On the Competitive Ratio of the Work Function Algorithm for the k-Server Problem

Yair Bartal [1]

Computer Science School, Hebrew University, Jerusalem, Israel

Elias Koutsoupias [2]

Department of Informatics, University of Athens, Athens, Greece and Computer Science Department, Univesity of California, Los Angeles, USA

Abstract

The k-server problem is one of the most fundamental online problems. The problem is to schedule k mobile servers to visit a sequence of points in a metric space with minimum total mileage. The k-server conjecture of Manasse, McGeogh, and Sleator states that there exists a k-competitive online algorithm. The conjecture has been open for over 15 years. The top candidate online algorithm for settling this conjecture is the Work Function Algorithm (WFA) which was shown to have competitive ratio at most $2k-1$. In this paper we lend support to the conjecture that WFA is in fact k-competitive by proving that it achieves this ratio in several special metric spaces: the line, the star, and all metric spaces with $k+2$ points.

1 Introduction

The k-server problem [11] together with its special case, the paging problem, is probably the most influential online problem. The famous k-server conjecture has been open for over 15 years. Yet, the problem itself is easy to state: There are k servers that can move in a metric space. Their purpose is to service a sequence of requests. A request is simply a point of the metric space and servicing it entails moving a server to the requested point. The objective is to

Email addresses: yair@cs.huji.ac.il (Yair Bartal), elias@di.uoa.gr (Elias Koutsoupias).

[1] Supported in part by a grant of the Israeli Science Foundation (195/02).
[2] Supported in part by NSF grant CCR-0105752.

minimize the total distance traveled by all servers. In the online version of the problem, the requests are presented one-by-one. The notorious k-server conjecture states that there is an online algorithm that has competitive ratio k on any metric space. The top candidate online algorithm for settling the k-server conjecture is the Work Function Algorithm (WFA) which was shown [7,9,8] to have competitive ratio at most $2k - 1$.

In this paper we prove three results that support the conjecture that the WFA is k-competitive. The first and second result establish that the WFA is k-competitive for the line and for the "symmetric weighted cache" (represented by weighted star instances). It was known [3,4] that the k-server conjecture holds for these instances, but the algorithm employed was not the WFA, but the elegant Double Coverage algorithm, which has no natural extension for non-tree like metric spaces. Our third result is a new proof of the WFA is k-competitive for metric spaces of $k + 2$ points. This was first shown in [7,10] using an involved potential. Our proof here uses a simpler potential.

There is an interesting underlying connection between the three results of this work. In all cases, the number of minimizers (to be defined later) is at most $k + 1$. Although this fact by itself cannot guarantee that the WFA is k-competitive, it is at the heart of our proofs.

2 Preliminaries

We summarize here our notation, conventions and definitions. For a more thorough discussion that includes the history of the problem see [9,2]. Let $\rho = r_1 \dots r_n$ be a request sequence. The *work function* $w_i(X)$ is defined for each multiset X of k points (any such multiset is called a configuration). The value $w_i(X)$ is the optimal cost for servicing $r_1 \dots r_i$ starting at some fixed initial configuration A_0 and ending at the configuration X. The Work Function Algorithm (WFA) works as follows: Let A_i be its configuration just before servicing request r_{i+1}. To service r_{i+1}, it moves to configuration A_{i+1} that contains r_{i+1} and minimizes $w_{i+1}(A_{i+1}) + d(A_i, A_{i+1})$.

Chrobak and Larmore [5] introduced the concept of *extended cost* of the WFA (which they call pseudocost): The extended cost for request r_{i+1} is equal to the maximum increase of the work function: $\max_X\{w_{i+1}(X) - w_i(X)\}$. They showed that the extended cost is equal to the online plus the off-line cost (see also [9]). Consequently, to prove that the Work Function Algorithm is c-competitive, it suffices to bound the total extended cost by $(c+1)\text{OPT}(\rho) +$ const, where $\text{OPT}(\rho)$ is the optimal (off-line) cost to service ρ and const a constant that is independent of the request sequence.

For general metric spaces, the best known upper bound on the competitive ratio for the k-server problem is $2k-1$ [7,9] (see also [8] for a simpler proof), which improved the previous exponential (in k) bounds [6,1]. The algorithm employed in [7,9] to establish the $2k-1$ bound is the WFA. The proof is based on some fundamental properties (Quasiconvexity and Duality) of work functions. Here we will make use of the Duality property which characterizes the configurations that achieve the maximum $\max_X\{w_{i+1}(X) - w_i(X)\}$.

Lemma 1 (Duality Lemma [7,9]) *Let X be a configuration that minimizes*

$$w_i(X) - \sum_{x \in X} d(r_{i+1}, x).$$

Then X minimizes also

$$w_{i+1}(X) - \sum_{x \in X} d(r_{i+1}, x)$$

and maximizes the extended cost

$$\max_X \{w_{i+1}(X) - w_i(X)\}.$$

A configuration X that minimizes $w_i(X) - \sum_{x \in X} d(p, x)$ will be called a *minimizer* of p with respect to w_i.

3 The WFA for the line

In this section, we will show that the WFA is k-competitive in the line. To simplify the presentation, we assume that all requests are in a fixed interval $[a, b]$. Let us denote the configuration that contains m copies of a and $k-m$ copies of b as $a^m b^{k-m}$. We shall call these configurations *extreme*. Observe that there are exactly $k+1$ extreme configurations that correspond to $m = 0, \dots, k$. The next lemma shows that we can generally assume that minimizers are extreme configurations.

Lemma 2 *Assume that all initial points and all requests are in the interval $[a, b]$. For any point $p \in [a, b]$ and any work function w_i, there is $m \in \{0, \dots, k\}$ such that $a^m b^{k-m}$ is a minimizer of p with respect to w_i.*

PROOF. Clearly, there is a minimizer X of p with respect to w_i that contains only requested or initial points. This minimizer contains only points in the interval $[a, b]$. Assume that there is a point $x \in X$ in the interval $[a, p]$. What will happen if we slide x to a? The work function $w_i(X)$ can increase by at most $d(a, x)$ while the distance of x from p will increase by exactly $d(a, x)$. Therefore $X - x + a$ is also a minimizer of p. More precisely,

$$w_i(X - x + a) - \sum_{y \in X - x + a} d(p, y) \leq (w_i(X) + d(a, x)) - (\sum_{y \in X} d(p, y) + d(a, x))$$
$$= w_i(X) - \sum_{y \in X} d(p, y)$$

Similarly, we can slide all points of X to either a or b. If X has m points in $[a, p]$, then $a^m b^{k-m}$ is a minimizer of p. □

Theorem 3 *The* WFA *is k-competitive in the line.*

PROOF. We first show the somewhat simpler result that the WFA is k-competitive in an interval $[a, b]$ and then extend it to the infinite line.

We define a potential Φ_i to be the sum of w_i on all extreme configurations:

$$\Phi_i = \sum_{j=0}^{k} w_i(a^j b^{k-j}).$$

We will show that Φ_n is an upper bound (within a constant) of the extended cost. By Lemma 2, there is m such that $a^m b^{k-m}$ is a minimizer of r_{i+1} with respect to w_i. The increase of the potential, $\Phi_{i+1} - \Phi_i$, is equal to the increase of the work function on all extreme configurations. Since the work function increases monotonically, i.e., $w_{i+1}(X) \geq w_i(X)$, the increase $\Phi_{i+1} - \Phi_i$ of the potential is at least $w_{i+1}(a^m b^{k-m}) - w_i(a^m b^{k-m})$, which is the extended cost to service r_{i+1}. It follows, by telescoping, that the total extended cost, i.e., the sum of the extended cost for all requests, is bounded from above by $\Phi_n - \Phi_0$.

For a fixed interval $[a, b]$, the values of a work function cannot differ too much: for any work function w and any configurations X and Y: $w(X) - w(Y) \leq d(X, Y) \leq kd(a, b)$. This allows us to conclude that Φ_n is equal (within a constant) to $(k + 1)\text{OPT}(\rho_n) = (k + 1) \min_X\{w(X)\}$ and that Φ_0 is constant. The total extended cost is therefore bounded above by $(k+1)\text{OPT}(\rho_n) + \text{const}$ which implies the k-competitiveness of WFA.

We now turn to the case of the infinite line. The calculations of the previous paragraph are not sufficient for this case. The reason is that the definition of competitiveness requires that the term const in the expression $(k+1)\text{OPT}(\rho_n)+$ const is a constant independent of the request sequence. This is not the case when const depends on the length of the interval $[a, b]$. We can again assume that all requests are in an interval $[a, b]$ where a is the leftmost request of ρ_n and b is the rightmost one. But the interval $[a, b]$ is no longer independent of the request sequence.

We now calculate the difference $\Phi_n - \Phi_0$ more accurately and show that the additive term const is independent of $[a,b]$. We first compute the initial potential $\Phi_0 = \sum_{j=0}^{k} w(a^j b^{k-j}) = \sum_{j=0}^{k} d(a^j b^{k-j}, A_0)$. This is equal to $\frac{k(k+1)}{2} d(a,b) - \mathrm{cl}(A_0)$, where $\mathrm{cl}(A_0)$ is the sum of the distances between all pairs of points in A_0: $\mathrm{cl}(A_0) = \frac{1}{2}\sum_{a_1,a_2\in A_0} d(a_1,a_2)$. Similarly, if A_n is the final configuration of the optimal off-line algorithm, then $\Phi_n = \sum_{j=0}^{k} w(a^j b^{k-j}) \leq (k+1)w_n(A_n) + \sum_{j=0}^{k} d(a^j b^{k-j}, A_n) = (k+1)w_n(A_n) + \frac{k(k+1)}{2} d(a,b) - \mathrm{cl}(A_n)$. The only term that depends on $[a,b]$ is $\frac{k(k+1)}{2} d(a,b)$ and appears in both Φ_{n+1} and Φ_0. This is exactly what we want. We can now calculate $\Phi_n - \Phi_0 \leq (k+1)w_n(A_n) - \mathrm{cl}(A_n) + \mathrm{cl}(A_0) \leq (k+1)w_n(A_n) + \mathrm{cl}(A_0)$. Therefore the extended cost —which is bounded above by $\Phi_n - \Phi_0$— is at most $(k+1)\textsc{opt}(\rho_n) + \mathrm{const}$ and the additive term is indeed a constant that depends only on the initial configuration and not on the request sequence. $\square$

4 The WFA for weighted cache

It is well known that the problem of accessing pages in a weighted cache can be modeled by the k-server problem on weighted star instances (trees of depth 1). The leaves of the star represent pages and the leaves where servers reside correspond to the pages in the cache. The weight on the edge from the leaf to the center is half of the cost for fetching the corresponding page into the cache (since the server has to pay this cost twice per passing thru that leaf). The center of the star is denoted c. We show that WFA is k-competitive on such instances.

Recall that a minimizer of x is a configuration A that minimizes $m_i(A,x) = w_i(A) - \sum_{a\in A} d(a,x)$. It is easy to see that there is always a minimizer that does not include x (otherwise we can slide it away from x as in the case of the line in the previous section). Define $\mu_i(A,x)$ as follows:

$$\mu_i(A,x) = \begin{cases} w_i(A) - \sum_{a\in A} d(a,c) - d(c,x) & \text{if } x \notin A \\ w_i(A) - \sum_{a\in A-x} d(a,c) & \text{otherwise} \end{cases}$$

Alternatively we can define $\mu_i(A,x)$ to be equal to $m_i(A,x) + (k-1)d(c,x)$ (it is trivial to check that the two definitions are equivalent). It follows that a configuration A is a minimizer if and only if it minimizes $\mu_i(A,x)$.

Let the configuration of an adversary after request r_i be $U_i = \{u_1,\ldots,u_k\}$. We define:

$$\Phi(U_i, w_i) = \sum_{l=1}^{k} \min_A \mu_i(A, u_l).$$

Assume that the adversary services the next request r_{i+1} using the server at

u_j. The new adversary configuration is $U_i - u_j + r_{i+1}$. The next lemma bounds the change in Φ.

Lemma 4 *For any configuration U_i, any $u_j \in U_i$, and any r_{i+1}*

$$\Phi(U_i - u_j + r_{i+1}, w_i) - \Phi(U_i, w_i) \geq -d(u_j, r_{i+1}).$$

PROOF. Let A be an arbitrary configuration that does not contain r_{i+1}. We first show that there exists a configuration A' such that $\mu_i(A, r_{i+1}) \geq \mu_i(A', u_j) - d(u_j, r_{i+1})$.

If $u_j \notin A$ then let $A' = A$. We have

$$\begin{aligned}
\mu_i(A, r_{i+1}) &= w_i(A) - \sum_{a \in A} d(a, c) - d(c, r_{i+1}) \\
&\geq w_i(A) - \sum_{a \in A} d(a, c) - d(c, u_j) - d(u_j, r_{i+1}) \\
&= \mu_i(A', u_j) - d(u_j, r_{i+1}).
\end{aligned}$$

If $u_j \in A$ then let $A' = A - u_j + r_{i+1}$. We have

$$\begin{aligned}
\mu_i(A, r_{i+1}) &= w_i(A) - \sum_{a \in A} d(a, c) - d(c, r_{i+1}) \\
&= w_i(A) - \sum_{a \in A - u_j + r_{j+1}} d(a, c) - d(c, u_j) \\
&\geq w_i(A - u_j + r_{i+1}) - d(u_j, r_{i+1}) - \sum_{a \in A - u_j + r_{i+1}} d(a, c) - d(c, u_j) \\
&= \mu_i(A', u_j) - d(u_j, r_{i+1}).
\end{aligned}$$

It follows that

$$\Phi(U_i - u_j + r_{i+1}, w_i) - \Phi(U_i, w_i) = \min_A \mu_i(A, r_{i+1}) - \min_A \mu_i(A, u_j) \geq -d(u_j, r_{i+1}).$$

$\square$

Lemma 5 *For any configuration U_{i+1} that contains the last request r_{i+1} of w_{i+1}*

$$\Phi(U_{i+1}, w_{i+1}) - \Phi(U_{i+1}, w_i) \geq \max_X \{w_{i+1}(X) - w_i(X)\}.$$

PROOF. Let B be a minimizer of r_{i+1} with respect to w_i that does not contain r_{i+1}. Then by the Duality Lemma (Lemma 1), B is also a minimizer of r_{i+1} with respect to w_{i+1}. From the monotinicity property of work functions we have:

$$\mu_{i+1}(A, u_l) \geq \mu_i(A, u_l)$$

for all A and u_l. It follows that

$$\begin{aligned}
\Phi(U_{i+1}, w_{i+1}) - \Phi(U_{i+1}, w_i) &\geq \min_A \mu_{i+1}(A, r_{i+1}) - \min_A \mu_i(A, r_{i+1}) \\
&= w_{i+1}(B) - \left(\sum_{b \in B} d(b, c)\right) - d(c, r_{i+1}) \\
&\quad - \left[w_i(B) - \left(\sum_{b \in B} d(b, c)\right) - d(c, r_{i+1}) \right] \\
&= w_{i+1}(B) - w_i(B)
\end{aligned}$$

The proof is complete, since by the Duality Lemma:

$$w_{i+1}(B) - w_i(B) = \max_X \{w_{i+1}(X) - w_i(X)\}.$$

□

We can now combine the two above lemmata to get the main result of this section.

Theorem 6 *The work function algorithm is k-competitive for the weighted star.*

PROOF.

Let w_0, w_n be the initial and final work functions, and U_0, U_n be the initial and final adversary configurations respectively.

Let EXT and OPT denote the total extended cost and the optimal offline cost.

Combining Lemmas 4 and 5 we get that

$$\Phi(U_{i+1}, w_{i+1}) - \Phi(U_i, w_i) \geq \max_X \{w_{i+1}(X) - w_i(X)\} - d(u_j, u'_j),$$

where $u'_j = r_{i+1}$. The distance $d(u_j, u'_j) = d(U_i, U_{i+1})$ is the cost of the adversary to service r_{i+1}.

Summing for all requests and assuming that the adversary moves optimally, we get

$$\Phi(U_n, w_n) - \Phi(U_0, w_0) \geq \text{EXT} - \text{OPT}.$$

Since $\Phi_n = \Phi(U_n, w_n) \leq k \cdot w_n(U_n)$, and $\Phi_0 = \Phi(U_0, w_0) = -\mathrm{cl}(U_0)$ (the sum of the distances between all pairs of points in U_0), we obtain

$$\text{EXT} \leq \Phi_n - \Phi_0 + \text{OPT} \leq (k+1) \cdot \text{OPT} + \mathrm{cl}(U_0).$$

The total extended cost is bounded above by $k+1$ times the optimal cost plus a constant depending only on the initial configuration. We conclude that the work function algorithm is k-competitive for weighted star metric spaces. □

5 Metric spaces with $k+2$ points

In this section, we show that the k-server conjecture holds for metric spaces of $k+2$ points. This result was first shown in [7,10], but we give a simpler proof here. As in [7,10], instead of studying the k-server problem on $k+2$ points, it is simpler to consider the "dual" problem which is called the *2-evader problem*. In the 2-evader problem, 2 evaders occupy *distinct* points of a metric space M of $k+2$ points. The evaders respond to a sequence of ejections (requests) which is simply a sequence of points. If an evader occupies the point of an ejection, it has to move to some other point. The objective is to minimize the total distance traveled by the 2 evaders.

The 2-evader problem is equivalent to the k-server problem: servers occupy the points not occupied by evaders, and an ejection for the evaders is a request for the servers. This equivalence allows the theory of the k-server problem and in particular the notion of the extended cost and the Duality Lemma to be transfered to the evader problem. See [10] for a more extensive discussion of the evader problem and its equivalence to the k-server problem. The extended cost is again equal to the maximum increase of the work function. The corresponding Duality Lemma is:

Lemma 7 (Duality Lemma for the 2-evader problem) *Assume that $\{x, y\}$ minimizes the expression $w_i(x, y) + d(r_{i+1}, x) + d(r_{i+1}, y)$. Then $\{x, y\}$ minimizes also $w_{i+1}(x, y) + d(r_{i+1}, x) + d(r_{i+1}, y)$ and maximizes the extended cost:*

$$\max_{x,y}\{w_{i+1}(x, y) - w_i(x, y)\}.$$

As in the k-server problem, a configuration $\{x, y\}$ that minimizes $w_i(x, y) + d(p, x) + d(p, y)$ is called a minimizer of p with respect to w_i. Recall also that for the k-server problem there is always a minimizer for the k-server problem of a point x that does not contain x. Similarly, in the case of the 2-evader problem, there is a minimizer of a point x that contains x. In particular, such a minimizer of r_{i+1} is a configuration $\{r_{i+1}, x\}$ that minimizes $w_i(r_{i+1}, x) + d(r_{i+1}, x)$.

Using the Duality Lemma we are ready to prove the main theorem of this section. We will make use of the following notational convenience: whenever we write $w(x, y)$, we implicitly mean that x and y are distinct.

Theorem 8 *The* WFA *algorithm is k-competitive in every metric space of $k+2$ points.*

PROOF. The argument again is based on a potential. We want to find a potential Φ_i that "includes" a minimizer of r_{i+1}. It is easy to see that the potential $\hat{\Phi}_i = \sum_a \min_x\{w_i(a, x) + d(a, x)\}$ includes a minimizer of r_{i+1} and can be used to prove that the WFA algorithm is $(k + 1)$-competitive. This follows from $\hat{\Phi}_{i+1} - \hat{\Phi}_i \geq \min_x\{w_{i+1}(r_{i+1}, x) + d(r_{i+1}, x)\} - \min_x\{w_i(r_{i+1}, x) + d(r_{i+1}, x)\}$; by the Duality Lemma, the last expression is equal to the extended cost to service r_{i+1}. Clearly, the total extended cost is $\hat{\Phi}_n - \hat{\Phi}_0$. Since $\hat{\Phi}_n$ is within a constant from $(k + 2)\text{OPT}(\rho)$ and $\hat{\Phi}_0$ is constant, it follows that the WFA has competitive ratio at most $k + 1$.

How should we change $\hat{\Phi}_i$ to reduce the competitive ratio to k? Fix a configuration $\{b_1, b_2\}$ which minimizes $w(x, y) + d(x, y)$. The crucial observation is that $\{b_1, b_2\}$ is a minimizer of both b_1 and b_2. Thus, the number of distinct minimizers is at most $k+1$. Equivalently, even if we subtract $\min_{x,y}\{w_i(x, y)+d(x, y)\}$ from $\hat{\Phi}_i$, the resulting expression still contains a minimizer for every point and in particular of r_{i+1}. This suggests the following potential:

$$\Phi_i = \sum_a \min_x\{w_i(a, x) + d(a, x)\} - \min_{x,y}\{w_i(x, y) + d(x, y)\}. \tag{1}$$

Notice that $\Phi_i = \sum_{a \neq b_1} \min_x\{w_i(a, x) + d(a, x)\} = \sum_{a \neq b_2} \min_x\{w_i(a, x) + d(a, x)\}$. Since b_1 and b_2 are distinct, at least one of them is not equal to r_{i+1}; without loss of generality, say $b_1 \neq r_{i+1}$. By expressing

$$\Phi_i = \sum_{a \neq b_1} \min_x\{w_i(a, x) + d(a, x)\},$$

we observe that the sum includes the term corresponding to r_{i+1}. For the potential Φ_{i+1}, we also get

$$\begin{aligned}
\Phi_{i+1} &= \sum_a \min_x\{w_{i+1}(a, x) + d(a, x)\} - \min_{x,y}\{w_{i+1}(x, y) + d(x, y)\} \\
&\geq \sum_a \min_x\{w_{i+1}(a, x) + d(a, x)\} - \min_y\{w_{i+1}(b_1, y) + d(b_1, y)\} \\
&= \sum_{a \neq b_1} \min_x\{w_{i+1}(a, x) + d(a, x)\}.
\end{aligned}$$

Therefore, by subtracting, we get $\Phi_{i+1} - \Phi_i \geq \min_x\{w_{i+1}(r_{i+1}, x) + d(r_{i+1}, x)\} - \min_x\{w_i(r_{i+1}, x) + d(r_{i+1}, x)\}$ which is equal to the extended cost to service

r_{i+1}. By applying to Φ_i the same argument we used for $\hat{\Phi}_i$, we establish that the WFA algorithm is k-competitive. $\square$

There is an important difference between the expressions in the potential of the above proof and the potential in the proof in [10] —which is equal to the weight of the minimum spanning tree of the graph with weights $w_i(x, y) + d(x, y)$. The potential here involves a max operator (the minus min part of (1)) while the potential of [10] has only a min operator.

6 Conclusions

We showed that the WFA algorithm is k-competitive for the line, the weighted cache and for all metric spaces of $k + 2$ points. In all cases, we exploited the fact that the number of different minimizers is $k+1$ —this was explicit in the case of the line and metric spaces of $k + 2$ points and implicit in the proof of the weighted cache. This suggests that it may be worth investigating the cardinality of the set of minimizers for other special metric spaces, even for general metric spaces. Even if a metric space is guaranteed to have at most $k + 1$ minimizers, we don't know how to use this fact in general to establish that the WFA is k-competitive for this metric space. Is there a simple sufficient condition for this? Finally, as an intermediate step towards establishing the k-server conjecture, can we show that the WFA is k-competitive for trees?

References

[1] Yair Bartal and Edward Grove. The Harmonic k-server algorithm is competitive. *Journal of the ACM*, 47(1):1–15,2000.

[2] Allan Borodin and Ran El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

[3] Marek Chrobak, Howard Karloff, Tom H. Payne, and Sundar Vishwanathan. New results on server problems. *SIAM Journal on Discrete Mathematics*, 4:172–181, 1991.

[4] Marek Chrobak and Lawrence L. Larmore. An optimal online algorithm for k servers on trees. *SIAM Journal on Computing*, 20:144–148, 1991.

[5] Marek Chrobak and Lawrence L. Larmore. The server problem and online games. In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, volume 7, pages 11–64, 1992.

[6] Amos Fiat, Yuval Rabani, and Yiftach Ravid. Competitive k-server algorithms. *Journal of Computer and System Sciences*, 48:410–428, 1994.

[7] Elias Koutsoupias. *On-line algorithms and the k-server conjecture*. PhD thesis, University of California, San Diego, La Jolla, California, June 1994.

[8] E. Koutsoupias. Weak adversaries for the k-server problem. In *Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, New York City, NY, pages 444–449, 17–19 October 1999.

[9] Elias Koutsoupias and Christos Papadimitriou. On the k-server conjecture. *Journal of the ACM*, 42(5):971–983, September 1995.

[10] Elias Koutsoupias and Christos Papadimitriou. The 2-evader problem. *Information Processing Letters*, 57(5):249–252, March 1996.

[11] Mark Manasse, Lyle A. McGeoch, and Daniel Sleator. Competitive algorithms for online problems. In *Proc. 20th Symp. Theory of Computing*, pages 322–333, 1988.