Matching on the Line Admits No $o(\sqrt{\log n})$-Competitive Algorithm

Enoch Peserico
Università degli Studi di Padova, Italy

Michele Scquizzato ✉
Università degli Studi di Padova, Italy

Abstract

We present a simple proof that the competitive ratio of any randomized online matching algorithm for the line exceeds $\sqrt{\log_2(n+1)}/15$ for all $n = 2^i - 1 : i \in \mathbb{N}$, settling a 25-year-old open question.

2012 ACM Subject Classification Theory of computation → Online algorithms

Keywords and phrases Metric matching, online algorithms, competitive analysis

Digital Object Identifier 10.4230/LIPIcs.ICALP.2021.103

Category Track A: Algorithms, Complexity and Games

Related Version *Extended Version*: `https://arxiv.org/abs/2012.15593`

Funding *Michele Scquizzato*: supported, in part, by Univ. Padova grant BIRD197859/19.

Acknowledgements We are indebted to Kirk Pruhs and the anonymous reviewers for their constructive criticism and insightful observations.

1 Online matching, on the line

In *online metric matching* [7, 9] n points of a metric space are designated as *servers*. One by one n *requests* arrive at arbitrary points of the space; upon arrival each must be matched to a yet unmatched server, at a cost equal to their distance. Matchings should minimize the ratio between the total cost and the *offline* cost attainable if all requests were known beforehand. A matching algorithm is $c(n)$*-competitive* if it keeps this ratio no higher than $c(n)$ for all possible placements of servers and requests.

It is widely acknowledged [1, 10, 14] that the line is the most interesting metric space for the problem. Matching on the line models many scenarios, like a shop that must rent to customers skis of approximately their height, where a stream of requests must be serviced with minimally mismatched items from a known store. Despite matching being specifically studied on the line since at least 1996 [8], no tight competitiveness bounds are known.

As for upper bounds, the line is a doubling space and thus admits an $O(\log n)$-competitive randomized algorithm [5]; a sequence of recent developments [1, 12, 13] yielded the same ratio without randomization. Better bounds have been obtained only by algorithms with additional power, such as that to re-assign past requests [6, 11] or predict future ones [2].

As for lower bounds, the competitive ratio is at least 4.591 for randomized algorithms and 9 for deterministic ones since the *cow-path* problem is a special case of matching on the line [8]. These bounds were conjectured tight [8] until a complex adversarial strategy yielded a lower bound of 9.001 for deterministic algorithms [4]. Beyond some $\Omega(\log n)$ bounds for restricted classes of algorithms [3, 10, 12], there has been no further progress on the lower-bound side before this work.

48th International Colloquium on Automata, Languages, and Programming (ICALP 2021).
Editors: Nikhil Bansal, Emanuela Merelli, and James Worrell; Article No. 103; pp. 103:1–103:3
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2 An $\Omega(\sqrt{\log n})$-competitiveness bound

We prove a simple $\Omega(\sqrt{\log n})$ lower bound on the competitive ratio of randomized online matching algorithms for the line.

For any $n = 2^i - 1$ with $i \in \mathbb{N}$ consider the $[0, n+1]$ interval; for each positive integer $j \leq n$ place a server at point j, and place n requests over $\log_2(n+1)$ rounds as follows. On the r^{th} round (for $1 \leq r \leq \log_2(n+1)$) partition the interval into $(n+1)/2^r$ subintervals of length 2^r, choose within each uniformly and independently at random an *origin* point, and place a request on the closest integer multiple of 2^{-n} breaking ties arbitrarily. "Discretizing" requests instead of directly using the corresponding origins prevents some technical difficulties – see our remark at the end.

We prove in Lemma 1 that the expected distance between the ℓ^{th} leftmost server and the ℓ^{th} leftmost origin is $O(\sqrt{\log n})$, so servers and requests can be matched with an expected offline cost $O(n\sqrt{\log n})$. Conversely, we prove in Lemma 2 that *any* online matching algorithm ALG incurs an expected $\Omega(n)$ cost in any given round, for a total cost $\Omega(n \log n)$. The two results can be combined to prove that on *some* request sequence ALG incurs $\Omega(\sqrt{\log n})$ times the offline cost.

▶ **Lemma 1.** *The expected distance between the ℓ^{th} leftmost origin and the ℓ^{th} leftmost server is at most $\sqrt{\log_2(n+1)} + 3$.*

Proof. Let S_ℓ be ℓ^{th} leftmost server and g_ℓ be the number of origins to its left. Note that if g_ℓ equals respectively ℓ or $\ell - 1$, the ℓ^{th} origin is the first immediately to the left, or to the right of S_ℓ; and since the first round placed one origin in every subinterval of size 2, such an origin is within distance 3 of S_ℓ. By the same token, denoting by δ_ℓ the quantity $|g_\ell - (\ell - \frac{\ell}{n+1})|$, the ℓ^{th} leftmost origin is within distance $2\delta_\ell + 3$ of S_ℓ. Note that δ_ℓ is the absolute deviation from the mean of r_ℓ, since r_ℓ is the sum of n independent indicator random variables each denoting whether a given origin was placed to the left of S_ℓ, with total expectation $\frac{n}{n+1}\ell = \ell - \frac{\ell}{n+1}$ (by construction, the expected density of origins is constant throughout the main interval). At most one such variable in a given round has variance greater than 0, albeit obviously at most 1/4: that corresponding to the origin placed in a subinterval holding S_ℓ strictly in its interior. Adding the individual variances we obtain that the variance of r_ℓ, i.e. the expectation of δ_ℓ^2, is at most $\log_2(n+1)/4$; and since by Jensen's inequality $E[\delta_\ell] \leq E[\delta_\ell^2]^{\frac{1}{2}}$, the expected distance between S_ℓ and the ℓ^{th} leftmost origin is at most $\sqrt{\log_2(n+1)} + 3$. ◀

▶ **Lemma 2.** *Any randomized online matching algorithm incurs an expected cost greater than $(n+1)/12$ in each round.*

Proof. Consider an origin placed uniformly at random in a subinterval of size 2^r during the r^{th} round. Assume m unmatched servers in the interior points of that subinterval divide it into $m+1$ segments of (integer) length $d_0, \ldots, d_m$. Then the probability the corresponding *request* falls within a segment of length d is $d/2^r$, in which case the expected distance of the *request* from the segment's closer endpoint is $d/4$. Adding over all the s_r segments in all the round's subintervals, applying Jensen's inequality, and noting that s_r does not exceed the number of subintervals (i.e. $(n+1)/2^r$) plus the total number of unmatched servers (i.e. $(n+1)/2^{r-1} - 1$), the expected cost to service all requests in the round is at least:

$$\sum_{h=1}^{s_r} \frac{d_h}{4} \cdot \frac{d_h}{2^r} \geq \frac{1}{4 \cdot 2^r} s_r \left(\frac{n+1}{s_r}\right)^2 > \frac{(n+1)^2}{4 \cdot 2^r} \cdot \frac{2^r}{3(n+1)} = \frac{n+1}{12}.$$ ◀

We can then easily prove the following:

▶ **Theorem.** *The competitive ratio of any randomized online matching algorithm for the line exceeds* $\sqrt{\log_2(n+1)}/15$ *for all* $n = 2^i - 1 : i \in \mathbb{N}$.

Proof. Let $C_A(\sigma)$ be the expected cost incurred by a randomized online matching algorithm ALG on a request sequence σ, and $C_O(\sigma)$ the offline cost; and let p_σ be the probability of generating σ through the origin-request process described earlier. Since $\forall a_i, b_i > 0$ we have that $(\sum_i a_i)/(\sum_i b_i)$ is a convex linear combination of the individual ratios a_i/b_i, focusing on the case $\sqrt{\log_2(n+1)}/15 \geq 1$ for which $\sqrt{\log_2(n+1)} + 3 + 2^{-n} < (5/4)\sqrt{\log_2(n+1)}$:

$$\max_{\sigma : p_\sigma \neq 0} \frac{C_A(\sigma)}{C_O(\sigma)} \geq \frac{\sum\limits_{\sigma : p_\sigma \neq 0} C_A(\sigma) p_\sigma}{\sum\limits_{\sigma : p_\sigma \neq 0} C_O(\sigma) p_\sigma} > \frac{(n+1)\log_2(n+1)/12}{n(\sqrt{\log_2(n+1)} + 3 + 2^{-n})} > \frac{\sqrt{\log_2(n+1)}}{15}. \quad ◀$$

▶ Remark. Without discretized requests the term $\sum_{\sigma : p_\sigma \neq 0} C_A(\sigma) p_\sigma$ in the theorem's proof would have been an integral, potentially ill-defined (for example, if ALG serviced requests for rational points in an interval with one server and for irrational points with another).

References

1. Antonios Antoniadis, Neal Barcelo, Michael Nugent, Kirk Pruhs, and Michele Scquizzato. A $o(n)$-competitive deterministic algorithm for online matching on a line. *Algorithmica*, 81(7):2917–2933, 2019.
2. Antonios Antoniadis, Christian Coester, Marek Eliás, Adam Polak, and Bertrand Simon. Online metric algorithms with untrusted predictions. In *Proceedings of the 37th ICML*, pages 345–355, 2020.
3. Antonios Antoniadis, Carsten Fischer, and Andreas Tönnis. A collection of lower bounds for online matching on the line. In *Proceedings of the 13th LATIN*, pages 52–65, 2018.
4. Bernhard Fuchs, Winfried Hochstättler, and Walter Kern. Online matching on a line. *Theor. Comput. Sci.*, 332(1-3):251–264, 2005.
5. Anupam Gupta and Kevin Lewi. The online metric matching problem for doubling metrics. In *Proceedings of the 39th ICALP*, pages 424–435, 2012.
6. Varun Gupta, Ravishankar Krishnaswamy, and Sai Sandeep. Permutation strikes back: The power of recourse in online metric matching. In *Proceedings of the 23rd APPROX*, pages 40:1–40:20, 2020.
7. Bala Kalyanasundaram and Kirk Pruhs. Online weighted matching. *J. Algorithms*, 14(3):478–488, 1993.
8. Bala Kalyanasundaram and Kirk Pruhs. Online network optimization problems. In *Online Algorithms: The State of the Art*, pages 268–280. Springer-Verlag, 1998. From the Dagstuhl Seminar on Online Algorithms, 1996.
9. Samir Khuller, Stephen G. Mitchell, and Vijay V. Vazirani. On-line algorithms for weighted bipartite matching and stable marriages. *Theor. Comput. Sci.*, 127(2):255–267, 1994.
10. Elias Koutsoupias and Akash Nanavati. The online matching problem on a line. In *Proceedings of the 1st WAOA*, pages 179–191, 2003.
11. Nicole Megow and Lukas Nölke. Online minimum cost matching with recourse on the line. In *Proceedings of the 23rd APPROX*, pages 37:1–37:16, 2020.
12. Krati Nayyar and Sharath Raghvendra. An input sensitive online algorithm for the metric bipartite matching problem. In *Proceedings of the 58th IEEE FOCS*, pages 505–515, 2017.
13. Sharath Raghvendra. Optimal analysis of an online algorithm for the bipartite matching problem on a line. In *Proceedings of the 34th SoCG*, pages 67:1–67:14, 2018.
14. Rob van Stee. SIGACT news online algorithms column 27: Online matching on the line, part 1. *SIGACT News*, 47(1):99–110, 2016.