



Online Bottleneck Semi-matching

Man Xiao¹, Shu Zhao¹, Weidong Li^{1(✉)}, and Jinhua Yang²

¹ School of Mathematics and Statistics, Yunnan University, Kunming, China

² Dianchi College, Kunming, China

Abstract. We introduce the online bottleneck semi-matching (OBSM) problem, which is to assign a sequence of requests to a given set of m servers, such that the maximum cost is minimized. We present a lower bound $m + 1$ and an online algorithm with competitive ratio $2m - 1$ for the OBSM problem on a line, where the distance between every pair of adjacent servers is the same. When $m = 2$, we present an optimal online algorithm with competitive ratio 3 for the OBSM problem. When $m = 3$, we present two optimal online algorithms with competitive ratio at most $3 + \sqrt{2}$ for the OBSM problem on a line, which matches the previous best lower bound proposed about thirty years ago.

Keywords: Online bottleneck matching · Online algorithm · Capacity limits · Competitive ratio

1 Introduction

We are given a metric space (X, d) , where X is a (possibly infinite) set of points and $d(\cdot, \cdot)$ is a distance function. Let $S = \{s_1, s_2, \dots, s_m\} \subseteq X$ be a set of servers and $R = \{r_1, r_2, \dots, r_n\} \subseteq X$ be a set of requests arriving one-by-one in an online fashion. When a request $r_j \in R$ arrives, it must be immediately and irrevocably matched to some previously unmatched server s_i . The cost of matching r_j to s_i is $d(r_j, s_i)$. The classical Online Minimum Matching (OMM) [10] is to find a matching M such that the total cost of matching all requests is minimized. The Online Bottleneck Matching (OBM) [10] is to find a matching M such that the maximum cost of matching all requests is minimized.

We use competitive ratio to evaluate the performance of an online algorithm A . For an input instance I , let $C^A(I)$ (C^A for short) and $C^{OPT}(I)$ (C^{OPT} for short) be the costs of the feasible solution obtained by an online algorithm A and an optimal offline algorithm, respectively. An online algorithm A is ρ -competitive (or the competitive ratio of A is at most ρ) if $C^A \leq \rho C^{OPT}$ for any input instance I .

Kalyanasundaram and Pruhs [10] introduced the OMM problem and proved that the PERMUTATION algorithm is $(2n - 1)$ -competitive and optimal. Bansal et al. [5] presented an $O(\log^2 n)$ -competitive randomized algorithm for the OMM problem. Kalyanasundaram and Pruhs [11] proposed an interesting question whether one can design an optimal online algorithm for the OMM problem on

a line. Gupta and Lewi [7] gave an $O(\log n)$ -competitive randomized algorithm for the OMM problem on a line. Fuchs et al. [6] showed that no online algorithm can achieve a competitive ratio strictly less than 9.001 for the OMM problem on a line. Antoniadis et al. [4] designed a deterministic online algorithm with competitive ratio $O(n^{\log(3+\epsilon)-1}/\epsilon)$ for any $\epsilon > 0$ for the OMM problem on a line. Nayyar and Raghvendra [13] proved that the competitive ratio of the deterministic online algorithm proposed in [15] is $O(\log^2 n)$, which is improved to $O(\log n)$ [16], for the OMM problem on a line. Recently, Peserico and Scquizzato [14] proved that the competitive ratio of any randomized online algorithm for the OMM problem on a line exceeds $\sqrt{\log_2(n+1)}/15$.

Kalyanasundaram and Pruhs [10] introduced the OBM problem and proved that the PERMUTATION algorithm is $(2n-1)$ -competitive. Idury and Schaffer [8] gave a lower bound approximately $1.44n$ for the OBM problem on a line. Anthony and Cheung [2, 3] used resource augmentation analysis to examine the performance of several classic online algorithms for the OBM problem and its variant where a specified number of requests can be reject or skip.

A generalized version of the OMM problem, which is called online b -matching [10], online transportation [10], the fire station problem [12], the school assignment problem [12], or online facility assignment [1], is also considered, where each server can be matched multiple times. Recently, Itoh et al. [9] presented several lower bounds on the competitive ratio for this problem with different number of servers.

In this paper, we consider a variant of the OBM problem, called online bottleneck semi-matching (OBSM), where each server can be matched multiple times and obtain several interesting results. The remainder of the paper is organized as follows. We first introduce the problem definition and theoretical results in Sect. 2. We provide several lower and upper bounds for the OBSM problem with arbitrary number of servers in Sect. 3. We then present several optimal online algorithms for the OBSM problem with 2 or 3 servers in Sect. 4 and Sect. 5, respectively. Finally, we conclude the paper with future research directions in Sect. 6.

2 Preliminaries

We are given a metric space (X, d) and a set $S = \{s_1, s_2, \dots, s_m\} \subseteq X$ of servers, where X is a (possibly infinite) set of points and $d(\cdot, \cdot)$ is a distance function. Let $R = \{r_1, r_2, \dots, r_n\} \subseteq X$ be a set of requests arriving one-by-one in an online fashion, where $2 \leq m \leq n$. Each server $s_i \in S$ is characterized by the capacity $c_i \in \mathbb{N}$ that satisfies

$$\sum_{i=1}^m c_i = n.$$

It means that the server s_i can be matched with exact c_i requests.

When a request $r_j \in R$ arrives, an online algorithm A must immediately and irrevocably assign a server $s_i = \pi(r_j) \in S$ which is matched less than c_i times to

service that request. The cost of matching r_j to $\pi(r_j)$ is $d(r_j, \pi(r_j))$. After all the requests are matched, we obtain a *semi-matching* M where each vertex $r_j \in R$ is matched exactly once. The online bottleneck semi-matching (OBSM) problem is to find an assignment $\pi : R \mapsto S$ such that $|\{r_j | \pi(r_j) = s_i\}| = c_i$ and the maximum cost $\max_j d(r_j, \pi(r_j))$ of matching all requests is minimized. Clearly, the OBSM problem is a generalized version of the OBM problem considered in [2, 10]. A most related problem is the online minimum semi-matching (OMSM) problem, which is to find an assignment $\pi : R \mapsto S$ such that $|\{r_j | \pi(r_j) = s_i\}| = c_i$ and the maximum cost $\sum_{j=1}^n d(r_j, \pi(r_j))$ of matching all requests is minimized.

It is widely acknowledged that the line is the most interesting metric space for the related online matching problems [4, 14]. If X is a line, without loss of generality, assume that the servers are placed in an increasing order of their indices, i.e.,

$$0 = p(s_1) < p(s_2) < \dots < p(s_m),$$

where $p(s_i)$ is the position of server s_i on the line, for $i = 1, 2, \dots, m$. Let $p(r)$ be the position of request r . The distance between $a, b \in S \cup R$ can be described as $d(a, b) = |p(a) - p(b)|$. Moreover, for $i = 1, 2, \dots, m - 1$, let $d_i = p(s_{i+1}) - p(s_i)$ be the distance between two adjacent servers.

3 Online Bottleneck Semi-matching with Arbitrary Number of Servers

In this section, we consider the OBSM problem with arbitrary number of servers. If each server $s_i \in S$ is replaced by c_i servers with capacity 1, the OBSM problem is exactly the OBM problem considered in [2]. Therefore, following from Theorem 3 and Theorem 4 in [2], we have

Theorem 1. The competitive ratio of the PERMUTATION algorithm is at most $(2n - 1)$ for the OBSM problem.

Kalyanasundaram and Pruhs [10] conjectured that the competitive ratio of the PERMUTATION algorithm is $(2m - 1)$ for the OMSM problem. However, this conjecture is still open. Similarly, it is interesting to design an online algorithm for the OBSM problem whose competitive ratio is a function of m . We obtain several related results for the OBSM problem in this section.

Theorem 2. The competitive ratio of any deterministic online algorithm for the OBSM problem on a line is at least

$$2 + \frac{1}{2^{\frac{1}{m-1}} - 1}.$$

Proof. Similarly to the construction in [8]. ■

Theorem 3. The competitive ratio of any deterministic online algorithm is at least $m + 1$ for the OBSM problem on a line, even if $d_i = 1$ for every $i = 1, 2, \dots, m - 1$.

Proof. Let A be any deterministic online algorithm, and π be the corresponding assignment. Our adversary first gives $c_i - 1$ requests at $p(s_i)$ for each $i = 1, 2, \dots, m$. If A matches some request r with a server not at $p(r)$, then the adversary gives m more requests with one at each position of the server s_i . An optimal offline assignment π^* matches every request r with the server at the same position $p(r)$. Therefore, $C^A > 0$ and $C^{OPT} = 0$, implying that the competitive ratio of A is infinity.

Suppose that A matches each request r with a server at $p(r)$. For convenience, let r_1, r_2, \dots, r_m be the last m requests and $x = \frac{1}{m}$. The adversary gives the requests r_i at $p(s_{i+1}) - x$ for $i = 1, 2, \dots, m - 1$ one by one (see Fig. 1).

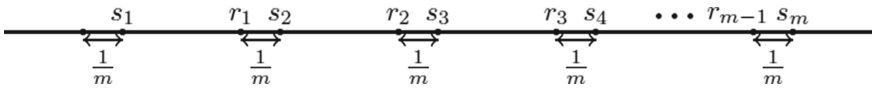


Fig. 1. The $m - 1$ requests.

If A matches each request r_i with server s_{i+1} for $1, 2, \dots, m - 1$. The adversary gives the last request r_m at $p(s_m) + 1 - x$. Clearly, we have $C^{OPT} = 1 - x = 1 - \frac{1}{m}$, and

$$C^A \geq d(r_m, s_1) = m - 1 + 1 - x = m - \frac{1}{m} \geq (m + 1)C^{OPT}.$$

Otherwise, the adversary gives the last request r_m at $p(s_1) - x$. Clearly, we have $C^{OPT} = x = \frac{1}{m}$. We distinguish the follow three cases.

Case 1. A matches request r_1 with server s_1 . Clearly, we have

$$C^A \geq d(r_m, \pi(r_m)) \geq 1 + x = 1 + \frac{1}{m} \geq (m + 1)C^{OPT}.$$

Case 2. A matches request r_1 with server s_2 . Let $k \in \{2, \dots, m - 1\}$ be the minimum index such that A does not match request r_k with server s_{k+1} . If A matches request r_k with server s_l with $l \leq k$, by the minimality of k , r_k must be matched with s_1 , implying that

$$C^A \geq d(r_k, s_1) \geq 2 - x \geq 1 + \frac{1}{m} \geq (m + 1)C^{OPT}.$$

If A matches r_k with a server s_u with $u \geq k + 2$, we have

$$C^A \geq d(r_k, s_u) \geq 1 + x = 1 + \frac{1}{m} \geq (m + 1)C^{OPT}.$$

Case 3. A matches request r_1 with a server s_k ($k \geq 3$). Clearly, we have

$$C^A \geq d(r_1, s_k) \geq 1 + x = 1 + \frac{1}{m} \geq (m + 1)C^{OPT}.$$

Therefore, the theorem holds. ■

As mentioned in [2, 10], GREEDY assigns the nearest available server to each request as it arrives. It is proved that the GREEDY algorithm performs exponentially poorly for the OMM problem [10] and the OBM problem [2]. However, GREEDY performs well for the OBSM problem on a line with $d_i = 1$.

Theorem 4. The competitive ratio of GREEDY is at most $2m - 1$ for the OBSM problem on a line with $d_i = 1$ for every $i = 1, 2, \dots, m - 1$.

Proof. Without loss of generality, assume that

$$p(s_i) = i - 1, \text{ for } i = 1, 2, \dots, m.$$

Let $I_1 = (-\infty, \frac{1}{2}]$, $I_m = (m - 1 - \frac{1}{2}, +\infty)$, and $I_i = (i - 1 - \frac{1}{2}, i - 1 + \frac{1}{2}]$ for $i = 2, 3, \dots, m - 1$. Let

$$R_i = \{r_i \in R | p(r_i) \in I_i\}.$$

be the set of requests whose positions lie in I_i for $i = 1, 2, \dots, m$.

Clearly, if $|R_i| = c_i$ for every i , GREEDY produces an optimal solution. Else, we have

$$C^{OPT} \geq \frac{1}{2}.$$

Let $r \in R$ be the request attaching the maximum in the feasible solution produced by GREEDY, which implies that $C^A = d(r, \pi(r))$. We distinguish the following three cases.

Case 1. $p(r) \in R_1$.

If $p(r) \in (-\infty, -\frac{1}{2}]$, we have $C^{OPT} \geq d(r, s_1) = |p(r)|$, and

$$C^A \leq |p(r)| + d(s_1, s_m) = |p(r)| + m - 1 \leq |p(r)|(1 + \frac{m - 1}{|p(r)|}) \leq (2m - 1)C^{OPT}.$$

If $p(r) \in (-\frac{1}{2}, \frac{1}{2}]$, we have

$$C^A \leq d(r, s_m) \leq m - 1 + \frac{1}{2} = (2m - 1) \cdot \frac{1}{2} \leq (2m - 1)C^{OPT}.$$

Case 2. $p(r) \in R_i$ with $i \in \{2, 3, \dots, m - 1\}$.

Clearly, we have

$$C^A \leq d(r, s_m) \leq m - 1 \leq (2m - 1)C^{OPT}.$$

Case 3. $p(r) \in R_m$.

If $p(r) \in (m - 1 - \frac{1}{2}, m - 1 + \frac{1}{2}]$, we have

$$C^A \leq d(r, s_1) \leq m - 1 + \frac{1}{2} = (2m - 1) \cdot \frac{1}{2} \leq (2m - 1)C^{OPT}.$$

If $p(r) \in (m - 1 + \frac{1}{2}, +\infty)$, we have $C^{OPT} \geq d(r, s_m) = p(r) - (m - 1) \geq \frac{1}{2}$ and

$$\begin{aligned} C^A &\leq d(r, s_1) \leq p(r) = p(r) - (m - 1) + (m - 1) \\ &\leq d(r, s_m) + (2m - 2)C^{OPT} \leq (2m - 1)C^{OPT}. \end{aligned}$$

Therefore, we have $C^A \leq (2m - 1)C^{OPT}$ in any case. ■

4 Online Bottleneck Semi-matching with Two Servers

When $m = 2$, it is proved that the PERMUTATION algorithm is an optimal online algorithm with competitive ratio 3 for the OMM problem [10] and the OBM problem [2]. Recently, Itoh et al. [9] proved that GREEDY algorithm is an optimal online algorithm with competitive ratio 3 for the OMSM problem.

By Theorem 2, the lower bound for the OBSM problem is 3 when $m = 2$. Recall that GREEDY assigns the nearest available server to each request as it arrives. We obtain

Theorem 5. The competitive ratio of the GREEDY algorithm is 3.

Proof. Let R be a minimal instance with the least number of requests whose competitive ratio is maximized. For $j = 1, 2, \dots, n$, if r_j is matched with the server in the offline optimal solution π^* and the feasible solution π produced by the GREEDY algorithm. Without loss of generality, assume that $\pi(r_j) = \pi^*(r_j) = s_1$. If $C^A = d(r_j, s_1)$, we have $C^{OPT} = C^A$, implying that the GREEDY algorithm produces an optimal solution. Else, we construct a new instance $R' = R \setminus \{r_j\}$ with $c'_1 = c_1 - 1$. It is easy to verify that $C^A(R') = C^A(R)$ and $C^{OPT}(R') = C^{OPT}(R)$, which contradicts the minimality of R . Therefore, we have

$$\pi(r_j) \neq \pi^*(r_j), \text{ for each } j = 1, 2, \dots, n. \tag{1}$$

Let r_{j_1} be the request attaching the maximum in the feasible solution produced by the GREEDY algorithm, which means $C^A = d(r_{j_1}, \pi(r_{j_1}))$. Without loss of generality, assume that $\pi(r_{j_1}) = s_1$, which means that

$$C^A = d(r_{j_1}, s_1) \geq d(r_{j_1}, s_2), \text{ and } \pi^*(r_{j_1}) = s_2.$$

Let r_{j_2} be the request attaching the maximum in the optimal solution, which means $C^{OPT} = d(r_{j_2}, \pi^*(r_{j_2}))$. We distinguish the following two cases.

Case 1. $\pi^*(r_{j_2}) = s_1 = \pi(r_{j_1})$.

By (1), we have $j_1 \neq j_2$ and $\pi^*(r_{j_1}) = \pi(r_{j_2}) = s_2$. If $j_1 < j_2$, by the choice of GREEDY, we have $d(r_{j_1}, s_1) \leq d(r_{j_1}, s_2)$. Therefore,

$$C^A = d(r_{j_1}, s_1) \leq d(r_{j_1}, s_2) = d(r_{j_1}, \pi^*(r_{j_1})) \leq C^{OPT}.$$

If $j_1 > j_2$, by the choice of GREEDY, we have $d(r_{j_2}, s_2) \leq d(r_{j_2}, s_1)$. Therefore,

$$\begin{aligned} C^A = d(r_{j_1}, s_1) &\leq d(r_{j_1}, s_2) + d(s_1, s_2) \\ &\leq d(r_{j_1}, s_2) + d(s_1, r_{j_2}) + d(r_{j_2}, s_2) \\ &\leq d(r_{j_1}, \pi^*(r_{j_1})) + 2d(r_{j_2}, s_1) \\ &= d(r_{j_1}, \pi^*(r_{j_1})) + 2d(r_{j_2}, \pi^*(r_{j_2})) \\ &\leq 3C^{OPT}. \end{aligned}$$

Case 2. $\pi^*(r_{j_2}) = s_2$.

Clearly, we have $C^{OPT} = d(r_{j_2}, s_2) \leq d(r_{j_2}, s_1)$. Since $c_1 + c_2 = n$, there is a request r_j with $j < j_1$ satisfying that $\pi^*(r_j) = s_1$ and $\pi(r_j) = s_2$. By the choice of GREEDY, we have $d(r_j, s_2) \leq d(r_j, s_1) = d(r_j, \pi^*(r_j)) \leq C^{OPT}$. Therefore,

$$\begin{aligned} C^A = d(r_{j_1}, s_1) &\leq d(r_{j_1}, s_2) + d(s_2, s_1) \\ &\leq d(r_{j_1}, \pi^*(r_{j_1})) + d(s_2, r_j) + d(r_j, s_1) \\ &\leq C^{OPT} + 2d(r_j, s_1) \\ &\leq 3C^{OPT}. \end{aligned}$$

Thus, $C^A \leq 3C^{OPT}$ in any case. ■

5 Online Bottleneck Semi-matching on a Line with Three Servers

When $m = 3$ and X is a line, Kalyanasundaram and Pruhs [10] claimed that the optimal competitive ratio for the OMM problem on a line is 3.6494359. Recently, Itoh et al. [9] gave a lower bound $1 + \sqrt{6}$ on the competitive ratio for the OMSM problem on a line with $d_1 = d_2 = 1$. However, the optimal competitive ratio is not given. Idury and Schaffer [8] gave a lower bound $3 + \sqrt{2}$ on the competitive ratio for the OBM problem on a line. In this section, we consider the OBSM problem on a line with $m = 3$. Without loss of generality, assume that

$$p(s_1) = 0, p(s_2) = 1, \text{ and } p(s_3) = 1 + \alpha \geq 2.$$

When $\alpha \leq \sqrt{2}$, we design an optimal online algorithm with a competitive ratio of $3 + \alpha$. When $\alpha > \sqrt{2}$, we design an optimal online algorithm with a competitive ratio of $3 + \frac{2}{\alpha}$. Clearly, the upper bound on the competitive ratio for the OBM problem on a line is $3 + \sqrt{2}$, which matches the lower bound given in [8].

Theorem 6. When $1 \leq \alpha \leq \sqrt{2}$, the competitive ratio of any online algorithm A for the OBSM problem on a line is at least $3 + \alpha$.

Proof. Let $c_1 = c_2 = c_3 = 1$. The first request r_1 arrives at $p(r_1) = p(s_2) - \frac{1}{2+\alpha}$. We distinguish the following three cases.

Case 1. r_1 is matched with s_1 .

The last two requests r_2 and r_3 arrive at $p(r_2) = p(s_1) - \frac{1}{2+\alpha}$ and $p(r_3) = p(s_3)$, respectively. Therefore, $C^{OPT} = \frac{1}{2+\alpha}$ and

$$C^A \geq d(r_2, s_2) = 1 + \frac{1}{2+\alpha} \geq (3+\alpha)C^{OPT}.$$

Case 2. r_1 is matched with s_2 .

The second request r_2 arrives at $p(r_2) = p(s_2) + \frac{\alpha}{2}$. If r_2 is matched with s_1 , the last request r_3 arrives at $p(r_3) = p(s_1) - \frac{\alpha}{2}$. Since $\frac{1}{2+\alpha} \leq \frac{1}{2} \leq \frac{\alpha}{2}$, we have $C^{OPT} = \frac{\alpha}{2}$ and $C^A = d(r_3, s_3) = 1 + \alpha + \frac{\alpha}{2}$, then

$$C^A \geq d(r_3, s_3) = 1 + \alpha + \frac{\alpha}{2} \geq (3 + \frac{2}{\alpha})C^{OPT} \geq (3 + \alpha)C^{OPT},$$

as $\alpha \leq \sqrt{2}$.

If r_2 is matched with s_3 , the last request r_3 arrives at $p(r_3) = p(s_3) + \frac{1+\alpha}{2+\alpha}$. Since $\alpha \leq \sqrt{2}$, we have $\frac{1+\alpha}{2+\alpha} \geq \frac{\alpha}{2}$, which implies that $C^{OPT} = d(r_3, s_3) = \frac{1+\alpha}{2+\alpha}$, and

$$C^A = d(r_3, s_1) = 1 + \alpha + \frac{1+\alpha}{2+\alpha} \geq (3+\alpha)C^{OPT}.$$

Case 3. r_1 is matched with s_3 .

The last two requests r_2 and r_3 arrive at $p(r_2) = p(s_1)$ and $p(r_3) = p(s_3)$, respectively. Clearly, $C^{OPT} = \frac{1}{2+\alpha}$, and

$$C^A \geq d(r_1, s_3) = \frac{1}{2+\alpha} + \alpha = \frac{\alpha^2 + 2\alpha + 1}{2+\alpha} \geq (3+\alpha)C^{OPT},$$

as $\alpha \geq 1$.

Therefore, the theorem holds. ■

For convenience, let

$$\begin{aligned} I_1 &= (-\infty, p(s_2) - \frac{1}{2+\alpha}), \\ I_2 &= [p(s_2) - \frac{1}{2+\alpha}, p(s_2) + \frac{\alpha^2 + \alpha - 1}{2+\alpha}], \\ \text{and } I_3 &= (p(s_2) + \frac{\alpha^2 + \alpha - 1}{2+\alpha}, +\infty). \end{aligned}$$

Three intervals are depicted in Fig. 2. For each server s_i , we say that s_i is *available* if it is matched less than c_i times. Otherwise, we say that s_i is *full*.

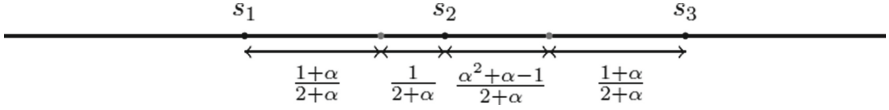


Fig. 2. Three intervals in ALGORITHM A1

ALGORITHM A1:

When a new request r_j arrives, we distinguish the following three cases.

Case 1. $p(r_j) \in I_1$. Match r_j with the first available server in the sequence (s_1, s_2, s_3) .

Case 2. $p(r_j) \in I_2$. Match r_j with the first available server in the sequence (s_2, s_1, s_3) .

Case 3. $p(r_j) \in I_3$. Match r_j with the first available server in the sequence (s_3, s_2, s_1) .

Theorem 7. When $1 \leq \alpha \leq \sqrt{2}$, the competitive ratio of **Algorithm A1** for the OBSM problem on a line is at most $3 + \alpha$.

Proof. We omitted the proof due to space constraints. ■

Theorem 8. When $\alpha > \sqrt{2}$, the competitive ratio of any online algorithm A for the OBSM problem on a line is at least $3 + \frac{2}{\alpha}$.

Proof. Let $c_1 = c_2 = c_3 = 1$. The first request arrives at $p(r_1) = p(s_2) - \frac{1}{2+\alpha}$. We distinguish the following three cases.

Case 1. r_1 is matched with s_1 .

The last two requests arrive at $p(r_2) = p(s_1) - \frac{1}{2+\alpha}$ and $p(r_3) = p(s_3)$, respectively. Therefore, $C^{OPT} = \frac{1}{2+\alpha}$ and

$$C^A \geq d(r_2, s_2) = 1 + \frac{1}{2+\alpha} \geq (3 + \alpha)C^{OPT} \geq (3 + \frac{2}{\alpha})C^{OPT},$$

where the last inequality follows from the assumption $\alpha > \sqrt{2}$.

Case 2. r_1 is matched with s_2 .

The second request arrives at $p(r_2) = p(s_2) + \frac{\alpha}{2}$. If r_2 is matched with s_1 , the last request r_3 arrives at $p(r_3) = p(s_1) - \frac{\alpha}{2}$. Therefore, $C^{OPT} = \frac{\alpha}{2}$ and

$$C^A = d(r_3, s_3) = 1 + \alpha + \frac{\alpha}{2} \geq (3 + \frac{2}{\alpha})C^{OPT}.$$

If r_2 is matched with s_3 , the last request r_3 arrives at $p(r_3) = p(s_3) + \frac{\alpha}{2}$. Therefore, $C^{OPT} = \frac{\alpha}{2}$, and

$$C^A = d(r_3, s_1) = 1 + \alpha + \frac{\alpha}{2} \geq (3 + \frac{2}{\alpha})C^{OPT}.$$

Case 3. r_1 is matched with s_3 .

The last two requests arrive at $p(r_2) = p(s_1)$ and $p(r_3) = p(s_3)$. Therefore, $C^{OPT} = \frac{1}{2+\alpha}$, and

$$C^A = d(r_1, s_3) \geq \frac{1}{2+\alpha} + \alpha \geq (\alpha^2 + 2\alpha + 1)C^{OPT} \geq (3 + \frac{2}{\alpha})C^{OPT}.$$

as $\alpha > \sqrt{2}$.

Therefore, the theorem holds. ■

ALGORITHM A2:

When a new request r_j arrives, we distinguish the following three cases.

Case 1. $p(r_j) \in (-\infty, p(s_2) - \frac{\alpha}{2+2\alpha})$. Match r_j with the first available server in the sequence (s_1, s_2, s_3) .

Case 2. $p(r_j) \in [p(s_2) - \frac{\alpha}{2+2\alpha}, p(s_2) + \frac{\alpha}{2}]$. Match r_j with the first available server in the sequence (s_2, s_1, s_3) .

Case 3. $p(r_j) \in (p(s_2) + \frac{\alpha}{2}, +\infty)$. Match r_j with the first available server in the sequence (s_3, s_2, s_1) (Fig. 3).

Theorem 9. When $\alpha > \sqrt{2}$, the competitive ratio of ALGORITHM A2 for the OBSM problem on a line is at most $3 + \frac{2}{\alpha}$.

Proof. We omitted the proof due to space constraints. ■

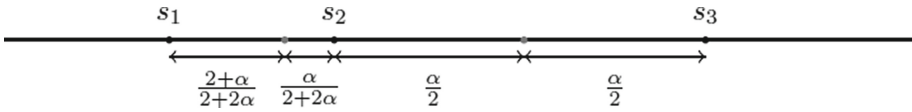


Fig. 3. Three intervals in ALGORITHM A2

6 Conclusion

We propose an online algorithm for the OBSM problem on a line with competitive ratio $2m - 1$, where the distance between every pair of adjacent servers is the same. It is interesting to design an online algorithm with competitive ratio dependent of m for a general metric space. We conjecture that PERMUTATION [2, 10] achieves a $(2m - 1)$ competitive ratio for the OBSM problem and the OMSM problem.

When the number of servers is three, we design two optimal online algorithms for the OBSM problem on a line, whose competitive ratio is dependent on the ratio of two distances between two adjacent servers. In addition, we close the gap for the OBM problem with three servers, which has been open about thirty years. Although Kalyanasundaram and Pruhs [10] claimed the optimal competitive ratio of the OMM problem with three servers is 3.6494359, it is interesting to design several optimal online algorithms with competitive ratio dependent on the ratio of two distances between two adjacent servers, as in Sect. 5.

Acknowledgement. The work is supported in part by the National Natural Science Foundation of China [No. 12071417], Program for Excellent Young Talents of Yunnan University, Training Program of National Science Fund for Distinguished Young Scholars, and IRTSTYN.

References

1. Ahmed, A.R., Rahman, M.S., Kobourov, S.: Online facility assignment. *Theoret. Comput. Sci.* **806**, 455–467 (2020)
2. Anthony, B.M., Chung, C.: Online bottleneck matching. *J. Comb. Optim.* **27**(1), 100–114 (2012). <https://doi.org/10.1007/s10878-012-9581-9>
3. Anthony, B.M., Chung, C.: Serve or skip: the power of rejection in online bottleneck matching. *J. Comb. Optim.* **32**(4), 1232–1253 (2015). <https://doi.org/10.1007/s10878-015-9948-9>
4. Antoniadis, A., Barcelo, N., Nugent, M., Pruhs, K., Scquizzato, M.: A $o(n)$ -competitive deterministic algorithm for online matching on a line. *Algorithmica* **81**, 2917–2933 (2019)
5. Bansal, N., Buchbinder, N., Gupta, A., Naor, J.S.: A randomized $O(\log^2 k)$ -competitive algorithm for metric bipartite matching. *Algorithmica* **68**, 390–403 (2014)
6. Fuchs, B., Hochstattler, W., Kern, W.: Online matching on a line. *Theoret. Comput. Sci.* **332**(1), 251–264 (2005)
7. Gupta, A., Lewi, K.: The online metric matching problem for doubling metrics. In: *Proceedings of International Colloquium on Automata, Languages, and Programming (ICALP)*, pp. 424–435 (2012)
8. Idury, R., Schaffer, A.A.: A better lower bound for on-line bottleneck matching, manuscript (1992)
9. Itoh, T., Miyazaki, S., Satake, M.: Competitive analysis for two variants of online metric matching problem, *Discrete Mathematics, Algorithms and Applications*, ID 2150156 (2021)
10. Kalyanasundaram, B., Pruhs, K.: Online weighted matching. *J. Algorithms* **14**(3), 478–488 (1993) Preliminary version appeared in *Proceedings of the 2nd Annual ACM-SIAM Symposium on Discrete algorithms (SODA)*, pp. 234–240 (1991)
11. Kalyanasundaram, B., Pruhs, K.: On-line network optimization problems. In: Fiat, A., Woeginger, G.J. (eds.) *Online Algorithms*. LNCS, vol. 1442, pp. 268–280. Springer, Heidelberg (1998). <https://doi.org/10.1007/BFb0029573>
12. Kalyanasundaram, B., Pruhs, K.: The online transportation problem. *SIAM J. Discret. Math.* **13**(3), 370–383 (2000)
13. Nayyar, K., Raghvendra, S.: An input sensitive online algorithm for the metric bipartite matching problem. In: *Proceedings of IEEE 58th Annual Symposium on Foundations of Computer Science*, pp. 505–515 (2017)
14. Peserico, E., Scquizzato, M.: Matching on the line admits no $o(\sqrt{\log n})$ -competitive algorithm. In: *Proceedings of the 48th International Colloquium on Automata, Languages, and Programming (ICALP)*, Article No. 103 (2021)
15. Raghvendra, S.: A robust and optimal online algorithm for minimum metric bipartite matching. In: *Proceedings of Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2016)*, ID 18 (2016)
16. Raghvendra, S.: Optimal analysis of an online algorithm for the bipartite matching problem on a line. In: *Proceedings of 34th International Symposium on Computational Geometry*, ID 67 (2017)