A Decomposition Theorem and Bounds for Randomized Server Problems

Avrim Blum [*] Howard Karloff [†] Yuval Rabani [‡] Michael Saks [§]

Abstract

We prove a lower bound of $\Omega\left(\sqrt{\log k/\log\log k}\right)$ for the competitive ratio of randomized algorithms for the k-server problem against an oblivious adversary. The bound holds for arbitrary metric spaces (of at least $k+1$ points) and provides a new lower bound for the metrical task system problem as well. This improves the previous best lower bound of $\Omega(\log\log k)$ for arbitrary metric spaces [KRR], more closely approaching the conjectured lower bound of $\Omega(\log k)$. We also prove a lower bound of $\Omega\left(\frac{\log k}{\log\log k}\right)$ for the server problem on $k+1$ equally-spaced points on a line, which corresponds to some natural motion-planning problems.

Our results are deduced from a general decomposition theorem for a simpler version of both the k-server and the metrical task system problems, which we call the "pursuit-evasion game." We show that if a metric space $\mathcal{M}$ can be decomposed into two spaces $\mathcal{M}_L$ and $\mathcal{M}_R$, both very far away from each other, then the randomized competitive ratio for this game on $\mathcal{M}$ can be expressed nearly exactly in terms of the ratios on each of the two subspaces. This provides a natural way of analyzing the competitive ratio of a complex space made up of several regions in terms of the ratios of its components.

1 Introduction and Main Results

Online computation is a setting in which randomization can be shown to have a provable advantage over determinism. Many examples of this have appeared in the literature (see, e.g., [BLS, FKLMSY, MS, KMMO, Ira, Vis, BRS, FFKRRV]). One well-studied on-line problem is the k-server problem. Here an algorithm controls k servers, each of which is at some point in a metric space $\mathcal{M}$. At each time step the algorithm is given a *request*, which is a point in $\mathcal{M}$, and must serve it by moving a server to that point if none is there already. The algorithm is charged a cost equal to the total distance moved. The measure of success used is the *competitive ratio* [ST, KMRS]: roughly, this is the worst-case ratio of the cost charged to the algorithm on a request sequence, to the optimal offline cost of servicing that sequence (the optimal cost had it known the entire sequence in advance). For randomized algorithms, there are several natural notions for how requests are generated (see, e.g., [BBKTW], [RS]), and in this paper we consider requests generated by the weakest, *oblivious* adversary. This is an adversary that chooses the request sequence in advance while knowing the algorithm but not the random choices the algorithm makes. Intuitively, this can be interpreted by saying that after each successive request, the adversary "knows" only a probability distribution over states of the algorithm rather than the precise state.

It has been shown that for any metric space of at least $k+1$ points no deterministic online algorithm can achieve a competitive ratio better than k [MMS] (note that the problem is nontrivial only if there are at least $k+1$ points). The well-known k-server conjecture [MMS] says that for any metric space, there is a deterministic online algorithm that can achieve a competitive ratio of k. However, the best upper bounds known for general spaces, both for randomized and deterministic algorithms, are all exponential in k ([FRR],[Gro]).

The power of randomization in this setting was first demonstrated for the *uniform* metric space on $k+1$ points in which all pairs of distinct points are equidistant. For this space there is an $O(\log k)$-competitive algorithm, and indeed this is a lower bound. In fact, the infimum of the competitive ratios of all randomized algorithms, the *randomized competitive ratio*, is known to be exactly $1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{k} \sim \ln k$

*School of Computer Science, Carnegie Mellon University. E-mail: avrim@theory.cs.cmu.edu. Supported by an NSF Post-doctoral Fellowship.

†Department of Computer Science, University of Chicago. E-mail: howard@cs.uchicago.edu. This author was supported in part by NSF grant CCR 9107349 and by DIMACS.

‡Dept. of Computer Science, Tel-Aviv University. E-mail: rabani@math.tau.ac.il. Work partially done while visiting DIMACS.

§Dept. of Mathematics, Rutgers University and Dept. of Computer Science and Engineering, UCSD. E-mail: saks@math.rutgers.edu. This work was supported in part by NSF grant CCR89-11388 and AFOSR grants 89-0512 and 90-0008.

0-8186-2900-2/92 $3.00 © 1992 IEEE

[FKLMSY, MS, BLS]. Various people have speculated on the following conjecture:

Conjecture 1.1 *For any k and any metric space $\mathcal{M}$ on more than k points, the randomized competitive ratio of the k-server problem on $\mathcal{M}$ is $\Theta(\log k)$.*

Unlike the deterministic case where the lower bound is relatively easy and only the upper bound seems difficult, neither bound has been proved in the randomized case. For the lower bound, the previous best result is given by

Theorem 1.2 ([KRR]) *Let k be a positive integer and $\mathcal{M}$ be a metric space with at least $k+1$ points. Then the randomized competitive ratio of the k-server problem on $\mathcal{M}$ is $\Omega(\min\{\log k, \log\log|\mathcal{M}|\})$.*

If $\mathcal{M}$ is sufficiently large (exponential in k) then the lower bound in Conjecture 1.1 holds. For arbitrary spaces, however, in particular those whose size is polynomial in k, the lower bound obtained is $\Omega(\log\log k)$. One of the main results of this paper is to improve this $\Omega(\log\log k)$ bound to $\Omega\left(\sqrt{\frac{\log k}{\log\log k}}\right)$.

Theorem 1.3 *For any metric space $\mathcal{M}$ with at least $k+1$ points, the k-server problem on $\mathcal{M}$ has randomized competitive ratio $\Omega(\sqrt{\frac{\log k}{\log\log k}})$.*

The competitive ratio of the k-server problem for a space $\mathcal{M}$ is at least as large as the ratio on any subspace, as the adversary can restrict its requests to that subspace. Thus, to prove a lower bound on the competitive ratio of the k-server problem, it suffices to restrict attention to the case that the space has exactly $k+1$ points. One way to view this special case is to think of the algorithm as occupying a single point of the space (corresponding to the unique location where there is no server) and to think of the adversary as probing points of this space. When the adversary probes the point on which the algorithm stands, the algorithm must move to a different location. We call this the ***pursuit-evasion (PE) game*** and call the adversary the Pursuer and the algorithm the Evader. This paper is about the pursuit-evasion game. Using new analytic tools to study this game, we obtain lower bounds on the randomized competitive ratios of the k-server problem for *all* metric spaces, not just for those with exponentially many points. It should be noted that the pursuit-evasion game bears some resemblance to the cat-and-mouse game of [CDRS]. However, that game models the case of randomized algorithms against a more powerful adversary.

The pursuit-evasion game has applications to robotics. Imagine a robot walking down a long hallway of some width n (e.g., if $n=3$ then the robot may walk either down the left side, the center, or the right side of the hallway). The hallway contains rectangular obstacles, and when the robot meets an obstacle, it must go left or right around it. Any algorithm at all must travel the length of the hallway, so we will not charge for that. Instead we look at the left/right motion of the robot and compare it to the least possible left/right motion by an algorithm that knew the placement of the obstacles in advance. If the hallway has width n, then this is the pursuit-evasion game for the metric space of n equally-spaced points on the real line, a metric space we call $\mathcal{L}(n)$. The above lower bound of $\Omega(\sqrt{\frac{\log(n-1)}{\log\log(n-1)}})$ applies, of course, but for this case we have a better lower bound.

Theorem 1.4 *The pursuit-evasion game on $\mathcal{L}(n)$ has randomized competitive ratio $\Omega(\frac{\log n}{\log\log n})$. Thus, if $n > k$ the k-server problem on $\mathcal{L}(n)$ has competitive ratio $\Omega(\frac{\log k}{\log\log k})$.*

This nearly matches the conjectured bounds. The best randomized algorithm known for the line $\mathcal{L}(n)$ (see [BRS]) has competitive ratio $2^{O(\sqrt{\log n \log\log n})}$.

For general spaces $\mathcal{M}$, the special case of Conjecture 1.1 with $k = |\mathcal{M}|-1$ can be stated as:

Conjecture 1.5 *For any metric space $\mathcal{M}$ on n points, the randomized competitive ratio of the pursuit-evasion game on $\mathcal{M}$ is $\Theta(\log n)$.*

As noted, the lower bounds of Conjecture 1.5 and Conjecture 1.1 are equivalent. On the other hand, an upper bound for the pursuit-evasion game does not have immediate application to the upper bound conjecture for the general k-server problem. It is possible, however, that among all metric spaces with $k+1$ or more points, the largest randomized competitive ratios occur on those metric spaces with exactly $k+1$ points. In any case, we believe that a solution to the pursuit-evasion game would be a major step towards the solution of the more general problem and would also be interesting in its own right, particularly since there is currently no general upper bound for the randomized competitive ratio of the pursuit-evasion game that is better than the deterministic ratio of $|\mathcal{M}|-1$.

Previously, Conjecture 1.5 was known only for the case of uniform (or nearly uniform) spaces mentioned earlier. Here, we establish Conjecture 1.5 for highly "nonuniform" spaces. If $C > 1$, a metric space is C-*unbalanced* if for any three distinct points, the ratio of the largest distance to the smallest nonzero distance is at least C. For example, the metric space consisting of 4 points in a rectangle with side lengths 1 and C is C-unbalanced.

Theorem 1.6 *There is a polynomial $p(n)$ such that for all n, the pursuit-evasion problem on any $p(n)$-unbalanced metric space with n points has randomized competitive ratio between $\ln n$ and $3\ln n$.*

The proof will appear in the full version of the paper.

Finally, it is worth mentioning that bounds on the competitive ratio of the pursuit-evasion game carry over to the *task system* model of [BLS]. In particular, Theorems 1.3, 1.4 and 1.6 hold if we replace "the pursuit-evasion game on $\mathcal{M}$" by "the task system on $\mathcal{M}$."

1.1 Overview of the method

Theorems 1.3, 1.4 and 1.6 are proved as a consequence of a *decomposition theorem* for the competitive ratio of the pursuit-evasion game. (Henceforth, when we say "competitive ratio" we will mean the "randomized competitive ratio.") In words this theorem says that if $\mathcal{M}$ is a metric space that can be partitioned into two parts $\mathcal{M}_L$ and $\mathcal{M}_R$ such that the distance between the two sets is sufficiently large relative to the maximum distance within the sets, then the competitive ratio of the pursuit-evasion game on $\mathcal{M}$ can be expressed almost exactly in terms of the competitive ratios of the games on $\mathcal{M}_L$ and $\mathcal{M}_R$. That one can prove almost-tight upper *and* lower bounds on the competitive ratio is perhaps surprising. The precise statement is a bit long and is given as Theorem 1.8. A simpler version, useful when $\mathcal{M}_L$ and $\mathcal{M}_R$ have nearly the same competitive ratio, can be stated more easily, as follows. Here, let $\delta(\mathcal{M})$ be the diameter of the finite space $\mathcal{M}$ and let $\lambda(\mathcal{M})$ be its competitive ratio. (We use the convention that the competitive ratio of a one point space is 0.)

Theorem 1.7 *For any $\epsilon > 0$ there exists an increasing polynomial p such that if $(\mathcal{M}_L, \mathcal{M}_R)$ is a partition of metric space $\mathcal{M}$ with $\lambda(\mathcal{M}_R), \lambda(\mathcal{M}_L) \in [\lambda_{\min}, \lambda_{\max}]$ and $\frac{\delta(\mathcal{M})}{\max\{\delta(\mathcal{M}_L),\delta(\mathcal{M}_R)\}} \geq p(\lambda_{\max})$, then*

1. *$\lambda(\mathcal{M}) \leq \lambda_{\max} + 1 + \epsilon$.*
2. *$\lambda(\mathcal{M}) \geq \lambda_{\min} + 1 - \epsilon$.*

That is, if $\mathcal{M}_R$ and $\mathcal{M}_L$ have nearly the same competitive ratio λ, then the competitive ratio of $\mathcal{M}$ is very close to $1 + \lambda$. The decomposition theorem provides a tool for estimating the competitive ratio of the pursuit-evasion game on a space by partitioning it into smaller spaces and applying induction. For example, the simpler version provided above can be used to derive Theorem 1.4 as follows (this uses a known fact that if $\mathcal{N}$ is a subspace of $\mathcal{M}$, then $\lambda(\mathcal{N}) \leq \lambda(\mathcal{M})$.) Let $\epsilon = 1/2$ and let $p()$ be the increasing polynomial whose existence is given by Theorem 1.7. Let t be the greatest integer such that $\lceil p(\log n)\rceil^t \leq n$ and let $n' = \lceil p(\log n)\rceil^t$. Note that $t = \Omega(\frac{\log n}{\log\log n})$. Let $j = n'/\lceil p(\log n)\rceil = \lceil p(\log n)\rceil^{t-1}$ and let $\mathcal{M}_L$ and $\mathcal{M}_R$ be the leftmost j points and rightmost j points of $\mathcal{M} = \mathcal{L}(n')$ respectively. By choice of j, either $\lambda(\mathcal{M}_L) > \log n$ and we are done, or else the condition on δ of Theorem 1.7 is satisfied. So, $\lambda(\mathcal{L}(n)) \geq \lambda(\mathcal{L}(n')) \geq \lambda(\mathcal{M}_L \cup \mathcal{M}_R) \geq \lambda(\mathcal{M}_L) + 1/2$. We can continue on $\mathcal{M}_L$ letting $j' = j/\lceil p(\log n)\rceil$ and so forth until after t steps we have run out of points. The competitive ratio of $\mathcal{L}(n)$ is thus at least $t/2$, which is $\Omega(\frac{\log n}{\log\log n})$.

The full decomposition theorem provides sharp bounds on $\lambda(\mathcal{M})$ even if $\mathcal{M}_L$ and $\mathcal{M}_R$ have different competitive ratios. Let $Z(x)$ be the function on $x \geq 0$ which is equal to 1 if $x = 0$, and equal to $x/(e^x - 1)$ if $x > 0$. The theorem says that the competitive ratio of $\mathcal{M}$ is well approximated by $\max\{\lambda(\mathcal{M}_L), \lambda(\mathcal{M}_R)\} + Z(|\lambda(\mathcal{M}_L) - \lambda(\mathcal{M}_R)|)$. Here is the theorem in all gory detail.

Theorem 1.8 *Let $\mathcal{M}$ be a metric space with $|\mathcal{M}| \geq 3$ and let $(\mathcal{M}_R, \mathcal{M}_L)$ be a partition. Let $\alpha_R, \alpha_L \geq 0$ with $\alpha_{\max}$ their maximum and α_{diff} their absolute difference. Let $\delta = \delta(\mathcal{M})$ and $\delta_{\max} = \max\{\delta(\mathcal{M}_L), \delta(\mathcal{M}_R)\}$. Suppose that $\alpha_{\max} \geq 1$ and $\frac{\delta}{\delta_{\max}} \geq \max\{324\alpha_{\max}, \alpha_{\max}^3\}$. Then*

1. *if $\alpha_L \geq \lambda(\mathcal{M}_L)$ and $\alpha_R \geq \lambda(\mathcal{M}_R)$, then $\lambda \leq \alpha_{\max} + Z(\alpha_{\text{diff}})(1+\zeta)$, and*
2. *if $\alpha_L \leq \lambda(\mathcal{M}_L)$ and $\alpha_R \leq \lambda(\mathcal{M}_R)$, then $\lambda \geq \alpha_{\max} + Z(\alpha_{\text{diff}})(1-\zeta)$,*

where $\zeta = 23e^{\alpha_{\text{diff}}}\sqrt{\frac{\delta_{\max}}{\delta}\alpha_{\max}^3}$.

For the proof of Theorem 1.3, the cases of the decomposition theorem needed are the case above where $\mathcal{M}_L$ and $\mathcal{M}_R$ have nearly the same competitive ratio, and the "highly unbalanced" case where $\mathcal{M}_L$ is large and $\mathcal{M}_R$ is a single point. [KRR] proved their lower bound using a Ramsey-type theorem for metric spaces: every metric space has a fairly large subspace that is highly structured, either roughly uniform or highly "superincreasing." We prove a similar type of theorem here: any metric space of n points must contain at least one of the following *three* objects: (A) a roughly uniform space of around $2^{\sqrt{\log n/\log\log n}}$ points, (B) two far-away spaces with small diameter and each having around $n/2^{\sqrt{\log n \log\log n}}$ points, or (C) one point very far from a subspace containing nearly all the rest (around $n - n/2^{\sqrt{\log n/\log\log n}}$ points).

The proof of Theorem 1.8 consists of two parts. We introduce and completely analyze a new game, called the *Walker-Jumper* game, which abstracts the essential elements of the analysis of a decomposed problem. Then we formally demonstrate that the competitive ratio of a decomposed problem can be tightly bounded by the competitive ratio of an associated Walker-Jumper game. The full proof is quite long and only part of it is given here.

In Section 3, we present an informal discussion of the proof to motivate the connection with the Walker-Jumper game. In Section 4 we define the Walker-Jumper game and state a theorem which gives its exact competitive ratio, and describe optimal strategies for each of the two players. In section 5, we give a precise statement of the lemma connecting the Walker-Jumper game to the decomposition theorem. The application of the decomposition theorem to prove Theorems 1.3 is described in Section 6.

2 Notation and Definitions

A metric space $\mathcal{M}$ consists of a set and a symmetric nonnegative distance function defined on all pairs of points, which is zero only on pairs of equal points and which satisfies the triangle inequality. We will abuse notation and use $\mathcal{M}$ to denote both the metric space and the underlying set. The underlying set is assumed to be finite unless otherwise noted. The associated metric is denoted by $d = d_{\mathcal{M}}$. The diameter of the space $\delta = \delta(\mathcal{M})$ is the maximum distance between any pair of points.

A sequence of points in a metric space is referred to as a *path* in the space. For a path $\mathbf{x}$ we define $|\mathbf{x}|$ to be the number of points in $\mathbf{x}$, and *length*($\mathbf{x}$) to be the sum of the distances between successive pairs of points in $\mathbf{x}$.

In the pursuit-evasion game there are two players, the Pursuer and the Evader, who play on some (finite) metric space $\mathcal{M}$. At the beginning of each move the Evader is at some point of the space (there is no requirement that the starting point be any particular fixed location). The Pursuer then probes some point of $\mathcal{M}$. If he picks the point the Evader occupies then the Evader must move to some other point, otherwise the Evader may stay where he is. The cost to the Evader in responding to a sequence of probes is the total distance he travels. We denote the pursuit-evasion game on $\mathcal{M}$ by PE($\mathcal{M}$).

The Pursuer is *oblivious* to the position of the Evader. Thus, a deterministic strategy for the Pursuer is just a finite probe sequence ρ from the metric space. A randomized strategy is a probability distribution $\tilde{\rho}$ over such sequences. A sequence σ is a *response path* for ρ if $|\sigma| = |\rho|$ and $\sigma_i \neq \rho_i$ for all i. We will think of a deterministic strategy for the Evader as a function A that maps probe sequences ρ to response paths $A(\rho)$ such that for any ρ and any point a, $A(\rho a)$ extends $A(\rho)$. A is called a *response algorithm.* A randomized strategy for the Evader is a distribution $\tilde{A}$ over response algorithms.

The cost function $C_A(\rho)$ is the total length of the path $A(\rho)$, and $C_{\tilde{A}}(\tilde{\rho})$ is the expectation of $C_A(\rho)$ with respect to the product distribution of $\tilde{A}$ and $\tilde{\rho}$. We define $C_{\mathrm{OPT}}(\rho)$ to be the minimum of $C_A(\rho)$ over all A, and $C_{\mathrm{OPT}}(\tilde{\rho})$ to be the expectation of $C_{\mathrm{OPT}}(\rho)$ with respect to the distribution $\tilde{\rho}$. We say that an algorithm $\tilde{A}$ is *c-competitive* if there exists a constant K such that for all ρ, $C_{\tilde{A}}(\rho) \leq c \cdot C_{\mathrm{OPT}}(\rho) + K$. The *competitive ratio* of a space $\mathcal{M}$ is the infimum over all c such that there exists a c-competitive algorithm. We denote the competitive ratio of $PE(\mathcal{M})$ by $\lambda(\mathcal{M})$. We adopt the convention that the competitive ratio of a 1-point space is 0.

Definition 2.1 *For $s > 0$, an* s-block *of $\mathcal{M}$ is a probe sequence with the property that its optimal cost is between $s - \delta$ and s, where δ denotes $\delta(\mathcal{M})$. Note that any probe sequence ρ can be parsed uniquely into subsequences $\rho_1\rho_2\cdots\rho_k$ where each successive ρ_i is chosen to be of minimal length subject to having optimal cost at least $s - \delta$, and the last has cost no more than $s - \delta$. All blocks except possibly the last are s-blocks. We refer to this as the* s-block partition *of ρ.*

3 Overview of the Proof of the Decomposition Theorem

The decomposition theorem concerns a space $\mathcal{M}$ partitioned into two subspaces, the *left space* $\mathcal{M}_L$ and the *right space* $\mathcal{M}_R$. $\mathcal{M}$ has the property that the diameter δ of $\mathcal{M}$ is much larger than the diameters δ_L and δ_R of $\mathcal{M}_L$ and $\mathcal{M}_R$ respectively. In the present discussion, we assume that each space consists of at least two points; the degenerate case that one of the spaces consists of a single point will require some special treatment, which we omit from this abstract.

We want to express the competitive ratio λ of the big space in terms of the competitive ratios λ_L and λ_R of the two subspaces. The key idea is to abstract the behaviors of the Pursuer and the Evader so as to focus on their movements *between* the spaces, treating their movements within each space as a "black box." This idea leads to the formulation of a new game, called the Walker-Jumper game. In this section we provide an informal discussion that leads naturally to the definition of the Walker-Jumper game. Throughout the discussion we make various plausible but unjustified assumptions and approximations, which are cleaned up in the proof.

At each step, the Evader is either in $\mathcal{M}_L$ or in $\mathcal{M}_R$. While the Pursuer probes the opposite space, the Evader need do nothing. While the Pursuer probes the subspace occupied by the Evader, it seems apparent that either the Evader should follow his optimal randomized response algorithm for that space (achieving, over that interval of moves, a competitive ratio equal to that for that space) or he should move to the other space. By randomizing his choice of when to switch between spaces, he can hope to "fool" the Pursuer as to his location.

We view the probe sequence of the Pursuer as a sequence of left phases and right phases, where a left or right phase consists of probes only into the corresponding space. When the Evader uses a randomized strategy, the Pursuer only has a probability distribution on the location of the Evader. To maximize the competitive ratio, the Pursuer wants to construct a probe sequence that (i) has a good chance of catching the Evader often, and (ii) has a low offline cost. For the first goal, it would seem that he should always probe on the side with higher probability of containing the Evader, while for the second goal, it would seem that he would do well *not* to switch between spaces too often (this will make it easier for an offline algorithm to "hide" safely in one space for long intervals of moves) and thus make each phase long.

Recall that for $s > 0$, an s-block for a metric space is a probe sequence whose optimal response cost is approximately s. For some large integer D (which we will not specify at this point) we define $s = \delta/D$ and define ***left blocks*** and ***right blocks*** to be probe sequences which are s-blocks with respect to spaces $\mathcal{M}_L$ and $\mathcal{M}_R$ respectively. Note that an s-block for $\mathcal{M}_L$ or $\mathcal{M}_R$ is not an s-block with respect to the entire space. In fact, the optimal cost is 0 since an offline Evader may just stay at one location on the opposite space. Since δ is much larger than δ_L and δ_R, we may choose s so that $\lambda_L\delta_L$ and $\lambda_R\delta_R$ are much less than s, which in turn is much less than δ.

For a given probe sequence of $\mathcal{M}$, let us parse each left phase according to its s-block partition with respect to $\mathcal{M}_L$ and parse each right phase similarly. It is reasonable to expect that if s is small relative to the typical cost of a phase then we may ignore the "remainder" block of the phase and simply assume that each left (right) phase is a concatenation of left (right) blocks.

With this assumption, we view the entire probe sequence as a sequence of left and right blocks. If the Pursuer chooses to add a right block, the Pursuer can force the Evader to pay roughly $s\lambda_R$ if the Evader is on the right since λ_R is the competitive ratio of $\mathcal{M}_R$. (For this to work, we needed to bound from both above and below the additive constant in the definition of the competitive ratio. This is accomplished by Lemmas 5.2 and 5.3, and motivates the choice of s to be much larger than $\lambda_R\delta_R$ and $\lambda_L\delta_L$.) Similarly, when the Pursuer picks a left block, the Pursuer can force an Evader who stays on the left to pay roughly $s\lambda_L$.

To summarize, when the Pursuer adds a block, if the Evader is on the opposite side he pays nothing. If the Evader is on the same side, he either moves to the other side immediately, paying roughly δ, or he stays on the original side and incurs an expected cost of $s\lambda_R$ or $s\lambda_L$ depending on the side. We assume that δ is much larger than both $s\lambda_R$ and $s\lambda_L$, so that it would not pay for him to move to the opposite side at the beginning and back at the end the block. Thus, we have a good approximation to the cost of each block to the Evader, which depends on *only* (i) the side from which each block is chosen by the Pursuer and (ii) the side on which the Evader finishes responding to each block.

We'd like to get a similar estimate for the offline cost. For each probe sequence ρ, define $C_L(\rho)$ to be the minimum cost of a response sequence σ whose last point is on the left, and define $C_R(\rho)$ in the analogous way. We refer to these respectively as the left-optimal and right-optimal cost of ρ. The optimal cost of ρ is just the minimum of these. Since $C_L(\rho)$ and $C_R(\rho)$ differ by at most a constant (the diameter of $\mathcal{M}$), we may take $C_R(\rho)$ as a good estimate of $C_{\mathrm{OPT}}(\rho)$. We want to understand how C_L and C_R change when the Pursuer adds a left block or a right block. Let ρ be a sequence and consider adding a right block β to ρ. It is easy to see that $C_L(\rho\beta) = C_L(\rho)$, since the offline Evader who ends the previous block on the left can just stay where he is. On the other hand, $C_R(\rho\beta) \approx \min\{C_L(\rho)+\delta, C_R(\rho)+s\}$, since the Evader may either finish the previous block on the left and pay δ to move right at the end of β or finish the previous block on the right and respond to β on the right, paying about s. (This again uses that $s \gg \delta_L, \delta_R$ so that the optimal cost of an s-block is actually near s.)

Let us summarize this discussion by considering the evolution of the parameter $w = (C_R - C_L)/s$. Note that w is always between $-D$ and D (remember, $\delta = Ds$). Each right block increases w by (roughly) 1 subject to $w \leq D$, and similarly, each left block decreases w by 1, subject to $w \geq -D$. We can thus visualize the evolution of w as a walk on the integer points between $-D$ and D, with a right step corresponding to an increase in C_R by s and a left step corresponding to an increase in C_L by s.

Notice that if the Pursuer adds a right block that causes w to reach D, then the Pursuer may add an arbitrary number of right blocks without affecting C_L, C_R or w. Let us assume that he does this, i.e., a right step from $D-1$ to D corresponds to a huge number of right blocks. The effect of this on the Evader is clear: on such a step he must move to the left space (if he is not there already) or incur a huge cost. (Note that the Evader can compute C_R, C_L and w online and thus recognize this situation.) Similarly we may assume that a left step from $-D+1$ to $-D$ corresponds to a huge number of left blocks and the Evader must move to the right space.

We now can make the following estimates of the offline cost and the Evader's cost. The offline cost is about s times the number of steps of w to the right. The Evader's cost is 0 on any step taken by w in the direction opposite the space the Evader occupies. On a step taken in the direction of the Evader's space, the Evader's cost is $\delta = sD$ if he chooses to move to the other space and is $s\lambda_R$ or $s\lambda_L$ (depending on his space) if he doesn't move. Whenever a right (left) step is taken by w that reaches D $(-D)$, the Evader must move to the left (right) space.

This idealization suggests that we can model our problem by a game between two players, the *Walker* who walks on the line (and corresponds to the Pursuer) and the *Jumper* who jumps between the left and right side (and corresponds to the Evader). In the next section we define this game precisely and analyze it.

4 Walker-Jumper games

The Walker-Jumper game $WJ[D, \alpha_R, \alpha_L]$ has parameters D (a positive integer) and nonnegative real numbers α_R and α_L. The players are referred to as the Walker (Wendy) and the Jumper (Jack). The game "board" is the set $I_D = \{-D, -D+1, \ldots, D-1, D\}$. At each integer time $t \geq 0$, Wendy is at some position $w_t \in I_D$ and Jack is at some position $j_t \in \{-D, D\}$. The initial position for Wendy is $w_0 = 0$ and Jack can choose either $j_0 = -D$ or $j_0 = D$.

At each time t, a legal move for Wendy consists of one step either to the left ($w_t = w_{t-1} - 1$) or to the right ($w_t = w_{t-1} + 1$). A deterministic strategy for Wendy is a sequence $\mathbf{w} = (w_0, w_1, \ldots)$ of such w_t's, with $w_0 = 0$.

Jack's answer to request w_t is either to stay where he is ($j_t = j_{t-1}$) or to *jump* to his other allowed position ($j_t = -j_{t-1}$), subject only to the constraint that if Wendy arrives at Jack's location ($w_t = j_{t-1}$), then Jack *must* jump ($j_t = -w_t$). Formally, a deterministic algorithm for Jack is a function A that maps each finite request sequence $\mathbf{w} = (0, w_1, \ldots, w_s)$ to a sequence $(j_0, j_1, j_2, \ldots, j_s)$ in $\{-D, D\}^{s+1}$ satisfying this constraint.

We define the sequence Δw by $(\Delta w)_i = w_i - w_{i-1}$ and refer to Δw_i as the *direction* of move i. If j_t has the same sign as $(\Delta w)_t$, i.e., Wendy's move at time t brought her closer to the location that Jack occupies after his move, then we say that Wendy *hit* Jack. It is a *left hit* or a *right hit* depending on the direction of Wendy's move.

The cost function for algorithm A on request sequence $\mathbf{w}$, $C_A(\mathbf{w})$ (representing the cost to Jack), is given as follows.

1. Each jump by Jack costs him D.

2. Each right hit by Wendy costs Jack α_R.

3. Each left hit by Wendy costs Jack α_L.

As usual, we also consider randomized strategies, i.e., probability distributions over deterministic strategies, for both players. If $\tilde{A}$ and $\tilde{\mathbf{w}}$ are randomized strategies then $C_{\tilde{A}}(\tilde{\mathbf{w}})$ is defined to be the expectation of $C_A(\mathbf{w})$ with respect to the distributions. A useful intuition in thinking about the randomized version of the game is to imagine that Jack always knows Wendy's position while Wendy knows only a probability distribution on Jack's position, which is determined by Jack's algorithm.

We are interested in the (randomized) competitive ratio $\lambda = \lambda(WJ[D, \alpha_R, \alpha_L])$ of this game with respect to a nonstandard yardstick, $C_{\text{BASE}}(\mathbf{w})$, the total number of Wendy's steps to the right. That is, we say an algorithm A is c-competitive if there exists a constant K such that for all $\mathbf{w}$, $C_A(\mathbf{w}) \leq c \cdot C_{\text{BASE}}(\mathbf{w}) + K$, and define λ to be the infimum over all c such that there exists a c-competitive algorithm. Note that $C_{\text{BASE}}(\mathbf{w})$ is always between $(|\mathbf{w}| - D)/2$ and $(|\mathbf{w}| + D)/2$. Hence we can think of $C_{\text{BASE}}(\mathbf{w})$ as roughly $|\mathbf{w}|/2$.

Let $\alpha_{\max} = \max\{\alpha_R, \alpha_L\}$. We will only be concerned with the game for $D \geq \alpha_{\max}$. To develop some intuition for this game, let us first show that $\alpha_{\max} \leq \lambda \leq \alpha_L + \alpha_R + 1$.

To see the lower bound consider the following pure strategy for Wendy (without loss of generality, assume that $\alpha_L \geq \alpha_R$). Wendy moves right D times to position D and then alternately moves between $D-1$ and D. Each time Wendy moves from D to $D-1$, Jack must start at $-D$ and pays $\alpha_{\max}$ (if he doesn't move) or D (if he does). After t steps by Wendy, Jack's cost is at least $\alpha_{\max}\frac{t-D}{2}$, which is at least $\alpha_{\max} C_{\text{BASE}}(\mathbf{w}) - \alpha_{\max} D$.

To see the upper bound of $\alpha_L + \alpha_R + 1$, consider the following pure strategy for Jack: never jump unless a jump is required (because Wendy arrives at the same location). It is easy to see that if Wendy takes t steps, then Jack jumps at most $1 + t/(2D)$ times. Also, on any t-step sequence for Wendy, she takes at most $t/2 + D$ steps in each direction. So, Jack's cost can be bounded above by $(\alpha_R + \alpha_L)(t/2 + D) + t/2 + D$.

Since $C_{\mathbf{BASE}}$ is within an additive constant of $t/2$, this implies the upper bound.

As we shall see, the trivial lower bound above is much closer to the truth than the trivial upper bound. The main result of this section is an *exact* expression for the competitive ratio $\lambda(WJ[D,\alpha_R,\alpha_L])$. Define $\beta_R = 1-\alpha_R/(2D)$ and $\beta_L = 1-\alpha_L/(2D)$.

Theorem 4.1 *For nonnegative real numbers α_R and α_L and positive integer $D \geq max\{\alpha_L,\alpha_R\}$, the competitive ratio λ of the game $WJ(D,\alpha_R,\alpha_L)$ is given by*

$$\lambda = \begin{cases} \alpha_R+\beta_R & \text{if } \alpha_R=\alpha_L, \\ \frac{\alpha_R\beta_L^{2D}-\alpha_L\beta_R^{2D}}{\beta_L^{2D}-\beta_R^{2D}} & \text{if } \alpha_R\neq\alpha_L. \end{cases}$$

By an elementary but tedious sequence of estimates (which is omitted from this abstract) and the definition of the function $Z(x)$ from the introduction, we obtain the following corollary.

Corollary 4.2 *For nonnegative real numbers α_R and α_L, define $\alpha_{\max} = max\{\alpha_R,\alpha_L\}$ and $\alpha_{\text{diff}} = |\alpha_R-\alpha_L|$. Suppose that $\alpha_{\max}\geq 1$. Let $D\geq\alpha_{\max}^2$ be a positive integer. Then the competitive ratio λ of $WJ(D,\alpha_R,\alpha_L)$ satisfies*

$$\alpha_{\max}+Z(\alpha_{\text{diff}})(1-\epsilon)\leq\lambda\leq\alpha_{\max}+Z(\alpha_{\text{diff}})(1+\epsilon)$$

where $\epsilon=\frac{4\alpha_{\max}^2}{D}$.

4.1 The lower bound: a strategy for the Walker

Below is a simple randomized strategy for Wendy. The idea of the strategy is to force the expected cost to Jack to be essentially independent of what Jack does.

Walker Strategy

1. The strategy is defined in terms of two parameters σ_R and σ_L, which are real numbers between 0 and 1/2. If $w_t = D$ or $w_t=-D$ then w_{t+1} is forced. Otherwise $|w_t|<D$ and Wendy moves as follows. If move t was to the right ($w_t=w_{t-1}+1$) then Wendy goes left at step $t+1$ with probability σ_L and to the right with probability $1-\sigma_L$. Similarly if move t was to the left ($w_t=w_{t-1}-1$) then Wendy goes right at step $t+1$ with probability σ_R and to the left with probability $1-\sigma_R$.

2. The parameters σ_R and σ_L are defined by $\sigma_R = \alpha_L/(2D) = 1-\beta_L$ and $\sigma_L=\alpha_R/(2D)=1-\beta_R$.

To analyze this strategy, we first state a criterion for lower bounding the competitive ratio, which follows easily from the definition. Let $\mathbf{w^t}$ denote the first t terms of $\mathbf{w}$.

Proposition 4.3 *Let $\tilde{\mathbf{w}}$ be a distribution over infinite sequences for Wendy. Suppose that b is a positive real such that for any algorithm A for Jack and any positive integer t,*

$$C_A(\tilde{\mathbf{w}}^{\mathbf{t}})\geq t(b-g(t)),$$

where $g(t)$ tends to 0 as t tends to ∞. Then $\lambda\geq 2b$.

It is easy to show that we need consider only lazy Jumper algorithms (Jack never jumps when Wendy moves away from him), so fix a lazy deterministic strategy A for Jack and define $C(\mathbf{w}) = C_A(\mathbf{w})$. We are interested in bounding from below the expectation of $C(\mathbf{w})$ with respect to the above distribution for Wendy. It will be convenient to introduce a modified cost function, $C^*(\mathbf{w}) = C(\mathbf{w})+\psi(\mathbf{w})$, where the correction ψ is a "potential function" that depends only on the final step: if the last step by Wendy scored a hit on Jack (precisely, $j_s = D\cdot(w_s-w_{s-1})$) then $\psi(\mathbf{w}) = D-\alpha_R$ if the hit was to the right and $D-\alpha_L$ if the hit was to the left. Otherwise $\psi(\mathbf{w})=0$. Thus, $C(\mathbf{w})\geq C^*(\mathbf{w})-D$.

The purpose of introducing this modified cost is that with respect to this cost measure, the cost to Jack of any given step does not depend on what Jack does at that step. More precisely, define Jack's *modified cost at step t* to be $(\Delta C^*)_t(\mathbf{w}) = C^*(\mathbf{w^t})-C^*(\mathbf{w^{t-1}})$. Since we are assuming that Jack is following a lazy strategy, Jack has an option to move only if $j_{t-1} = D\cdot(\Delta w)_t$. (Otherwise, $(\Delta C^*)_t(\mathbf{w}) = 0$.) In this case, if he does not jump then his true cost increases by $\alpha_{(\Delta w)_t}$ while his modified cost increases by $\alpha_{(\Delta w)_t}+\psi(\mathbf{w^t})-\psi(\mathbf{w^{t-1}}) = D-\psi(\mathbf{w^{t-1}})$. If he jumps at time t, then his true cost goes up by D but his modified cost goes up by $D+\psi(\mathbf{w^t})-\psi(\mathbf{w^{t-1}}) = D-\psi(\mathbf{w^{t-1}})$, which is the same as if he did not jump. The reader can now check the following:

Lemma 4.4 *For any fixed lazy strategy A for Jack, the change in the modified cost at time t satisfies* $(\Delta C^*)_t(\mathbf{w}) =$

$$\begin{cases} \alpha_R \text{ if } ((\Delta w)_t,(\Delta w)_{t-1},j_{t-1}) & = & (+1,+1,+D) \\ 0 & = & (+1,+1,-D) \\ \alpha_R-D & = & (-1,+1,+D) \\ D & = & (-1,+1,-D) \\ \alpha_L & = & (-1,-1,-D) \\ 0 & = & (-1,-1,+D) \\ \alpha_L-D & = & (+1,-1,-D) \\ D & = & (+1,-1,+D). \end{cases}$$

Recall that if $|w_{t-1}| = D$ then, by the rules of the game, it must be the case that $j_{t-1} = -w_{t-1}$, $(\Delta w)_{t-1} = w_{t-1}/D$, and $(\Delta w)_t = j_{t-1}/D$. In this case, $(\Delta C^*)_t(\mathbf{w}) = D$. If $|w_{t-1}|<D$ then when

Wendy moves at time t, her move can depend on $(\Delta w)_{t-1}$, the direction of her last move, but not on j_{t-1}, which she does not know. So we try a strategy for Wendy in which her probability of moving in each direction depends on the direction of her last move. This motivates condition 1 in the definition of the strategy in section 4.1. So consider a strategy satisfying this condition, with the σ_R and σ_L of the walker strategy of section 4.1 as yet unspecified. We now can write down an expression in terms of w_{t-1} and j_{t-1}, for the expectation (with respect to this randomized strategy) of the change of the modified cost at time t, for the case $|w_{t-1}| < D$. It is $(\Delta C^*)_t(\tilde{\mathbf{w}}) =$

$$\begin{cases} \alpha_R - D\sigma_L & \text{if } ((\Delta w)_{t-1}, j_{t-1}) = (+1,+D) \\ D\sigma_L & \qquad\qquad\qquad\qquad = (+1,-D) \\ \alpha_L - D\sigma_R & \qquad\qquad\qquad\qquad = (-1,-D) \\ D\sigma_R & \qquad\qquad\qquad\qquad = (-1,+D). \end{cases}$$

By selecting σ_L and σ_R so that this expectation is independent of j_{t-1} (i.e., $\alpha_R - D\sigma_L = D\sigma_L$ and $\alpha_L - D\sigma_r = D\sigma_r$), we obtain condition 2 of the strategy.

Having motivated Wendy's strategy, we now continue with its analysis. Using $\sigma_L = \alpha_R/(2D)$ and $\sigma_R = \alpha_L/(2D)$, the change in the expected modified cost at time t can now be written as $(\Delta C^*)_t(\tilde{\mathbf{w}}) =$

$$\begin{cases} D & \text{if } |w_{t-1}| = D \\ \alpha_R/2 & \text{if } |w_{t-1}| < D \text{ and } (\Delta w)_{t-1} = +1 \\ \alpha_L/2 & \text{if } |w_{t-1}| < D \text{ and } (\Delta w)_{t-1} = -1\ . \end{cases}$$

To apply Proposition 4.3, we need to lower-bound Jack's expected cost against the sequence generated by the first j steps of Wendy's strategy. Wendy's strategy can be described by a Markov chain with state space $\{L_i : -D \le i < D\} \bigcup \{R_i : -D < i \le D\}$ where Wendy is in state L_i at time t if $w_t = i$ and $(\Delta_w)_t = -1$ (she is at point i and her last move was to the left) and Wendy is in state R_i at time t if $w_t = i$ and $(\Delta_w)_t = +1$ (she is at point i and her last move was to the right). For state U, let $N_t(U)$ denote the expected number of visits to state U during the first t steps by Wendy. Also, let $p(U)$ denote the steady state probability for state U. For large t, $N_t(U) = p(U)t(1 + o(1))$. Thus for $\tilde{\mathbf{w}}$ a sequence of t steps chosen according to Wendy's strategy,

$$\begin{aligned} C^*(\tilde{\mathbf{w}}) &= D[N_t(L_{-D}) + N_t(R_D)] + \\ &\quad \tfrac{1}{2}\textstyle\sum_{i=1-D}^{D-1}[N_t(L_i)\alpha_L + N_t(R_i)\alpha_R] \\ &= t(1+o(1)) \cdot [D(p(L_{-D}) + p(R_D)) + \\ &\quad \tfrac{1}{2}\textstyle\sum_{i=1-D}^{D-1}(p(L_i)\alpha_L + p(R_i)\alpha_R)]. \end{aligned}$$

Applying Proposition 4.3, we obtain

$$\begin{aligned} \lambda \ &\ge\ 2 \cdot [D(p(L_{-D}) + p(R_D)) + \\ &\quad \tfrac{1}{2}\textstyle\sum_{i=1-D}^{D-1}(p(L_i)\alpha_L + p(R_i)\alpha_R)]. \end{aligned} \tag{1}$$

We proceed to determine the steady state probabilities of the Markov chain. The transition matrix of the chain yields the following equations for the steady state probabilities:

$$\begin{aligned} p(L_i) &= (1-\sigma_R)p(L_{i+1}) + \sigma_L p(R_{i+1}) \\ &\quad \text{if } -D \le i < D-1, \\ p(L_{D-1}) &= p(R_D), \\ p(R_i) &= (1-\sigma_L)p(R_{i-1}) + \sigma_R p(L_{i-1}) \\ &\quad \text{if } 1-D < i \le D, \\ p(R_{1-D}) &= p(L_{-D}). \end{aligned}$$

Solving this system and recalling that $\beta_R = 1 - \alpha_R/(2D) = 1-\sigma_L$, and $\beta_L = 1-\alpha_L/(2D) = 1-\sigma_R$, we get the following solution (which can be easily checked against the defining equations):

$$\begin{aligned} p(L_i) &= \frac{J}{4D}\left(\frac{\beta_R}{\beta_L}\right)^i, \\ p(R_i) &= \frac{J}{4D}\left(\frac{\beta_R}{\beta_L}\right)^{i-1}, \end{aligned}$$

where

$$J = \begin{cases} 1 & \text{if } \alpha_R = \alpha_L, \\ \frac{(\alpha_R-\alpha_L)(\beta_L\beta_R)^D}{\beta_L(\beta_L^{2D}-\beta_R^{2D})} & \text{if } \alpha_R \ne \alpha_L \end{cases}$$

is chosen so that the sum of the probabilities is 1. Notice that $p(L_i) = p(R_{i+1})$ for all i and therefore $\sum_i p(L_i) = \sum_i p(R_{i+1}) = 1/2$. Thus we can rewrite the lower bound on λ from equation (1) as

$$\begin{aligned} \lambda &\ge 2 \cdot [D(p(L_{-D}) + p(R_D)) + \\ &\quad (\frac{1}{2} - p(L_{-D}))\frac{\alpha_L}{2} + (\frac{1}{2} - p(R_D))\frac{\alpha_R}{2}] \\ &= 2[D(\beta_L p(L_{-D}) + \beta_R p(R_D)) + \frac{\alpha_L+\alpha_R}{4}] \\ &= \frac{1}{2}[(\frac{J\cdot(\beta_L^{2D}+\beta_R^{2D})}{(\beta_L^{D-1}\beta_R^D)} + \alpha_L + \alpha_R]. \end{aligned}$$

It can now be checked that this simplifies to the expression in Theorem 4.1. ∎

4.2 The upper bound: a strategy for the Jumper

Below is an explicit description of a randomized strategy for Jack.

Jumper Strategy

1. The strategy is defined in terms of $2D+1$ parameters $0 = p_{-D} < p_{-D+1} < p_{-D+2} < \cdots < p_{D-1} < p_D = 1$ which are specified below. Initially, Jack chooses his initial position to be $-D$ with probability p_0. At round $t+1$, if Wendy

moves in the direction away from j_t, Jack does not move. If $j_t = D$ and Wendy moves rightward from $w_t = i-1$ to $w_{t+1} = i$ then Jack moves to $-D$ with probability $\frac{p_i - p_{i-1}}{1-p_{i-1}}$. If $j_t = -D$ and Wendy moves leftward from $w_t = i+1$ to $w_{t+1} = i$ then Jack moves to D with probability $\frac{p_{i+1}-p_i}{p_{i+1}}$.

2. The parameters p_i which are used are given by:

$$p_i = \begin{cases} \frac{1}{2} + \frac{i}{2D} & \text{if } \alpha_R = \alpha_L, \\ \frac{\beta_L^i \beta_R^{2D-i} - \beta_R^{2D}}{\beta_L^{2D} - \beta_R^{2D}} & \text{if } \alpha_R \neq \alpha_L. \end{cases}$$

Due to space limitation, we only hint at the proof that this strategy guarantees the upper bound. The first part of the definition of the strategy was chosen so that the following lemma holds:

Lemma 4.5 *Each time that Wendy is at position i, Jack is at $-D$ with probability p_i.*

The particular choice of values for the p_i's was made so that the expected cost r_i to Jack for each "round trip" by Wendy from i to $i+1$ and back is independent of i. Choosing the p_i's as above, we get that for each i,

$$r_i = \begin{cases} \alpha_R + \beta_R & \text{if } \alpha_R = \alpha_L, \\ \frac{\alpha_R \beta_L^{2D} - \alpha_L \beta_R^{2D}}{\beta_L^{2D} - \beta_R^{2D}} & \text{if } \alpha_R \neq \alpha_L. \end{cases}$$

The upper bound follows from the fact that Wendy's cost is well approximated by the number of "round trips" she makes.

5 Connecting the WJ Game and the Decomposition Theorem

Recall the notation of the decomposition theorem: $\mathcal{M}$ is a metric space partitioned into subspaces $\mathcal{M}_L$ and $\mathcal{M}_R$. Their respective diameters and competitive ratios (for the pursuit-evasion game) are denoted $\delta, \delta_L, \delta_R$ and $\lambda, \lambda_L, \lambda_R$. Also, $\delta_{\max} = \max\{\delta_L, \delta_R\}$ and $\lambda_{\max} = \max\{\lambda_L, \lambda_R\}$. The overview of the proof in Section 3 developed the Walker-Jumper game as a rough model for the pursuit-evasion game on a partitioned space. We now make this connection precise.

Lemma 5.1 *Let $(\mathcal{M}_L, \mathcal{M}_R)$ be a partition of $\mathcal{M}$ such that $\frac{\delta}{\delta_{\max}}$ is at least 32. Let α_R and α_L be nonnegative numbers and $\alpha_{\max}$ be their maximum. Let D be an integer satisfying $\max\{2\lambda_{\max}+2, \sqrt{\frac{\delta}{\delta_{\max}}}\} \leq D \leq \frac{\delta}{4\delta_{\max}}$, and let $\eta = 6\delta_{\max}D/\delta$. Then*

1. If $\alpha_L \geq \lambda_L$ and $\alpha_R \geq \lambda_R$, then

$$\lambda \leq \lambda(WJ[D, \alpha_R, \alpha_L])(1+\eta),$$

2. If $\alpha_L \leq \lambda_L$ and $\alpha_R \leq \lambda_R$, then

$$\lambda \geq \lambda(WJ[D, \alpha_R, \alpha_L])(1-\eta).$$

The high-level idea of Lemma 5.1 is as follows. The Walker-Jumper game was created as an idealization of the pursuit-evasion game where the adversary is required to probe in s-block "chunks" on each subspace, and the algorithm can only move between spaces in between the s-blocks. It is further an idealization in supposing that the optimal cost of an s-block is exactly s, that the on-line cost of an s-block is exactly $s\lambda_L$ or $s\lambda_R$ (so that s drops out when computing the ratio), and that the cost of moving between $\mathcal{M}_L$ and $\mathcal{M}_R$ is exactly δ. Lemma 5.1 shows that one can remove this idealization with a fudge-factor of η. The proof is surprisingly involved, however, and is omitted from this abstract. To give a flavor of what is needed we state two key lemmas, possibly of independent interest, which are used to bound the additive constants in the pursuit-evasion game.

Lemma 5.2 *Let $\mathcal{M}$ be a metric space of diameter δ and let λ denote the competitive ratio of its pursuit-evasion game. Then there exists a randomized algorithm $\tilde{A}$ such that for any probe sequence ρ,*

$$C_{\tilde{A}}(\rho) \leq \lambda \cdot C_{\mathrm{OPT}}(\rho) + \delta\lambda.$$

Lemma 5.3 *Let $\mathcal{M}$ be a metric space and λ the competitive ratio of its pursuit-evasion game. For any $s > 0$ there is a distribution $\tilde{\rho}_{\mathrm{s}}$ on s-blocks such that for any response algorithm A,*

$$C_A(\tilde{\rho}_{\mathrm{s}}) \geq \lambda s - (2\lambda\delta).$$

6 Applications of the Decomposition Theorem

The proof of Theorem 1.4 was given in the introduction. Here we use Theorem 1.8 to prove 1.3. The proof of Theorem 1.6, which is relatively straightforward will appear in the full paper.

We need a few additional definitions. If $\mathcal{M}$ is a metric space and $\mathcal{N}$ a subspace, we say that $\mathcal{N}$ is an *ϵ-small* subspace (relative to $\mathcal{M}$) if $\delta(\mathcal{N}) \leq \epsilon \cdot \delta(\mathcal{M})$. We say that $\mathcal{N}$ is *ϵ-uniform* subspace if the distance between any two distinct points in $\mathcal{N}$ is at least $\epsilon \cdot \delta(\mathcal{N})$. The following easy consequence of the fact that the competitive ratio for a uniform space is between $\ln n$ and $2\ln n$ is left to the reader (see also [KRR]).

Lemma 6.1 *The competitive ratio of an ϵ-uniform space on n points is between $\epsilon \ln n$ and $(2/\epsilon)\ln n$.*

It will be convenient to state some special cases of Theorem 1.8. First, the special case where the lower bounds on λ_R and λ_L are the same is worth noting.

Corollary 6.2 *Let $\mathcal{M}$ be a metric space, and let $(\mathcal{M}_R, \mathcal{M}_L)$ be a partition into two subspaces each of size at least 2. Let $\beta \geq 1$ be a lower bound on both λ_R and λ_L and suppose that both spaces $\mathcal{M}_R$ and $\mathcal{M}_L$ are $(2200\beta^3)^{-1}$-small. Then $\lambda \geq \beta + 1/2$.*

We also have

Corollary 6.3 *Let $\mathcal{M}$ be a metric space of at least three points and let $\mathcal{N}$ be a subspace. Suppose that $\beta \geq 1$ is a lower bound on $\lambda(\mathcal{N})$, and $\mathcal{N}$ is $(27000e^{3\beta})^{-1}$-small. Then $\lambda(\mathcal{M}) \geq \beta + e^{-2\beta}$.*

This follows by applying the main decomposition theorem to the space $\mathcal{N}$ and the space consisting of the single point of maximum distance from $\mathcal{N}$.

Proof of Theorem 1.3:

Let Q be a positive constant to be determined and define $f(n) = \max\{1, Q\sqrt{\ln n/\ln\ln n}\}$. We prove by induction on n that the competitive ratio of the pursuit problem on any metric space $\mathcal{M}$ with n points is at least $f(n)$. The inductive argument will work for n greater than some n_0; we will then choose Q so that $Q\sqrt{\ln n_0/\ln\ln n_0} = 1$. Thus, the basis case $n \leq n_0$ will be trivial. We will also need $Q \leq 1/7$ for part of the inductive argument.

The key to the inductive step is the following structural result for metric spaces. Define the function $T(n) = e^{4f(n)}$.

Lemma 6.4 *There exists an integer n_0 so that any metric space on $n \geq n_0$ points has either:*

1. *a 1/4-uniform subspace of size at least $T(n)$,*
2. *a pair of disjoint subspaces each of size at least $j(n) = nT(n)^{-6\ln\ln n}$, such that each is $(2200 \ln^3 n)^{-1}$-small with respect to their union, or*
3. *a $(27000n^3)^{-1}$-small subspace $\mathcal{N}$ whose size is at least $n(1 - 1/T(n))$.*

Applying the lemma to $\mathcal{M}$, we can now finish the inductive step of the theorem. In the first case, we may use Lemma 6.1 to obtain a lower bound of $f(n)$ on the competitive ratio. If condition (2) holds, we apply Corollary 6.2 to the union of these two spaces. By induction, each space has competitive ratio at least

$$
\begin{aligned}
f(j(n)) &\geq Q\sqrt{\frac{\ln n - (6\ln\ln n)(4f(n))}{\ln\ln n}} \\
&\geq Q\left(\sqrt{\frac{\ln n}{\ln\ln n}} - \frac{24f(n)}{\sqrt{\frac{\ln n}{\ln\ln n}}}\right) \\
&\geq Q\sqrt{\frac{\ln n}{\ln\ln n}} - 24Q^2 \\
&\geq Q\sqrt{\frac{\ln n}{\ln\ln n}} - (1/2)
\end{aligned}
$$

where the second inequality follows from the fact that $A > B \geq 0$ implies $\sqrt{A - B} \geq \sqrt{A} - B/\sqrt{A}$, the third from the assumption that $n \geq n_0$, and the last from $Q \leq 1/7$. Taking the last quantity as β in Corollary 6.2 we obtain the required lower bound of $Q\sqrt{\frac{\ln n}{\ln\ln n}}$ for $\lambda(\mathcal{M})$.

Finally if condition (3) holds, then by induction, the subspace has competitive ratio at least

$$
\begin{aligned}
f(n(1 - 1/T(n))) &\geq Q\sqrt{\frac{\ln n + \ln(1 - 1/T(n))}{\ln\ln n}} \\
&\geq Q\sqrt{\frac{\ln n - 1/T(n)}{\ln\ln n}} \\
&\geq Q\sqrt{\frac{\ln n}{\ln\ln n}} - Q/T(n),
\end{aligned}
$$

where the last inequality follows again by using the fact that $A > B \geq 0$ implies $\sqrt{A - B} \geq \sqrt{A} - B/\sqrt{A}$. Taking the last quantity as β in Corollary 6.3 we obtain the lower bound $\lambda(\mathcal{M}) \geq f(n) - Q/T(n) + e^{-2f(n)} \geq f(n)$. ∎

So it now remains to prove Lemma 6.4. Let $\mathcal{M}$ be an arbitrary metric space on n points and assume that the first conclusion does not hold, i.e., there is no 1/4-uniform subspace of size $T(n)$.

Lemma 6.5 *Let k be a positive integer and r, s be positive real numbers. If $\mathcal{M}$ is a space that does not have a 1/4-uniform space of size at least s then there is a partition $(\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_j)$ of the space such that $\mathcal{M}_0$ has at most r points and each of the remaining spaces is a 2^{-k} − small subspace of size at least r/s^k.*

Proof: By induction on the number of points n in $\mathcal{M}$. For $n \leq r$ the result is trivial. To prove the inductive step it will suffice to show that any space of size at least r with no 1/4-uniform subspace of size s has at least one 2^{-k}-small subspace of size at least r/s^k, as we can then remove that space and finish by induction (since if $\mathcal{N}$ is ϵ-small with respect to a subspace, then it is ϵ-small with respect to the entire space).

To prove the existence of one such space, proceed by induction on k. Since the case $k = 0$ is trivial, suppose $k > 0$. By the inductive hypothesis there is a subspace $\mathcal{N}$ of size at least r/s^{k-1} and diameter at most $\delta/2^{k-1}$. Let x_1, x_2 be two points in $\mathcal{N}$ at distance $\delta(\mathcal{N})$, and let $x_1, x_2, \ldots, x_t$ be a maximal 1/4-uniform subspace of $\mathcal{N}$ containing those two points. By hypothesis, $t < s$. Furthermore, by the maximality of the set, every point in $\mathcal{N}$ is within distance $\delta(\mathcal{N})/4$ of one of the x_i. Classify each point of $\mathcal{N}$ according to the x_i to which it is closest (breaking ties arbitrarily). Each class has diameter at most $\delta(\mathcal{N})/2$ and so is $1/2^k$-small, and the largest such class has size at least r/s^k. ■

To complete the proof of Lemma 6.4, we apply Lemma 6.5 with $k = \lfloor 6 \ln\ln n \rfloor - 2$, $s = T(n)$ and $r = n/T(n)^2$. This implies that we can partition $\mathcal{M}$ into spaces $\mathcal{M}_0, \mathcal{M}_1, \ldots \mathcal{M}_j$ where $\mathcal{M}_0$ has size at most $n/T(n)^2$ and all of the others are $2^{-\lfloor 6\ln\ln n\rfloor + 2} \leq (4400 \ln^3 n)^{-1}$-small (this inequality holds for $n \geq n_0$, if n_0 is sufficiently large) and have size at least $\frac{n}{T(n)^{k+2}} \geq \frac{n}{T(n)^{6\ln\ln n}}$. Now if there is a pair of spaces $\mathcal{M}_h$ and $\mathcal{M}_i$ with $i, h > 0$ whose maximum distance is at least $\delta(\mathcal{M})/2$, then they satisfy condition (2) above. So assume no such pair exists. Then the space $\mathcal{Q}_1 = \mathcal{M} - \mathcal{M}_0$ has diameter at most $\delta(\mathcal{M})/2$ and size at least $n(1 - T(n)^{-2})$. Apply the same argument to $\mathcal{Q}_1$ keeping fixed the values for k, s, and r. We obtain either a pair of spaces that are $(2200 \ln^3 n)^{-1}$-small relative to their union and have size at least $\frac{n}{T(n)^{6\ln\ln n}}$, or a space $\mathcal{Q}_2$ of diameter at most $\delta(\mathcal{M})/4$ and size at least $n(1 - 2T(n)^{-2})$. Repeat this at most $p = \lceil \log S(n) \rceil$ times, stopping if condition (2) is satisfied. If not, we obtain a space $\mathcal{Q}_p$ of diameter at most $\delta(\mathcal{M})/S(n)$ and size at least $n(1 - \lceil \log S(n) \rceil / T(n)^2)$. Now we can choose n_0 so that for $n \geq n_0$, $\lceil \log S(n) \rceil \leq T(n)$. Then the size of $\mathcal{Q}_p$ is at least $n(1 - 1/T(n))$ and so we have condition (3). ■

References

[BBKTW] S. Ben-David, A. Borodin, R.M. Karp, G. Tardos, and A. Wigderson. On the Power of Randomization in Online Algorithms. In *Proc. of the 22nd Ann. ACM Symp. on Theory of Computing*, pages 379–386, May 1990.

[BLS] A. Borodin, N. Linial, and M. Saks. An Optimal On-Line Algorithm for Metrical Task Systems. In *Proc. of the 19th Ann. ACM Symp. on Theory of Computing*, pages 373–382, May 1987. To appear in *JACM*.

[BRS] A. Blum, P. Raghavan, and B. Schieber. Navigating in Unfamiliar Geometric Terrain. In *Proc. of the 23rd Ann. ACM Symp. on Theory of Computing*, pages 494–504, May 1991.

[CDRS] D. Coppersmith, P. Doyle, P. Raghavan, M. Snir. Random Walks on Weighted Graphs and Applications to On-line Algorithms. In *Proc. of the 22nd Ann. ACM Symp. on Theory of Computing*, pages 369–378, May 1990.

[FFKRRV] A. Fiat, D.P. Foster, H.J. Karloff, Y. Rabani, Y. Ravid, and S. Vishwanathan. Competitive Algorithms for Layered Graph Traversal. In *Proc. of the 32nd Ann. Symp. on Foundations of Comp. Sci.*, pages 288–297, October 1991.

[FKLMSY] A. Fiat, R.M. Karp, M. Luby, L.A. McGeoch, D.D. Sleator, and N.E. Young. Competitive Paging Algorithms. *Journal of Algorithms*, 12:685–699, 1991.

[FRR] A. Fiat, Y. Rabani, and Y. Ravid. Competitive k-Server Algorithms. In *Proc. of the 31st Ann. IEEE Symp. on Foundations of Computer Science*, October 1990.

[Gro] E. Grove. The Harmonic k-Server Algorithm is Competitive. In *Proc. of the 23nd Ann. ACM Symp. on Theory of Computing*, pages 260–266, May 1991.

[Ira] S. Irani. Two Results on the List Update Problem. Research report RC-15922, IBM T.J. Watson Research Center, 1990.

[KMMO] A.R. Karlin, M.S. Manasse, L.A. McGeoch, and S. Owicki. Competitive Randomized Algorithms for Non-uniform Problems. In *Proc. of the 1st Ann. ACM-SIAM Symp. on Discrete Algorithms*, pages 301–309, January 1990.

[KMRS] A.R. Karlin, M.S. Manasse, L. Rudolph, and D.D. Sleator. Competitive Snoopy Caching. *Algorithmica*, 3(1):79–119, 1988.

[KRR] H.J. Karloff, Y. Rabani, and Y. Ravid. Lower Bounds for Randomized k-Server Algorithms. In *Proc. of the 23rd Ann. ACM Symp. on Theory of Computing*, May 1991.

[MMS] M.S. Manasse, L.A. McGeoch, and D.D. Sleator. Competitive Algorithms for On-line Problems. *Journal of Algorithms*, 11:208–230, 1990.

[MS] L.A. McGeoch and D.D. Sleator. A Strongly Competitive Randomized Paging Algorithm. Submitted to *Algorithmica*.

[RS] P. Raghavan and M. Snir. Memory versus Randomization in On-Line Algorithms. In *Proc. 16th ICALP*, July 1989.

[ST] D.D. Sleator and R.E. Tarjan. Amortized Efficiency of List Update and Paging Rules. *Communication of the ACM*, 28(2): 202–208, 1985.

[Vis] S. Vishwanathan. Randomized Online Graph Coloring. In *Proc. of the 31st Ann. IEEE Symp. on Foundations of Computer Science*, pages 464–469, October 1990.