \in T^2} C(t)$. By Lemma 4 and Lemma 6 we have

$$\text{WFA}[\mathcal{S}] \leq 16 \sum_{t \in T^1} C(t) \quad \text{and} \quad \text{OPT}[\mathcal{S}] \geq \frac{1}{bn}\left(\frac{1}{U_{\max}} \sum_{t \in T^1} C(t)^2\right). \tag{7}$$

Thus,

$$\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]} \leq 16bnU_{\max}\left(\frac{\sum_{t\in T^1} C(t)}{\sum_{t\in T^1} C(t)^2}\right). \tag{8}$$

Since $\mathcal{E}$ holds, we also have

$$\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]} \leq \frac{\ell \cdot 16 \sum_{t\in T^1} C(t)}{\ell \cdot n\gamma U_{\min}} \leq \frac{c}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(D)\right)\left(\frac{\sum_{t\in T^1} C(t)}{|T^1|}\right), \tag{9}$$

where the latter inequality holds for an appropriate constant c and since $\ell \geq |T^1|$. Observe that (9) is well-defined since $\sum_{t\in T^1} C(t) \geq \frac{1}{16}\text{WFA}[\mathcal{S}]$ (by (7)) and $\text{WFA}[\mathcal{S}] \geq 6Diam$ imply that $|T^1| \geq 1$.

Applying Fact 6 to (8) and (9), these two bounds are combined to

$$\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]} \leq \sqrt{16bcn \cdot \frac{U_{\max}}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(D)\right)} = O\left(\sqrt{n \cdot \frac{U_{\max}}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(D)\right)}\right),$$

which concludes the proof. □

7 Potential Function

In this section we use a potential function argument to derive an upper bound on the expected cost of WFA.

Lemma 7. *Let $\check{\mathcal{S}}$ be an adversarial task sequence of length ℓ, and let $\mathcal{S} = \langle \tau_1, \ldots, \tau_\ell \rangle$ be a smoothed sequence obtained from $\check{\mathcal{S}}$. For each t, $1 \le t \le \ell$, and a given node s, define a random variable $\Delta_t(s) := \min_{u \in V}\{r_t(u) + \delta(u, s)\}$. Let $\kappa > 0$. If $\mathbf{E}[\Delta_t(s)] \le \kappa$ for each $s \in V$ and for each t, $1 \le t \le \ell$, then $\mathbf{E}[\text{WFA}[\mathcal{S}]] \le 4\kappa\ell + Diam$.*

Before we proceed to prove the lemma, we provide some intuition. Assume we consider a simple greedy online algorithm ALG that always moves to a node which minimizes the transition plus request cost. That is, ALG services task τ_t by moving from its current position, say s'_{t-1}, to a node s'_t that minimizes the expression $\min_{u \in V}\{r_t(u) + \delta(u, s'_{t-1})\}$. Clearly, if the requirement of Lemma 7 holds, the total expected cost of ALG on $\mathcal{S}$ is $\sum_{t=1}^{\ell} \mathbf{E}[\Delta_t(s_{t-1})] \le \ell\kappa$. The above lemma shows that the expected cost of the work function algorithm WFA is at most 4 times the expected cost of ALG plus some additive term. In the analysis, it will sometimes be convenient to consider ALG instead of WFA.

Proof of Lemma 7 . For $1 \le t \le \ell$, we denote by s_t the node in which WFA resides after task τ_t has been processed; we use s_0 to refer to the node in which WFA resides initially.

We define a potential function Φ as

$$\Phi(t) := w_t(s_t) + tDiam/\ell.$$

Observe that

$$\Phi(\ell) - \Phi(0) = w_\ell(s_\ell) - w_0(s_0) + Diam \ge w_\ell(s_\ell) - w_\ell(s_0) + Diam \ge 0,$$

where the last inequality follows from Fact 3 and since $\delta(s_\ell, s_0) \le Diam$.

We define the *amortized cost* $C_a(t)$ incurred by WFA to process task τ_t as

$$\begin{aligned} C_a(t) &:= r_t(s_t) + \delta(s_{t-1}, s_t) + \Phi(t) - \Phi(t-1) \\ &= r_t(s_t) + \delta(s_{t-1}, s_t) + w_t(s_t) - w_{t-1}(s_{t-1}) + Diam/\ell \\ &= w_t(s_t) - w_{t-1}(s_t) + w_t(s_{t-1}) - w_{t-1}(s_{t-1}) + Diam/\ell, \end{aligned} \tag{10}$$

where the last equality follows from Fact 5. Using Fact 3 and (1) we obtain that for each $u \in V$

$$w_{t-1}(s_t) \ge w_{t-1}(u) - \delta(u, s_t) \quad \text{and} \quad w_t(s_t) \le w_{t-1}(u) + r_t(u) + \delta(u, s_t).$$

Combining these two inequalities, we obtain

$$\begin{aligned} & w_t(s_t) - w_{t-1}(s_t) \le r_t(u) + 2\delta(u, s_t) \quad \text{for each } u \in V, \\ \text{and hence} \quad & w_t(s_t) - w_{t-1}(s_t) \le 2 \min_{u \in V}\{r_t(u) + \delta(u, s_t)\} \;=\; 2\Delta_t(s_t). \end{aligned}$$

A similar argument shows that $w_t(s_{t-1}) - w_{t-1}(s_{t-1}) \le 2\Delta_t(s_{t-1})$. Hence, we can rewrite (10) as

$$C_a(t) \;\le\; 2\Delta_t(s_t) + 2\Delta_t(s_{t-1}) + Diam/\ell.$$

Since $\text{WFA}[\mathcal{S}] = \sum_{t=1}^{\ell} C_a(t) - \Phi(\ell) + \Phi(0)$ and $\Phi(\ell) - \Phi(0) \geq 0$, we obtain

$$\mathbf{E}[\text{WFA}[\mathcal{S}]] \leq \mathbf{E}\left[\sum_{t=1}^{\ell} C_a(t)\right] \leq 2\mathbf{E}\left[\sum_{t=1}^{\ell} (\Delta_t(s_t) + \Delta_t(s_{t-1}))\right] + Diam \leq 4\kappa\ell + Diam.$$

□

If $\ell \geq Diam$ then the above bound reduces to $O(\kappa\ell)$. Corrollary 1 together with the upper bound of Lemma 7 yield the following corollary.

Corollary 2. *Let $\check{\mathcal{S}}$ be an adversarial sequence of $\ell := \lceil c_2 n\gamma(U_{\min}/\sigma + \log(D))\rceil$ tasks for a fixed constant c_2. If $\gamma \geq U_{\max}$, and therefore $\ell \geq Diam$, the smoothed competitive ratio of* WFA *is*

$$\mathbf{E}\left[\frac{\text{WFA}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\right] = O\left(\frac{\kappa\ell}{n\gamma U_{\min}}\right) = O\left(\kappa\left(\frac{1}{\sigma} + \frac{\log(D)}{U_{\min}}\right)\right).$$

8 Random Tasks

We derive an upper bound on the expected competitive ratio of WFA if each request cost is chosen independently from a probability distribution f which is non-increasing in $[0, \infty)$.

We need the following fact; the proof is given in Appendix A.

Fact 7. *Let f be a continuous, non-increasing distribution over $[0, \infty)$ with mean μ and standard deviation σ. Then, $\mu \leq \sqrt{12}\sigma$.*

Theorem 5. *If each request cost is chosen independently from a non-increasing probability distribution f over $[0, \infty)$ with standard deviation σ then the expected competitive ratio of* WFA *is*

$$O\left(1 + \frac{\sigma}{U_{\min}} \log(D)\right).$$

Proof. Let $\mathcal{S}$ be a random task sequence of length $\ell := \lceil c_2 n\gamma(U_{\min}/\sigma) + \log(D))\rceil$, for an appropriate $\gamma \geq U_{\max}$, generated from f. Observe that since $\gamma \geq U_{\max}$, we have $\ell \geq Diam$. For any t and any node s, we have

$$\Delta_t(s) = \min_{u \in V}\{r_t(u) + \delta(u, s)\} \leq r_t(s).$$

Since $r_t(s)$ is chosen from f, Fact 7 implies that $\mathbf{E}[\Delta_t(s)] \leq \kappa := \sqrt{12}\sigma$. Thus, by Lemma 7, we have $\mathbf{E}[\text{WFA}[\mathcal{S}]] = 4\sqrt{12}\sigma\ell + Diam = O(\sigma\ell)$.

Note that we can use the lower bound established in Section 4 to bound the cost of OPT: The generation of $\mathcal{S}$ is equivalent to smoothing (according to f) an adversarial task sequence consisting of all-zero request vectors only. Here, we do not need that the distribution f is symmetric around its mean. The theorem now follows from Corrollary 2. □

9 β-Elementary Tasks

We can strengthen the upper bound on the smoothed competitive ratio of WFA if the adversarial task sequence only consists of β-elementary tasks. Recall that in a β-elementary task the number of non-zero request costs is at most β.

Theorem 6. *If the adversarial task sequence only consists of β-elementary tasks then the smoothed competitive ratio of* WFA *is*

$$O\left(\beta \cdot \frac{U_{\max}}{U_{\min}} \left(\frac{U_{\min}}{\sigma} + \log(D)\right)\right).$$

We state the following fact; the proof is given in Appendix A.

Fact 8. *Let f be a permissible probability distribution. Then, $\mathbf{E}[\max\{0, \varepsilon\}] \leq \sigma$, where ε is a random variable chosen from f.*

We first prove the following lemma.

Lemma 8. *Let s be an arbitrary node of G. Consider a β-elementary adversarial task $\check{r}_t := (\check{r}_t(v_1), \ldots, \check{r}_t(v_n))$, where $\beta < n$. Then, $\mathbf{E}[\Delta_t(s)] \leq \sigma + \beta U_{\max}$.*

Proof. Let $V_0 \subseteq V$ be the set of all nodes with original cost zero, i.e., $V_0 := \{u \in V : \check{r}_t(u) = 0\}$. Then, $|V_0| \geq n - \beta$, and V_0 is non-empty if $\beta < n$. Let v^* be a node from V_0 which is closest to s. We have $\delta(v^*, s) \leq \beta U_{\max}$. (Otherwise, there must exist at least $\beta + 1$ nodes with non-zero original cost, a contradiction.) Thus,

$$\mathbf{E}[\Delta_t(s)] \leq \mathbf{E}[\min_{u \in V_0}\{r_t(u) + \delta(u, s)\}] \leq \mathbf{E}[r_t(v^*) + \delta(v^*, s)] \leq \sigma + \beta U_{\max},$$

where the last inequality follows since $r_t(v^*) = \max\{0, \varepsilon(v^*)\}$, $\varepsilon(v^*)$ is a random variable chosen from f, and Fact 8. □

Proof of Theorem 6 . Consider an adversarial task sequence $\check{\mathcal{S}}$ of length $\ell := \lceil c_2 n \gamma (U_{\min}/\sigma + \log(D)) \rceil$, for an appropriate $\gamma \geq U_{\max}$, and let $\mathcal{S}$ be a random variable denoting a smoothed sequence obtained from $\check{\mathcal{S}}$. By Lemma 8, $\mathbf{E}[\Delta_t(s)] \leq \kappa := \sigma + \beta U_{\max}$, which, since we assume that $\sigma \leq 2U_{\min}$, is $O(\beta U_{\max})$. The theorem now follows from Corrollary 2. □

10 Lower Bounds

In this section we present existential and universal lower bounds. All our lower bounds hold for any deterministic online algorithm ALG and against an adaptive adversary.

10.1 Existential Lower Bound for β-Elementary Tasks

We show an existential lower bound for β-elementary tasks on a line. We prove that the upper bound $O(\beta(U_{\max}/U_{\min})(U_{\min}/\sigma + \log(D)))$ established in Theorem 6 is tight up to a factor of $U_{\max}/U_{\min}$ if the underlying graph is a line. Later, we will use Theorem 7 to obtain our first universal lower bound.

Theorem 7. *Let G be a line graph. If the adversarial task sequence only consists of β-elementary tasks then the smoothed competitive ratio of* any *deterministic online algorithm* ALG *is*

$$\Omega\left(\min\left\{\beta\cdot\left(\frac{U_{\min}}{\sigma}+1\right),\frac{n}{\beta}\cdot\frac{U_{\min}}{U_{\max}}\right\}\right).$$

Proof. We use an averaging technique (see [6]). Divide the line into $h := n/(2\beta)$ contiguous segments of 2β nodes. For simplicity assume that h is an integer. (This does not affect the asymptotic lower bound.) We refer to these segments by $S_1, S_2, \ldots, S_h$.

Let s_t be the node in which ALG resides after the tth task. In round t, the adversary issues a β-elementary task by placing ∞ cost on each node that is within distance $\lceil\beta/2\rceil - 1$ from s_{t-1}, and zero cost on all other nodes. Let the random variable $\mathcal{S}$ denote a smoothed task sequence.

We consider a set $\mathbf{B}$ of h offline algorithms, one for each segment. Let $\mathbf{B}_j$ denote the offline algorithm that resides in segment S_j; $\mathbf{B}_j$ always stays in S_j. In each round t, each $\mathbf{B}_j$ moves to a node v in S_j minimizing the transition cost plus the request cost. Define $\mathbf{B}[\mathcal{S}] := \sum_{j=1}^{h}\mathbf{B}_j[\mathcal{S}]$ as the total cost incurred by the offline algorithms on $\mathcal{S}$; $\mathbf{B}_j[\mathcal{S}]$ is a random variable denoting the total cost incurred by $\mathbf{B}_j$ on $\mathcal{S}$. Clearly, $\tilde{\mathbf{B}}[\mathcal{S}] := \mathbf{B}[\mathcal{S}]/h$ is an upper bound on OPT$[\mathcal{S}]$.

Consider any round t. At most two consecutive line segments can have ∞ request costs. Moreover, in each segment at most β of the 2β nodes may have ∞ costs. Let $C_j(t)$ be the cost incurred by $\mathbf{B}_j$ in round t. Consider a segment S_j that receives a ∞ request cost. Then, $\mathbf{E}[C_j(t)] \leq \beta U_{\max} + \sigma$ by Lemma 8. Assume S_j does not receive any ∞ request cost. Then, $\mathbf{E}[C_j(t)] \leq \sigma$ by Fact 8.

Since in any round at most two segments may receive ∞ costs, we conclude

$$\mathbf{E}[\tilde{\mathbf{B}}[\mathcal{S}]] = \frac{1}{h}\mathbf{E}\left[\sum_{j=1}^{h}\mathbf{B}_j[\mathcal{S}]\right] = \frac{1}{h}\mathbf{E}\left[\sum_{j=1}^{h}\sum_{t=1}^{\ell}C_j(t)\right] \leq \ell\left(\frac{2(\beta U_{\max}+\sigma)}{h}+\sigma\right).$$

By Markov's inequality, $\mathbf{P}[\tilde{\mathbf{B}}[\mathcal{S}] < 2\mathbf{E}[\tilde{\mathbf{B}}[\mathcal{S}]]] \geq \frac{1}{2}$. Since in each round, ALG is forced to travel at least a distance of $\lceil\beta/2\rceil$, we have ALG$[\mathcal{S}] \geq \ell\beta U_{\min}/2$.

We conclude

$$\mathbf{E}\left[\frac{\text{ALG}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\right] \geq \left(\frac{1}{2}\right)\frac{\ell\beta U_{\min}/2}{2\ell\left(\frac{2(\beta U_{\max}+\sigma)}{h}+\sigma\right)} = \Omega\left(\frac{\beta U_{\min}}{\beta^2 U_{\max}/n+\sigma}\right).$$

That is, we obtain a lower bound of $\Omega((n/\beta)\cdot(U_{\min}/U_{\max}))$ if $\beta \geq \sqrt{n/(U_{\max}/\sigma)}$ and of $\Omega(\beta\cdot(U_{\min}/\sigma))$ if $\beta \leq \sqrt{n/(U_{\max}/\sigma)}$. In the latter case, exploiting that $\sigma \leq 2U_{\min}$, we obtain a $\Omega(\beta\cdot(U_{\min}/\sigma+1))$ bound. □

Observe that on a line the β-elementary bound of Theorem 6 is stronger than the general upper bound of Theorem 4 only if

$$\beta \leq \sqrt{\frac{nU_{\min}}{U_{\max}(U_{\min}/\sigma+1)}}.$$

In this case, Theorem 7 provides a lower bound of $\Omega(\beta\cdot(U_{\min}/\sigma+1))$. That is, for a line graph these bounds differ by a factor of at most $U_{\max}/U_{\min}$.

10.2 Universal Lower Bounds

We derive two universal lower bounds on the smoothed competitive ratio of any deterministic algorithm. The first universal bound uses the following corollary of Theorem 7.

Corollary 3. *Let G be a line graph. Any deterministic algorithm* ALG *has smoothed competitive ratio* $\Omega(\min\{n, \sqrt{n(U_{\min}/U_{\max})(U_{\min}/\sigma+1)}\}$ *against an adaptive adversary.*

Proof. Fix $\beta := \sqrt{nU_{\min}/(U_{\max}(U_{\min}/\sigma+1))}$ and use the lower bound given in Theorem 7. □

Theorem 8. Any *deterministic algorithm* ALG *has a smoothed competitive ratio of*

$$\Omega\left(\min\left\{diam,\ \sqrt{diam \cdot \frac{U_{\min}}{U_{\max}} \cdot \left(\frac{U_{\min}}{\sigma}+1\right)}\right\}\right).$$

Proof. We extend Theorem 7 to arbitrary graphs in a straightforward way. Consider a path in G of edge length at least *diam*. The adversary enforces that ALG and OPT never leave this path by specifying ∞ cost for each node that is not part of the path. The desired lower bound now follows from Corrollary 3. □

Next, we prove the following universal lower bound.

Theorem 9. Any *deterministic algorithm* ALG *has a smoothed competitive ratio of*

$$\Omega\left(\min\left\{n, \frac{U_{\min}}{\sigma} + \frac{U_{\min}}{U_{\max}} \cdot \log(D)\right\}\right).$$

Proof. The adversary issues a sequence of ℓ tasks as described below. For each t, $1 \leq t \leq \ell$, let s_t denote the node at which the deterministic online algorithm ALG resides after the tth task; we use s_0 to refer to the initial position of ALG.

We prove two different lower bounds. Combining these two lower bounds, we obtain the bound stated above.

We first obtain a lower bound of $\Omega(\min\{n, U_{\min}/\sigma\})$ assuming that $U_{\min}/\sigma \geq 1$. In round t, the adversary enforces a request cost of $U_{\min}$ on s_{t-1} and zero request cost on all other nodes. Recall that the adversary is adaptive and therefore knows the position of ALG.

We use an averaging technique to relate the cost of ALG to the average cost of a collection of offline algorithms. Let $\mathbf{B}$ be a collection of n offline algorithms. We place one offline algorithm at each node, and each offline algorithm remains at its node during the processing of the task sequence. Let $\mathcal{S}$ be a random variable denoting a smoothing outcome of $\check{\mathcal{S}}$. We define $\mathbf{B}[\mathcal{S}]$ as the total cost incurred by the n algorithms to process $\mathcal{S}$. Clearly, the average cost $\tilde{\mathbf{B}}[\mathcal{S}] := \mathbf{B}[\mathcal{S}]/n$ is an upper bound on OPT$[\mathcal{S}]$. It suffices to prove that with constant probability ALG$[\mathcal{S}]/\tilde{\mathbf{B}}[\mathcal{S}] = \Omega(\min\{n, U_{\min}/\sigma\})$.

For the analysis, we view the smoothing process as being done into two stages.

Stage 1: Initially we smoothen ℓ zero tasks (all request costs are zero) according to the given smoothing distribution. Let the smoothed sequence be $\mathcal{S}' := \langle \tau_1', \dots, \tau_\ell' \rangle$.

Stage 2: For each t, $1 \leq t \leq \ell$, we replace the request cost of s_{t-1} in τ_t' by the outcome of smoothing $U_{\min}$. We use τ_t to refer to the obtained task.

Let $\mathbf{R}'(v) := \sum_{t=1}^{\ell} r'_t(v)$ be the total request cost accumulated in v with respect to $\mathcal{S}'$. Moreover, we define ℓ random variables $U_1, \ldots, U_\ell$: U_t refers to the smoothed request cost $r_t(s_{t-1})$ of task τ_t obtained in Stage 2. For each $1 \leq t \leq \ell$, let Z_t be a 0/1 random variable which is 1 if and only if $U_t \geq U_{\min}$. We define $Z := \sum_{t=1}^{\ell} Z_t$. In the sequel, we condition the smoothing outcome $\mathcal{S}$ on the following three events: (i) $\mathcal{E} := (\sum_{v \in V} \mathbf{R}'(v) \leq 2n\ell\sigma)$, (ii) $\mathcal{F} := (\sum_{t=1}^{\ell} U_t \leq 4\ell U_{\min})$, and (iii) $\mathcal{G} := (Z \geq \ell/4)$.

We first argue that the event $(\mathcal{E} \cap \mathcal{F} \cap \mathcal{G})$ occurs with at least constant probability. (i) Due to Fact 8, $\mathbf{E}[\mathbf{R}'(v)] \leq \ell\sigma$ for each $v \in V$. By Markov's inequality, we thus have $\mathbf{P}[\mathcal{E}] \geq 1/2$. (ii) By Fact 8 and since $\sigma \leq U_{\min}$, we also have $\mathbf{E}[U_t] \leq U_{\min} + \sigma \leq 2U_{\min}$ for each $1 \leq t \leq \ell$. Hence by Markov's inequality, $\mathbf{P}[\sum_{t=1}^{\ell} U_t \geq 4\ell U_{\min}] \leq 1/2$. (iii) Since the smoothing distribution f is a symmetric, we have $\mathbf{P}[U_t \geq U_{\min}] \geq 1/2$ for each $1 \leq t \leq \ell$. Thus, $\mathbf{E}[Z_t] \geq 1/2$. Moreover, the Z_t's are independent. Applying Chernoff's bound (see [9]), we obtain $\mathbf{P}[Z \leq \ell/4] \leq e^{-\ell/16}$.

Since event $\mathcal{E}$ is defined with respect to $\mathcal{S}'$, it is independent of the event $(\mathcal{F} \cap \mathcal{G})$. Therefore,

$$\mathbf{P}[\mathcal{E} \cap \mathcal{F} \cap \mathcal{G}] \geq \frac{1}{2} \cdot \left(1 - \left(\frac{1}{2} + e^{-\ell/16}\right)\right) \geq \frac{1}{8},$$

where the last inequality holds if $\ell \geq 64$.

Let $\mathcal{S}$ be any fixed outcome of the smoothing such that $(\mathcal{E} \cap \mathcal{F} \cap \mathcal{G})$ holds. Assume that to process sequence $\mathcal{S}$, ALG changes its position in k of the ℓ rounds. Let T_k refer to the set of rounds where ALG changes its position. We bound the cost of the offline algorithms as follows. In any round t, the total cost incurred by the offline algorithms at nodes different from s_{t-1} is at most $\sum_{v \in V} r'_t(v)$. If ALG does not move in round t, both ALG and $\mathbf{B}$ incur a cost of U_t. If ALG moves in round t, $\mathbf{B}$ incurs an additional cost of U_t, since one algorithm resides in s_{t-1}. Thus,

$$\mathbf{B}[\mathcal{S}] \leq \text{ALG}[\mathcal{S}] + \sum_{t \in T_k} U_t + \sum_{v \in V} \mathbf{R}'(v) \leq \text{ALG}[\mathcal{S}] + 4\ell U_{\min} + 2n\ell\sigma,$$

where the last inequality follows from $\mathcal{F}$ and $\mathcal{E}$.

Since also $\mathcal{G}$ holds, we can conclude that ALG incurs a cost of at least $\ell U_{\min}/4$: In each of the at least $\ell/4$ rounds, we have $r_t(s_{t-1}) = U_t \geq U_{\min}$. That is, no matter whether ALG moves or stays in these rounds, it incurs a cost of at least $U_{\min}$.

Thus, conditioned on the event $(\mathcal{E} \cap \mathcal{F} \cap \mathcal{G})$ we obtain for an appropriate constant c

$$\frac{\text{ALG}[\mathcal{S}]}{\tilde{\mathbf{B}}[\mathcal{S}]} \geq \frac{\text{ALG}[\mathcal{S}]}{17\text{ALG}[\mathcal{S}]/n + 2\ell\sigma} \geq c \cdot \min\left\{n, \frac{U_{\min}}{\sigma}\right\}.$$

Next we obtain a lower bound of $\Omega((U_{\min}/U_{\max})\log(D))$. Consider a node s of G with degree D. Let V_s be the set of nodes containing s and all the neighbors of s in G. Define G_s as the subgraph of G induced by V_s. The adversary makes sure that every reasonable online algorithm will always reside at a node in V_0 by specifying in each round a request cost of ∞ for each $v \notin V_0$. In addition, in each round t the adversary enforces the online algorithm to move by placing a request cost of ∞ at s_{t-1}. All other request cost are zero.

Let $\mathcal{S}$ be a smoothed task sequence obtained from $\check{\mathcal{S}}$. Since G_s is a star with $D+1$ nodes and the transition cost between any two nodes is at most $2U_{\max}$, Lemma 9 implies that there exists a deterministic offline algorithm $\mathbf{B}$ with $\mathbf{E}[\mathbf{B}[\mathcal{S}]] \leq 2c\ell U_{\max}/\log(D)$.

(Observe that we can apply Lemma 9 here since with respect to G_s the request sequence is elementary.) Applying Markov's inequality, we obtain $\mathbf{P}[\mathbf{B}[\mathcal{S}] \geq 4c\ell U_{\max}/\log(D)] \leq 1/2$. Since ALG has to move in each round to avoid ∞ cost, the cost of ALG for any smoothed sequence is at least $\ell U_{\min}$. Putting everything together, we obtain

$$\mathbf{E}\left[\frac{\text{ALG}[\mathcal{S}]}{\text{OPT}[\mathcal{S}]}\right] \geq \mathbf{E}\left[\frac{\text{ALG}[\mathcal{S}]}{\mathbf{B}[\mathcal{S}]}\right] \geq \left(\frac{1}{2}\right) \cdot \frac{\ell U_{\min}}{4c\ell U_{\max}/\log(D)} = \Omega\left(\frac{U_{\min}}{U_{\max}} \cdot \log(D)\right).$$

□

Lemma 9. *Let G be a clique with $m+1$ nodes and maximum edge length $U_{\max}$. Consider an adversarial sequence $\check{\mathcal{S}}$ of ℓ elementary tasks for a sufficiently large ℓ. Then, there exists an offline algorithm $\mathbf{B}$ such that for $m \geq 16$, $\mathbf{E}[\mathbf{B}[\mathcal{S}]] \leq c\ell U_{\max}/\log(m)$ for a constant c.*

Proof. We first consider an adversarial sequence $\check{\mathcal{S}} = \langle \check{\tau}_1, \ldots, \check{\tau}_k \rangle$ of $k := \lfloor \log(m)/2 \rfloor$ elementary tasks. We view the smoothing of the elementary tasks as being done in two stages.

Stage 1: Initially we smoothen k zero tasks (all request costs are zero) according to the given smoothing distribution. Let the smoothed sequence be $\mathcal{S}' := \langle \tau'_1, \ldots, \tau'_k \rangle$.

Stage 2: For each t, $1 \leq t \leq k$, we obtain a task τ_t from τ'_t as follows. Let v^* be the node with non-zero request cost $\check{r}_t(v^*)$ in $\check{\tau}_t$. We replace the request cost of v^* in τ'_t by the outcome of smoothing $\check{r}_t(v^*)$. Let $\mathcal{S} := \langle \tau_1, \ldots, \tau_k \rangle$ be the resulting task sequence.

For any node v_i, we define a 0/1 random variable X_i which is 1 if and only if the total request cost accumulated in v_i with respect to $\mathcal{S}'$ is zero. Since for each node v_i the request cost remains zero with probability at least $\frac{1}{2}$, we have $\mathbf{P}[X_i = 1] \geq (1/2)^k \geq 1/\sqrt{m}$. Note that the X_i's are independent. Let $\mathbf{X} := X_1 + \cdots + X_{m+1}$. We have $\mathbf{E}[\mathbf{X}] \geq \sqrt{m}$. Let $\mathcal{E}$ denote the event $(\mathbf{X} > \sqrt{m}/2)$. Using Chernoff's bound (see [9]), we obtain

$$\mathbf{P}[\neg\mathcal{E}] = \mathbf{P}[\mathbf{X} \leq \sqrt{m}/2] \leq e^{-\sqrt{m}/8}.$$

The offline algorithm $\mathbf{B}$ has two different strategies depending on whether event $\mathcal{E}$ holds or not.

Strategy 1: If event $\mathcal{E}$ holds, $\mathbf{B}$ moves at the beginning to a node v_i whose total accumulated request cost is zero and stays there. (Recall that $\mathbf{B}$ is offline.) Note that since $\mathcal{E}$ holds there are more than $\sqrt{m}/2 - k$ such nodes; for $m \geq 16$ there exists at least one such node.

Strategy 2: If event $\mathcal{E}$ does not hold, $\mathbf{B}$ always moves to a node with minimum request cost.

Since $\mathbf{B}$ only incurs the initial travel cost of at most $U_{\max}$ if $\mathcal{E}$ holds, we obtain

$$\mathbf{E}[\mathbf{B}[\mathcal{S}]] = \mathbf{E}[\mathbf{B}[\mathcal{S}] \mid \mathcal{E}]\ \mathbf{P}[\mathcal{E}] + \mathbf{E}[\mathbf{B}[\mathcal{S}] \mid \neg\mathcal{E}]\ \mathbf{P}[\neg\mathcal{E}] \leq U_{\max} + \mathbf{E}[\mathbf{B}[\mathcal{S}] \mid \neg\mathcal{E}] \cdot e^{-\sqrt{m}/8}.$$

Next, we bound $\mathbf{E}[\mathbf{B}[\mathcal{S}] \mid \neg\mathcal{E}]$. Clearly, the transition cost in each round is at most $U_{\max}$. The expected request cost incurred by $\mathbf{B}$ in round t is $\mathbf{E}[\min_{u \in V}\{r_t(u)\} \mid \neg\mathcal{E}]$. Consider a node v_i with $\check{r}_t(v_i) = 0$. The smoothed request cost of v_i is not affected by Stage 2. We have $\mathbf{E}[\min_{u \in V}\{r_t(u)\} \mid \neg\mathcal{E}] \leq \mathbf{E}[r_t(v_i) \mid \neg\mathcal{E}]$. Let $(X_1 = x_1, \ldots, X_{m+1} = x_{m+1})$ be any outcome such that $\neg\mathcal{E}$ holds. Since the request costs are chosen independently, we have $\mathbf{E}[r_t(v_i) \mid X_1 = x_1, \ldots, X_{m+1} = x_{m+1}] = \mathbf{E}[r_t(v_i) \mid X_i = x_i]$. If $x_i = 1$

then $\mathbf{E}[r_t(v_i) \mid X_i = x_i] = 0$, since all request costs at v_i must be zero. If $x_i = 0$ then $\mathbf{E}[r_t(v_i) \mid X_i = x_i] \leq \mathbf{E}[r_t(v_i) \mid r_t(v_i) > 0]$. (For $r_t(v_i)$ the event $(X_i = 0)$ means that either $r_t(v_i) = 0$ and $r_{t'}(v_i) > 0$ for some $t' \neq t$, or $r_t(v_i) > 0$.) By Fact 8, the expected cost $\mathbf{E}[r_t(v_i)]$ is at most σ. Moreover, $\mathbf{P}[r_t(v_i) > 0] \geq \mathbf{P}[r_t(v_i) \geq \sigma/c_f] \geq \frac{1}{4}$. Hence, $\mathbf{E}[r_t(v_i) \mid r_t(v_i) > 0] \leq 4\mathbf{E}[r_t(v_i)] \leq 4\sigma$. Putting everything together, we obtain

$$\mathbf{E}[\mathbf{B}[\mathcal{S}] \mid \neg\mathcal{E}] \leq \sum_{t=1}^{k} (\mathbf{E}[\min_{u \in V}\{r_t(u)\} \mid \neg\mathcal{E}] + U_{\max}) \leq k(4\sigma + U_{\max}) \leq 9kU_{\max},$$

where the last inequality holds since we assume that $\sigma \leq 2U_{\min} \leq 2U_{\max}$,

Altogether, we obtain for a sequence $\mathcal{S}$ of length k and for $m \geq 16$,

$$\mathbf{E}[\mathbf{B}[\mathcal{S}]] \leq U_{\max} + 9kU_{\max} \cdot e^{-\sqrt{m}/8} \leq 13U_{\max}.$$

We conclude the proof as follows. We split the entire adversarial sequence $\check{\mathcal{S}}$ of length ℓ into $j \geq 1$ subsequences of length k (the final one might have length less than k). On each subsequence, $\mathbf{B}$ performs as described above. We therefore obtain for the entire sequence $\mathcal{S}$ and an appropriate constant c

$$\mathbf{E}[\mathbf{B}[\mathcal{S}]] \leq \mathbf{E}\left[\sum_{t=1}^{j} 13U_{\max}\right] = 13jU_{\max} \leq \frac{c\ell U_{\max}}{\log(m)},$$

where the last inequality follows from the relation between ℓ and j and definition of k. □

10.3 Existential Lower Bounds

We provide two existential lower bounds showing that for a large range of parameters n, $U_{\min}$, $U_{\max}$, D, and Diam there exists a class of graphs on which *any* deterministic algorithm has a smoothed competitive ratio that asymptotically matches the upper bounds stated in Theorem 3 and Theorem 4. In order to prove these existential lower bounds, we first show the following lemma.

Lemma 10. *Given a number of nodes n, minimum edge cost $U_{\min}$, maximum edge cost $U_{\max}$, maximum degree $D \geq 3$, and diameter Diam such that*

$$\mathit{Diam} \geq 4U_{\min}\log_{D-1}(n), \quad \textit{and} \quad \mathcal{D} := \min\{\mathit{Diam}/U_{\max}, D\} \geq 17,$$

there exists a graph such that the smoothed competitive ratio of any deterministic algorithm ALG *is*

$$\Omega\left(\min\left\{\frac{nU_{\max}}{\mathit{Diam}}, \frac{\mathit{Diam}}{U_{\min}} \cdot \left(\frac{U_{\min}}{\sigma} + \log(\mathcal{D})\right)\right\}\right).$$

Proof. We construct a graph G as depicted in Figure 2. The graph consists of $m := \frac{1}{2}nU_{\max}/\mathit{Diam}$ cliques. Each clique has $\mathcal{D}$ nodes and the length of an edge between any two nodes is $U_{\min}$. We need to ensure that the maximum degree is at most D. Therefore, we connect each clique by a path to a $(D-1)$-ary tree T. Each such path consists of X edges of length $U_{\max}$. We assign a length of $U_{\min}$ to each edge in T. Each clique is attached to a leaf node of T; a leaf node may take up to $D-1$ cliques. Since m cliques need to be connected to T and we can attach at most $(D-1)^h$ cliques

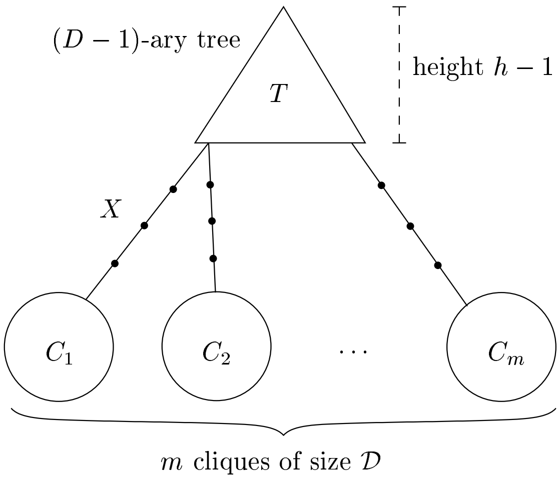


Figure 2:

to a tree of height $h-1$, we fix $h := \log_{D-1}(m)$. The total number of nodes in T is therefore $((D-1)^h - 1)/(D-2) \leq m$, since $D \geq 3$. It is easy to verify that $m + m \cdot (X-1) + m \cdot \mathcal{D} \leq n$, i.e., the total number of nodes in G is at most n. (If it is less than n, we let the remaining nodes become part of T.) The graph should have diameter Diam and thus we fix X such that $2(U_{\min} + X \cdot U_{\max} + (h-1)U_{\min}) = \mathit{Diam}$, i.e., $X := \lceil (\mathit{Diam}/2 - hU_{\min})/U_{\max} \rceil$. Moreover, we want that the minimum distance between any two nodes in different cliques is at least $\frac{1}{4}\mathit{Diam}$, i.e., $X \cdot U_{\max} \geq \frac{1}{8}\mathit{Diam}$. If $\mathit{Diam} \geq 4U_{\min} \log_{D-1}(n)$, this condition holds. (Also observe that in any graph of n nodes and maximum degree D, $\mathit{Diam}/U_{\min} \geq \log_{D-1}(n)$, i.e., our condition is slightly stronger.)

Consider the case $U_{\min}/\sigma > \log(\mathcal{D})$. We need to prove a lower bound of $\Omega(\min\{nU_{\max}/\mathit{Diam}, \mathit{Diam}/\sigma\})$. In each round, the adversary imposes an ∞ cost on all nodes of the graph except on those nodes that join a clique with its path. That is, the adversary restricts both ALG and OPT to stay in a "virtual" clique of size m with $U_{\min} = \frac{1}{4}\mathit{Diam}$ and $U_{\max} = \mathit{Diam}$. Applying the universal lower bound of Theorem 9 to this clique we obtain the desired lower bound of $\Omega(\min\{m, \mathit{Diam}/\sigma\})$.

Consider the case $U_{\min}/\sigma \leq \log(\mathcal{D})$. In each round, the adversary imposes an ∞ cost on all nodes in T and on all nodes that belong to a connecting path. Furthermore, in each round, the adversary forces the online algorithm ALG to leave its clique by specifying ∞ costs on all nodes of the clique in which ALG resides. All other request costs are zero.

We use an averaging technique. We define a collection of $m-1$ offline algorithms and compare the cost of ALG with the average cost of the offline algorithms. At most one algorithm resides in each clique. An offline algorithm $\mathbf{B}_i$ remains in its clique C_i until ∞ costs are imposed on C_i; at this point, $\mathbf{B}_i$ moves to the free clique. Within each clique, the offline algorithm follows the strategy as specified in the proof of Lemma 9. We may assume without loss of generality that each $\mathbf{B}_i$ starts in a different clique (see Appendix B).

Consider a smoothed sequence $\mathcal{S}$ of length ℓ. Let $\mathbf{B}[\mathcal{S}]$ be the total cost incurred by the offline algorithms and define $\mathbf{B}_i[\mathcal{S}]$ as the total cost of $\mathbf{B}_i$ on $\mathcal{S}$. The total cost of the offline algorithms to travel away from cliques with ∞ costs is at most $\ell Diam$. The expected cost of each algorithm in a clique with zero adversarial request cost is, due to Lemma 9, at most $c\ell U_{\min}/\log(\mathcal{D}-1)$; recall that each clique is of size $\mathcal{D} \geq 17$ and the maximum edge length in each clique is $U_{\min}$. Thus,

$$\mathbf{E}[\tilde{\mathbf{B}}[\mathcal{S}]] \leq \frac{\ell Diam}{m-1} + \frac{1}{m-1}\mathbf{E}\left[\sum_{i=1}^{m-1} \mathbf{B}_i[\mathcal{S}]\right] \leq \frac{\ell Diam}{m-1} + \frac{c\ell U_{\min}}{\log(\mathcal{D}-1)}.$$

By Markov's inequality, $\mathbf{P}[\tilde{\mathbf{B}}[\mathcal{S}] < 2\mathbf{E}[\tilde{\mathbf{B}}[\mathcal{S}]]] \geq \frac{1}{2}$. Clearly, $\textsc{alg}[\mathcal{S}] \geq \frac{1}{4}\ell Diam$. Therefore,

$$\mathbf{E}\left[\frac{\textsc{alg}[\mathcal{S}]}{\textsc{opt}[\mathcal{S}]}\right] \geq \left(\frac{1}{2}\right) \frac{\frac{1}{4}\ell Diam}{2(\frac{\ell Diam}{m-1} + \frac{c\ell U_{\min}}{\log(\mathcal{D}-1)})} = \Omega\left(\min\left\{m, \frac{Diam}{U_{\min}} \cdot \log(\mathcal{D})\right\}\right).$$

□

The next bound shows that if Theorem 3 gives a better upper bound than Theorem 4 then this bound is tight up to a factor of $\log(D)/\log(\mathcal{D}) \leq \log(n)$ for a large class of graphs.

Theorem 10. *There exists a class of graphs such that the smoothed competitive ratio of any deterministic algorithm* $\textsc{alg}$ *is*

$$\Omega\left(\min\left\{n, \frac{Diam}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(\mathcal{D})\right)\right\}\right),$$

where $\mathcal{D} = \min\{Diam/U_{\min}, D\}$.

Proof. If Theorem 3 gives a better upper bound than Theorem 4, we have

$$\frac{Diam}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(D)\right) \leq \sqrt{n \cdot \frac{U_{\max}}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(D)\right)},$$

which is equivalent to

$$\frac{nU_{\max}}{Diam} \geq \frac{Diam}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(D)\right).$$

Since $\log(D) \geq \log(\mathcal{D})$, we obtain from Lemma 10 the desired lower bound. □

Theorem 11. *There exist a class of graphs such that the smoothed competitive ratio of any deterministic algorithm* $\textsc{alg}$ *is*

$$\Omega\left(\min\left\{n, \sqrt{n\frac{U_{\max}}{U_{\min}}\left(\frac{U_{\min}}{\sigma} + \log(\mathcal{D})\right)}\right\}\right),$$

where $\mathcal{D} = \min\{Diam/U_{\min}, D\}$.

Proof. Let $U_{\min}/\sigma > \log(\mathcal{D})$. We fix $Diam$ such that $nU_{\max}/Diam = Diam/\sigma$, i.e., $Diam = \sqrt{n\sigma U_{\max}}$. The lower bound of Lemma 10 then reduces to $\Omega(\sqrt{nU_{\max}/\sigma})$.

Assume $U_{\min}/\sigma \leq \log(\mathcal{D})$. We fix $Diam$ such that $nU_{\max}/Diam = (Diam/U_{\min})\log(\mathcal{D})$, i.e., $Diam = \sqrt{nU_{\max}U_{\min}/\log(\mathcal{D})}$. The lower bound of Lemma 10 then reduces to $\Omega(\sqrt{n(U_{\max}/U_{\min})\log(\mathcal{D})})$. □

11 Concluding Remarks

In this chapter, we focused on the asymptotic behaviour of WFA if the request costs of an adversarial task sequence are perturbed by means of a symmetric additive smoothing model. We showed that the smoothed competitive ratio of WFA is much better than its worst case competitive ratio suggests and that it depends on topological parameters of the underlying graph. Moreover, all our bounds, except the one for β-elementary tasks, are tight up to constant factors. We believe that our analysis gives a strong indication that the performance of WFA in practice is much better than $2n - 1$.

An open problem would be to strengthen the universal lower bounds. Moreover, it would be interesting to obtain exact (and not only asymptotic) bounds on the smoothed competitive ratio of WFA.

Acknowledgements

We would like to thank Alberto Marchetti-Spaccamela for his comments on a first draft of this paper. Moreover, we would like to thank Kurt Mehlhorn for helpful suggestions and discussions; due to one of his suggestions we were able to significantly shorten the proof of Theorem 2.

A Proofs of Facts

Proof of Fact 3 . Assume x is the node that defines $w_t(v)$, i.e., $w_t(v) = w_{t-1}(x)+r_t(x)+\delta(x,v)$. We have $w_t(u) \le w_{t-1}(x)+r_t(x)+\delta(x,u) \le w_{t-1}(x)+r_t(x)+\delta(x,v)+\delta(v,u) = w_t(v)+\delta(v,u)$. □

Proof of Fact 4 . By (2), we have that $w_t(s_t) + \delta(s_{t-1}, s_t) \le w_t(v) + \delta(s_{t-1}, v)$ for all $v \in V$. In particular, for $v = s_{t-1}$ this implies $w_t(s_t) \le w_t(s_{t-1}) - \delta(s_{t-1}, s_t)$. On the other hand, due to Fact 3, $w_t(s_t) \ge w_t(s_{t-1}) - \delta(s_{t-1}, s_t)$. □

Proof of Fact 5 . Using (2) and Fact 4, we obtain

$$r_t(s_t) + \delta(s_{t-1}, s_t) = w_t(s_t) - w_{t-1}(s_t) + w_t(s_{t-1}) - w_t(s_t) = w_t(s_{t-1}) - w_{t-1}(s_t).$$

□

Proof of Fact 7. Let X be a random variable chosen from f. Define $\mathcal{E}$ as the event $(|X-\mu| \ge \mu/2)$. Using Chebyshev's inequality, we obtain

$$\mathbf{P}[\mathcal{E}] = \mathbf{P}\Big[|X-\mu| \ge \frac{\mu}{2}\Big] \le \frac{4\sigma^2}{\mu^2}. \tag{11}$$

Since f is continuous and non-increasing in $[0,\infty)$,

$$\mathbf{P}[\mathcal{E}] = \mathbf{P}\Big[|X-\mu| \ge \frac{\mu}{2}\Big] \ge \mathbf{P}\Big[X \le \frac{\mu}{2}\Big] \ge \frac{1}{2}\mathbf{P}\Big[\frac{\mu}{2} < X \le \frac{3\mu}{2}\Big] \ge \frac{1}{2}\mathbf{P}[\neg\mathcal{E}].$$

This implies that $\mathbf{P}[\mathcal{E}] \ge \frac{1}{3}$. Hence, (11) gives $\mu^2 \le 12\sigma^2$. □

Proof of Fact 8 . Define $Y := \max\{0, X\}$. Since $\mu = 0$, we have $\sigma^2 = \mathbf{E}[X^2]$. Let σ_Y denote the standard deviation of the distribution of Y. By the definition of $\mathbf{E}[X^2]$, $\mathbf{E}[Y^2] = \frac{1}{2}\mathbf{E}[X^2]$. Since $\sigma_Y^2 = \mathbf{E}[Y^2] - \mathbf{E}[Y]^2$ and $\sigma_Y^2 \ge 0$, we have $\mathbf{E}[Y]^2 \le \mathbf{E}[Y^2]$. This in turn implies that $\mathbf{E}[Y] \le \sigma/\sqrt{2}$. □

Proof of Fact 6 . Define $\mathcal{X} := \min\left\{\frac{A\sum_{i=1}^m X_i}{\sum_{i=1}^m X_i^2}, \frac{B\sum_{i=1}^m X_i}{m}\right\}$. First, note that

$$m(X_1^2 + X_2^2 + \cdots + X_m^2) \ \ge\ (X_1 + X_2 + \cdots + X_m)^2, \tag{12}$$

because

$$\frac{1}{2}\sum_{i,j}\left(X_i^2 + X_j^2\right) \ge \sum_{i=1}^m X_i^2 + \sum_{i,j,i\ne j} X_iX_j.$$

Define $Y := \sum_{i=1}^m X_i/m$. Note that Y is positive. Due to (12), we can write $\mathcal{X} \le \min\{A/Y, BY\}$. The latter expression is maximized if $A/Y = BY$, i.e., if $Y = \sqrt{A/B}$. Thus $\mathcal{X} \le \sqrt{AB}$. □

B Constant Additive Cost of the Offline Algorithm

We would like to point out that in our lower bound proofs we can assume without loss of generality that OPT incurs an additional additive cost of Z which is independent of the length of the input sequence. This does not change the asymptotics of the lower bounds, which can be seen as follows. We always prove a lower bound of say $\Omega(Y/X)$ on a task sequence of length ℓ by showing that with constant probability the expected cost of ALG is at least $Y \cdot \ell$ and the cost of OPT is at most $X \cdot \ell$. In order to make sure that the additive cost Z does not influence the competitive ratio, we only have to make sure that the task sequence under consideration is sufficiently long. If we choose ℓ such that $X \cdot \ell \geq Z$, we obtain a lower bound of $\Omega((Y \cdot \ell)/(X \cdot \ell + Z)) = \Omega(Y/X)$.

References

[1] S. Albers. The influence of lookahead in competitive paging algorithms (extended abstract). In *European Symposium on Algorithms*, pages 1–12, 1993.

[2] S. Albers. A competitive analysis of the list update problem with lookahead. *Theoretical Computer Science*, 197(1–2):95–109, 1998.

[3] L. Becchetti, S. Leonardi, A. Marchetti-Spaccamela, G. Schäfer, and T. Vredeveld. Average case and smoothed competitive analysis of the multi-level feedback algorithm. In *Proceedings of the Forty-Fourth Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 462–471, 2003.

[4] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis.* Cambridge University Press, 1998.

[5] A. Borodin, S. Irani, P. Raghavan, and B. Schieber. Competitive paging with locality of reference. *Journal of Computer and System Sciences*, 50(2):244–258, 1995.

[6] A. Borodin, N. Linial, and M. Saks. An optimal online algorithm for metrical task systems. *Journal of the ACM*, 39:745–763, 1992.

[7] E. Koutsoupias and C. Papadimitriou. Beyond competitive analysis. In *Proceedings of the Twenty-Fifth Symposium on Foundations of Computer Science*, pages 394–400, 1994.

[8] M. S. Manasse, L. A. McGeoch, and D. D. Sleator. Competitive algorithms for on-line problems. In *ACM Symposium on Theory of Computing*, pages 322–333, 1988.

[9] R. Motwani and P. Raghavan. *Randomized algorithms.* Cambridge University Press, 1 edition, 1995.

[10] D. Sleator and R. E. Tarjan. Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28:202–208, 1985.

[11] D. Spielman and S. H. Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. In *ACM Symposium on Theory of Computing*, 2001.