Vol.2011-AL-137 No.8
2011/11/18

3 点上の最適なオンラインページ移動

松 林 昭[†1]

ページ移動問題とは，辺重みを持つ無向グラフ $G = (V, E)$，正整数 D，点列 $s_0, r_1, \ldots, r_k \in V$ が与えられ，$\sum_{i=1}^{k}(d_{s_{i-1}r_i} + D \cdot d_{s_{i-1}s_i})$ を最小化するような点列 $s_1, \ldots, s_k \in V$ を求める問題である．ただし，d_{uv} は点 u と点 v の G 上の距離である．この問題は長年研究されているが，$|V| = 3$ という極端に単純な場合ですら，$D = 1, 2$ の場合を除き，決定的オンラインアルゴリズムの厳密な競合比は知られていない．本報告では 3 点上のページ移動問題に対する決定的なオンラインアルゴリズムの競合比が一般の D に対して $3 + \Theta(1/D)$ であることを示す．

Optimal Online Page Migration on Three Points

Akira Matsubayashi[†1]

The page migration problem is as follows: given a sequence of requests from nodes on a network to access a page stored in a node, to compute a sequence of migrations of the page so that the total sum of service costs and the migration costs is minimized, where a service cost is the distance of the request node and the page, and a migration cost is the distance of the migration multiplied by the page size $D \geq 1$. No tight competitive ratio of a deterministic online algorithm has been known even for an extreme case of three nodes, except for $D = 1, 2$. In this report, we prove that the tight competitive ratio of a deterministic online algorithm for the page migration problem on three nodes is $3 + \Theta(1/D)$.

1. Introduction

The problem of computing an efficient dynamic allocation of data objects stored in nodes of a network so that the cost to serve requests for the data objects and to reallocate the data objects is minimized commonly arises in network applications such as memory management in a shared memory multiprocessor system and Peer-to-Peer applications on the Internet. In this paper, we study one of the classical variations of the problem, called *the page migration problem*, in which requests are to be served using unicast communication, and we are allowed to migrate data objects, i.e, no replication is allowed. Serving a request costs the distance of communication, and migrating a data object costs the distance of migration multiplied by the data size $D \geq 1$. The objective function to be minimized is the total sum of the service costs and the migration costs. The page migration problem has been generalized to several settings such as k-page migration[3], file allocation problem[2),4),9], and data management on dynamic networks[1),5].

We consider deterministic online page migration algorithms. Black and Sleator[6] first studied competitive analysis of the page migration problem and presented optimal 3-competitive deterministic online algorithms on trees, uniform networks, and products of those networks, including grids and hypercubes. Currently best deterministic algorithm on general networks that achieves a competitive ratio of 4.086 was proposed by Bartal, Charikar, and Indyk[3]. This upper bound was improved in 10) to $2+\sqrt{2}$ for the specific case that $D = 1$. Moreover, an optimal 3-competitive deterministic algorithm on three nodes for $D = 1$ was presented in 8). In 11), a 3-competitive deterministic algorithm on three nodes for $D = 2$ and a lower bound of $3 + \Omega(1/D^2)$ for $D \geq 3$. were presented. The lower bound greater than 3, specifically $85/27 \approx 3.148$, for deterministic algorithms was first presented by Chrobak, Larmore, Reingold, and Westbrook[8]. This bound was proved for $D = 1$ on an arbitrarily large tree-of-rings network, i.e., a network whose blocks are rings, and was improved in 10) to 3.1639 by refining this technique. It was also mentioned in[8] that the lower bound is greater than 3 even on four nodes, although neither explicit value nor proof was given. An explicit lower bound of 3.1213 on five nodes was proved in 10).

Randomized algorithms have been investigated in e.g., 4), 8), 9), 12). The best-known randomized algorithms on general networks were presented by Westbrook[12]. The algorithms achieve an asymptotic competitive ratio of $(3 + \sqrt{5})/2 \approx 2.6180$ as $D \to \infty$ and a tight competitive ratio of 3 against oblivious and adaptive online adver-

†1 金沢大学大学院自然科学研究科電子情報科学専攻
Division of Electrical Engineering and Computer Science, Kanazawa University

Vol.2011-AL-137 No.8
2011/11/18

saries, respectively.

In this report, we provide a tight deterministic competitive ratio on three nodes. Specifically, we first prove that a typical work function algorithm achieves a competitive ratio of $3+1/D$ on three nodes. We then provide a lower bound of $3+\Omega(1/D)$, which is greater than 3 for any $D \geq 3$, for three node networks.

2. Preliminaries

The page migration problem can be formulated as follows: given an undirected graph $G=(V,E)$ with edge weights, $s_0, r_1, \ldots, r_k \in V$, and a positive integer D, to compute $s_1, \ldots, s_k \in V$ so that the cost function $\sum_{i=1}^{k}(d_{s_{i-1}r_i} + Dd_{s_{i-1}s_i})$ is minimized, where d_{uv} is the distance between nodes u and v on G. The terms $d_{s_{i-1}r_i}$ and $Dd_{s_{i-1}s_i}$ represent the cost to serve the request from r_i by the node s_{i-1} holding the page and the cost to migrate the page from s_{i-1} to s_i, respectively. We call s_i and r_i a *server* and a *client*, respectively. An *online* page migration algorithm determines s_i without information of $r_{i+1}, \ldots, r_k$. We denote by $A(\sigma)$ the cost of a page migration algorithm A for a sequence $\sigma = r_1 \cdots r_k$. A deterministic online page migration algorithm $\textsc{Alg}$ is *ρ-competitive* if there exists a constant value α such that $\textsc{Alg}(\sigma) \leq \rho \cdot \textsc{Opt}(\sigma) + \alpha$ for any σ, where $\textsc{Opt}$ is an optimal offline algorithm. We denote by $\textsc{Opt}_u(\sigma)$ the minimum (offline) cost to process σ so that $s_k = u$. For a node u and $k \geq 1$, we write a sequence consisting of k repetitions of u as u^k.

We suppose that graphs considered here have a node set $V = \{a,b,c\}$ and edge weights $x = d_{ab}$, $y = d_{ac}$, and $z = d_{bc}$ for edges (a,b), (a,c), and (b,c), respectively. We denote $L := x+y+z$ and assume that $\max\{x,y,z\} < L/2$.

3. $3+1/D$-Competitive Algorithm

An online algorithm that determines the output after processing σ using the information of $\textsc{Opt}_u(\sigma)$ for all possible outputs u is called a *work function algorithm* and has extensively been studied for related online problems[7)–9)]. $\textsc{Opt}_u(\sigma)$ is called a *work function* in this context. A work function algorithm is well-defined because $\textsc{Opt}_u(\sigma)$ can be computed by dynamic programming[8)], i.e., for a request issued from r after σ,

$$\textsc{Opt}_u(\sigma r) = \min_{v\in V}\{\textsc{Opt}_v(\sigma) + d_{rv} + Dd_{uv}\}, \text{ and } \textsc{Opt}_u(\emptyset) = Dd_{s_0 u},$$

where $\emptyset$ denotes an empty sequence. We consider a quite common work function algorithm denoted by WFA, which moves the server s to a nearest node among nodes v minimizing $\textsc{Opt}_v(\sigma) + d_{rv} + Dd_{sv}$ after servicing the request from r. We prove the following theorem:

Theorem 1 WFA is $3+1/D$-competitive on 3-node networks.

We suppose that WFA locates the server on s after processing σ, and that a request is issued from $r \in V$ after σ. In the rest of this section, for a function f of σ, we use the notation $f = f(\sigma)$ and $f' = f(\sigma r)$ for simplicity. For $u \in V$, let $\hat{u}$ be a nearest node to u among nodes v minimizing $\textsc{Opt}_v + d_{rv} + Dd_{uv}$. Then,

$$\textsc{Opt}'_s = \textsc{Opt}_{\hat{s}} + d_{r\hat{s}} + Dd_{s\hat{s}} \geq \textsc{Opt}_s + d_{r\hat{s}}, \text{ and} \quad (1)$$

$$\textsc{Opt}'_s \leq \textsc{Opt}'_{\hat{s}} + Dd_{s\hat{s}}. \quad (2)$$

These follow from $|\textsc{Opt}_u - \textsc{Opt}_v| \leq Dd_{uv}$ for any $u,v \in V$[8)]. It follows from (1) and (2) that $d_{r\hat{s}} \leq \textsc{Opt}'_{\hat{s}} - \textsc{Opt}_s + Dd_{s\hat{s}}$. Therefore, we have

$$\text{WFA}' - \text{WFA} = d_{rs} + Dd_{s\hat{s}} \leq d_{r\hat{s}} + (D+1)d_{s\hat{s}} \leq \textsc{Opt}'_{\hat{s}} - \textsc{Opt}_s + (2D+1)d_{s\hat{s}}. \quad (3)$$

By summing up (3) overall requests in σr, we obtain $\text{WFA}' \leq \textsc{Opt}'_{\hat{s}} + (2+1/D)M'$, where $M' = M(\sigma r)$ is D times the total sum of migration distances of WFA in processing σr. Hence, if

$$Dd_{\hat{s}u} + M' \leq \textsc{Opt}'_u \text{ for any } u \in V, \quad (4)$$

then by choosing u minimizing $\textsc{Opt}'_u$, we have $\text{WFA}' \leq \textsc{Opt}'_{\hat{s}} + (2+1/D)\textsc{Opt}' - (2D+1)d_{\hat{s}u} \leq (3+1/D)\textsc{Opt}' - (D+1)d_{\hat{s}u}$, which completes the proof of Theorem 1.

The rest of this section is devoted to prove (4). For this purpose, we accurately analyze the potential function, and therefore, generalize the network to a continuous ring R of length L containing a, b, and c with the preserved distances. Specifically, we define R as an interval $\{p \mid 0 \leq p < L\}$ modulo L, i.e., any real number p is equivalent to $p - \lfloor p/L \rfloor \cdot L$. We define an extended work function at a point $p \in R$ as

$$w'_p = \min_{q \in V \cup \{p\}}\{w_q + d_{rq} + Dd_{pq}\}, \text{ and}$$

$$w_p = Dd_{s_0 p} \text{ if } \sigma = \emptyset.$$

It should be noted that $w_u = \textsc{Opt}_u$ for any $u \in V$. For a point $p \in R$, $\hat{p}$ is a nearest

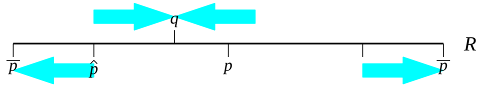

図 1 R 上で $\hat{q}$ が存在する範囲．上下の矢印はそれぞれ $d_{q\hat{q}} \le d_{q\hat{p}}$ と $d_{p\hat{p}} \le d_{p\hat{q}}$ を表す．
Fig. 1 Range in which $\hat{q}$ exists on R. Upper and lower arrows represent $d_{q\hat{q}} \le d_{q\hat{p}}$ and $d_{p\hat{p}} \le d_{p\hat{q}}$, respectively.

point to p among points $q \in V \cup \{p\}$ minimizing $w_q + d_{rq} + Dd_{pq}$. The farthest point of p on R is denoted by $\bar{p}$. For $p, q \in R$, we define $[p, q]$ as the closed interval of length d_{pq} between p and q on R if $d_{pq} < L/2$, R otherwise. Notations $(p, q]$, $[p, q)$, and (p, q) are used to denote the intervals obtained from $[p, q]$ by excluding p, q, and both p and q, respectively. Lemmas 1–3 below state basic properties of w_p.

Lemma 1 For any $p, q \in R$, it follows that $w_p - w_q \le Dd_{pq}$.

Proof The lemma clearly holds if $\sigma = \emptyset$. Otherwise, it follows from the minimality of w'_q that $w'_q \le w_{\hat{p}} + d_{r\hat{p}} + Dd_{q\hat{p}} = w'_p - Dd_{p\hat{p}} + Dd_{q\hat{p}} \le w'_p + Dd_{pq}$. □

Lemma 2 For any $p \in R$ and $q \in (p, \hat{p}]$, it follows that $\hat{q} = \hat{p}$.

Proof It follows from the minimality of w'_p that

$$w'_p = w_{\hat{p}} + d_{r\hat{p}} + Dd_{p\hat{p}} \le w_{\hat{q}} + d_{r\hat{q}} + Dd_{p\hat{q}}. \tag{5}$$

Applying $d_{p\hat{p}} = d_{pq} + d_{q\hat{p}}$ to the inequality, we obtain

$$w_{\hat{p}} + d_{r\hat{p}} + Dd_{q\hat{p}} \le w_{\hat{q}} + d_{r\hat{q}} + D(d_{p\hat{q}} - d_{pq}) \le w_{\hat{q}} + d_{r\hat{q}} + Dd_{q\hat{q}} = w'_q. \tag{6}$$

By the minimality of w'_q, (6) holds with equality. This means that (5) also holds with equality. Therefore, $\hat{p}$ minimizes $w_{\hat{p}} + d_{r\hat{p}} + Dd_{q\hat{p}}$, and $\hat{q}$ minimizes $w_{\hat{q}} + d_{r\hat{q}} + Dd_{p\hat{q}}$. By the minimalities of $d_{q\hat{q}}$ and $d_{p\hat{p}}$, it follows that $d_{q\hat{q}} \le d_{q\hat{p}}$ and $d_{p\hat{p}} \le d_{p\hat{q}}$. Because $q \in (p, \hat{p}]$, $\hat{q}$ exists only at $\hat{p}$ (Fig 1). □

Lemma 3 For any $p \in R$ and $q \in [p, \hat{p})$, it follows that $w_q - w_{\hat{p}} > (D-1)d_{\hat{p}q}$.

Proof Because q is nearer to p than $\hat{p}$ is, it follows that $w_{\hat{p}} + d_{r\hat{p}} + Dd_{p\hat{p}} < w_q + d_{rq} + Dd_{pq}$. Thus, because $d_{p\hat{p}} = d_{pq} + d_{q\hat{p}}$, we have $w_q - w_{\hat{p}} > d_{r\hat{p}} - d_{rq} + D(d_{p\hat{p}} - d_{pq}) \ge (D-1)d_{\hat{p}q}$. □

To prove (4), we utilize connection between the increased amount of the work function and its one-sided derivatives, which are defined as $m_{p-0} := \lim_{q \to p-0} \frac{w_q - w_p}{d_{pq}}$ and $m_{p+0} := \lim_{q \to p+0} \frac{w_q - w_p}{d_{pq}}$ for any $p \in R$. The following lemma guarantees that these derivatives exist and are integers.

Lemma 4 For any $p \in R$, m_{p-0} and m_{p+0} are integers with $-D \le m_{p\pm 0} \le D$.

Proof We prove the lemma by induction on σ. If $\sigma = \emptyset$, then $\{m_{p-0}, m_{p+0}\} \subseteq \{-D, D\}$ by the definition of w_p. Assume that the lemma holds for a sequence σ. By Lemma 2, if $p \ne \hat{p}$, then any point $q \in (p, \hat{p})$ has $\hat{q}$ with $q \ne \hat{q} = \hat{p}$. Therefore, $I := \{q \in R \mid q \ne \hat{q}\}$ is a union of disjoint intervals $(i, \hat{i})$ such that any point $q \in (i, \hat{i})$ has $\hat{q} = \hat{i}$. This means that $w'_q = w_{\hat{i}} + d_{r\hat{i}} + Dd_{q\hat{i}}$. Moreover, $w'_{\hat{i}} = w_{\hat{i}} + d_{r\hat{i}}$ because $\hat{\hat{i}} = \hat{i}$ by Lemma 2. Therefore, for any $p \in [i, \hat{i}]$ and $q \ne p$ in $(i, \hat{i})$, it follows that $(w'_q - w'_p)/d_{pq} = D(d_{q\hat{i}} - d_{p\hat{i}})/d_{pq} = \pm D$.

The set $R \setminus I$ is a union of disjoint intervals $[i, j]$ (with not necessarily distinct end-points i and j) such that any $p \in [i, j]$ has $\hat{p} = p$. Therefore, for $q \ne p$ in (i, j), it follows that

$$\frac{w'_q - w'_p}{d_{pq}} = \frac{(w_q + d_{rq}) - (w_p + d_{rp})}{d_{pq}} = \frac{w_q - w_p}{d_{pq}} + \frac{d_{rq} - d_{rp}}{d_{pq}}. \tag{7}$$

This approaches an integer as $q \to p$ because the first term approaches an integer by induction hypothesis, and because the second term approaches ± 1. By Lemma 1, the absolute value of (7) is at most D. Because $\hat{q} \in V$ for any $q \in R$, I consists of finite disjoint intervals. Therefore, an end-point of an interval of I is an end-point of an interval of $R \setminus I$, and vice versa. Thus, we have the lemma. □

Lemma 5 For any $p \in R \setminus V$, it follows that $m_{p-0} + m_{p+0} \le 0$, i.e., w_p is convex only in the region containing a node in V.

Proof We prove the lemma by induction on σ. If $\sigma = \emptyset$, then $m_{p-0} = m_{p+0} = -D$ for $p = \bar{s}_0$, and $\{m_{p-0}, m_{p+0}\} = \{-D, D\}$ for $p \in R \setminus \{s_0, \bar{s}_0\}$. Assume that the lemma holds for a sequence σ. If $m'_{p-0} \le m_{p-0}$ and $m'_{p+0} \le m_{p+0}$, then the lemma holds by the hypothesis. We assume without loss of generality that $m'_{p-0} > m_{p-0}$. There are two such cases from the proof of Lemma 4.

One case is that m'_{p-0} becomes D, i.e., for some interval $(i, \hat{i})$ in I with $i < \hat{i}$, $p \in (i, \hat{i})$ and $D(d_{q\hat{i}} - d_{p\hat{i}})/d_{pq} = D$ for any q with $i < q < p$. It should be noted that $p \ne \hat{i}$ because $p \notin V$. Then, for any $q \in (p, \hat{i})$, it follows that $D(d_{q\hat{i}} - d_{p\hat{i}})/d_{pq} = -D$, and hence $m'_{p+0} = -D$.

The other case is that $m'_{p-0} = m_{p-0} + 1$, i.e., for some interval $[i, j]$ in $R \setminus I$ with $i < j$, p is contained in $(i, j]$ and $(d_{rq_1} - d_{rp})/d_{pq_1} \to 1$ as $q_1 \to p$ with $i < q_1 < p \le r < p + L/2$. Because $p \ne r$ by $p \notin V$, it follows that $p < r$, and hence, we have $(d_{rq_2} - d_{rp})/d_{pq_2} \to -1$

as $q_2 \to p$ with $p < q_2 < \min\{j, r\}$. This means that $m'_{p+0} = m_{p+0} - 1$. Thus, we have the lemma. □

For $u \in V \setminus \{s\}$, let $m_{s\to u} := \lim_{q\to u,\ q\in[s,u)} \frac{w_q - w_u}{d_{uq}}$ and $m_s := \min\{m_{s\to u} \mid u \in V \setminus \{s\}\}$. We prove (4) together with two other claims in the following lemma:

Lemma 6 The following claims hold.

(1) For $\{p, q\} := V \setminus \{s\}$, $w_p \geq D(L - d_{sp}) + M$, or $w_q \geq D(L - d_{sq}) + M$, or $w_p + w_q \geq m_s d_{pq} + DL + 2M$.

(2) For any $u \in V$, $w_u + w_{\bar{u}} \geq w_s + \frac{DL}{2} + M$.

(3) For any $u \in V$, $w_u \geq Dd_{su} + M$.

Proof Claim 3 follows from Claim 2. This is because $w_s - w_{\bar{u}} \geq -Dd_{s\bar{u}}$ by Lemma 1, and therefore, for any $u \in V$, $w_u \geq w_s - w_{\bar{u}} + \frac{DL}{2} + M \geq -Dd_{s\bar{u}} + \frac{DL}{2} + M = -D(\frac{L}{2} - d_{su}) + \frac{DL}{2} + M = Dd_{su} + M$.

We prove Claims 1 and 2 by induction on events of service and migration of WFA for requests in σ. We suppose that w and m are updated to w' and m' in the event of WFA's service, respectively, and that M is updated to M' in the event of WFA's migration. If $\sigma = \emptyset$, then the claims hold. This is because $w_p + w_q - m_s d_{pq} - 2M = D(d_{sp} + d_{sq}) + Dd_{pq} = DL$, and because $w_u + w_{\bar{u}} - w_s - M = D(d_{su} + d_{s\bar{u}}) = \frac{DL}{2}$ for any $u \in V$. Assume that the claims hold for all events in σ, and that a request is issued from r after σ.

We first prove Claim 1 for the event of WFA's service for r. If $w_p \geq D(L - d_{sp}) + M$ or $w_q \geq D(L - d_{sq}) + M$, then $w'_p \geq w_p \geq D(L - d_{sp}) + M$ or $w'_q \geq w_q \geq D(L - d_{sq}) + M$ follows, and hence, Claim 1 holds for the event. Therefore, we assume that $w_p + w_q \geq m_s d_{pq} + DL + 2M$.

Case 1.1: $\hat{p} = s$. Then, $m'_{s\to p} = -D$, and hence $m'_s = -D \leq m_s$. This means that $w'_p + w'_q - m'_s d_{pq} \geq w_p + w_q - m_s d_{pq} \geq DL + 2M$ by induction hypothesis.

Case 1.2: $\hat{p} = q$. Then, it follows from Claim 3 of induction hypothesis that $w'_p \geq w_q + Dd_{pq} \geq Dd_{sq} + M + Dd_{pq} = D(L - d_{sp}) + M$.

Case 1.3: $\hat{q} \in \{s, p\}$. Similar to the case $\hat{p} \in \{s, q\}$.

Case 1.4: $\hat{p} = p$ and $\hat{q} = q$. If $m'_s \leq m_s + 1$, then $w'_p + w'_q - m'_s d_{pq} \geq w_p + d_{rp} + w_q + d_{rq} - (m_s + 1)d_{pq} \geq w_p + w_q - m_s d_{pq} \geq DL + 2M$ by induction hypothesis. If $m'_s > m_s + 1$, then $m_{s\to p}$ or $m_{s\to q}$, say, $m_{s\to p}$ increases by more than 1. By (the proof of) Lemma 4, this means that $m_{s\to p} < D - 1$, and that there exists $i \in (s, p)$ with $p \in (i, \hat{i}]$. It follows from Lemma 2 that $p = \hat{p} = \hat{i}$. Therefore, it follows from Lemma 3 that $w_j - w_p > (D - 1)d_{pj}$ for any $j \in (i, p)$, which contradicts $m_{s\to p} < D - 1$.

Second, we prove Claim 2 for the event of WFA's service for r. Because $w_{\bar{s}} = w_s + w_{\bar{s}} - w_s \geq \frac{DL}{2} + M$ by induction hypothesis, it follows that $w'_s + w'_{\bar{s}} - w'_s \geq w_{\bar{s}} \geq \frac{DL}{2} + M$. Therefore, without loss of generality, it suffices to prove that $w'_p + w'_{\bar{p}} \geq w'_{\bar{s}} + \frac{DL}{2} + M$.

Case 2.1: $\hat{p} = s$. Then, $\hat{s} = \hat{p} = s$ by Lemma 2. Therefore, it follows that $w'_s = w_s + d_{rs}$. Moreover, $w'_p = w_s + d_{rs} + Dd_{sp} \geq w_p + d_{rs}$ by Lemma 1. Thus, we have $w'_p + w'_{\bar{p}} - w'_s \geq w_p + d_{rs} + w_{\bar{p}} - (w_s + d_{rs}) \geq \frac{DL}{2} + M$ by induction hypothesis.

Case 2.2: $\hat{p} = q$. Then, $w'_p \geq D(L - d_{sp}) + M$ as shown in Case 1.2. Moreover, $w'_{\bar{p}} \geq w'_s - Dd_{s\bar{p}} = w'_s - D(\frac{L}{2} - d_{sp})$ by Lemma 1. Thus, we have $w'_p + w'_{\bar{p}} \geq D(L - d_{sp}) + M + w'_s - D(\frac{L}{2} - d_{sp}) = w'_s + \frac{DL}{2} + M$.

Case 2.3: $\hat{p} = p$. The proof for the case $\hat{\bar{p}} = s$ is similar to that for the case $\hat{p} = s$. If $\hat{\bar{p}} = p$, then it follows from Claim 3 of induction hypothesis that $w'_{\bar{p}} = w_p + d_{rp} + Dd_{p\bar{p}} \geq Dd_{sp} + M + \frac{DL}{2}$. Moreover, $w'_p \geq w'_s - Dd_{sp}$ by Lemma 3. Thus, we have $w'_p + w'_{\bar{p}} \geq w'_s + M + \frac{DL}{2}$. Assume the remaining case $\hat{\bar{p}} = q$. Then, $w_{\bar{p}} - w_q > (D-1)d_{\bar{p}q}$ by Lemma 3. This means $m_{s\to q} = D$ because $m_{s\to q}$ is an integer at most D by Lemma 4, and because there is no node of V between $\bar{p}$ and q, and therefore, no convex point in $(\bar{p}, q)$ by Lemma 5.

Case 2.3.1: $m_{s\to p} = D$. Then, it follows from Claim 1 of induction hypothesis that $w_p \geq D(L - d_{sp}) + M$, or $w_q \geq D(L - d_{sq}) + M$, or $w_p + w_q \geq Dd_{pq} + DL + 2M$. The third inequality implies the first or second inequality. Therefore, it follows that $w'_p \geq w_p \geq D(L - d_{sp}) + M$, or that $w'_{\bar{p}} = w_q + d_{rq} + Dd_{q\bar{p}} \geq D(L - d_{sq}) + M + d_{rq} + Dd_{q\bar{p}} \geq M + D(L - d_{s\bar{p}})$. Both cases can be proved using similar arguments for Case 2.2.

Case 2.3.2: $m_{s\to p} \leq D-1$. This means $w_{\bar{q}} - w_p \leq (D-1)d_{p\bar{q}}$ because there is no node of V between $\bar{q}$ and p, and therefore, no convex point in $(\bar{q}, p)$ by Lemma 5. Therefore, it follows that $w'_p + w'_{\bar{p}} = w_p + d_{rp} + w_q + d_{rq} + Dd_{q\bar{p}} \geq w_q + w_{\bar{q}} + d_{p\bar{q}} + d_{rp} + d_{rq} \geq w_s + \frac{DL}{2} + M + \frac{L}{2}$ by induction hypothesis. Because $w'_s \leq w_s + d_{rs} \leq w_s + \frac{L}{2}$ by the minimality of w'_s, we have $w'_p + w'_{\bar{p}} \geq w'_s + \frac{DL}{2} + M$.

Finally, we prove Claims 1 and 2 for the event of WFA's migration from s to another node, say, p after WFA services the request issued from r. After the service, it follows that

$$w'_s - w'_p = Dd_{sp}. \tag{8}$$

Therefore, it follows that $m'_p = -D$. Moreover, it follows from Claims 2 and 3 (for the event of WFA's service) that

$$w'_u + w'_{\bar{u}} \geq w'_s + \tfrac{DL}{2} + M \text{ for any } u \in V, \text{ and} \tag{9}$$

$$w'_p \geq Dd_{sp} + M. \tag{10}$$

Furthermore, because $\bar{q} \in (s,p)$, it follows that

$$w'_s - w'_{\bar{q}} = Dd_{s\bar{q}} = D(\tfrac{L}{2} - d_{sq}). \tag{11}$$

We obtain $w'_s \geq 2Dd_{sp} + M$ from (8) and (10), and $w'_q \geq D(L - d_{sq}) + M$ from (9) with $u = q$ and (11). Thus, we have $w'_s + w'_q - m'_p d_{sq} \geq 2Dd_{sp} + M + D(L - d_{sq}) + M + Dd_{sq} = DL + 2(Dd_{sp} + M) = DL + 2M'$. Moreover, it follows from (8) and (9) that $w'_u + w'_{\bar{u}} - w'_p \geq w'_s + \frac{DL}{2} + M + Dd_{sp} - w'_s = \frac{DL}{2} + M'$ for any $u \in V$. □

Therefore, the proof of (4) is completed, and hence we have Theorem 1.

4. Lower Bound

In this section we prove the following theorem:

Theorem 2 There exists no deterministic ρ-competitive page migration algorithm on 3-node networks if $\rho = 3 + o(1/D)$. In particular, there exists no deterministic 3-competitive page migration algorithm on 3-node networks if $D \geq 3$.

In this section we assume without loss of generality that $y \geq x \geq z$. Let Alg be a deterministic online page migration algorithm. We denote σ also by σ_v if Alg leaves the last server on a node v after processing σ.

Lemma 7 Let $P \subseteq V$, $Q := V \setminus P$, and let $p \in P$ and $q \in Q$ be joined by an edge with the minimum weight w overall edges joining P and Q. If there exist $\rho > 3$ and a sequence σ_q of clients such that $(\rho-1)\mathrm{Opt}_p(\sigma_q) + \mathrm{Opt}_q(\sigma_q) - \mathrm{Alg}(\sigma_q) + (\rho-5)Dw < 0$, then there exists a sequence $\sigma' = \sigma'_p$ with $\mathrm{Alg}(\sigma_q\sigma') > \rho \cdot \mathrm{Opt}_p(\sigma_q\sigma')$ or a sequence $\sigma'' = \sigma''_q$ with $\mathrm{Alg}(\sigma_q\sigma'') > \rho \cdot \mathrm{Opt}_q(\sigma_q\sigma'')$.

Proof We prove that $\sigma' := p^{k_1}q^{l_1}\cdots p^{k_{i-1}}q^{l_{i-1}}p^{k_i}$ or $\sigma'' := p^{k_1}q^{l_1}\cdots p^{k_i}q^{l_i}$ is a desired sequence for some i. Here, k_j (resp. l_j) $(1 \leq j \leq i)$ is the minimum positive integer such that Alg moves the server from a node of Q (resp. P) to a node P (resp. Q) after processing $\sigma_q p^{k_1}q^{l_1}\cdots p^{k_{j-1}}q^{l_{j-1}}p^{k_j}$ (resp. $\sigma_q p^{k_1}q^{l_1}\cdots p^{k_j}q^{l_j}$).

Assume for contradiction that $\mathrm{Alg}(\sigma_q\sigma') \leq \rho \cdot \mathrm{Opt}_p(\sigma_q\sigma')$ and $\mathrm{Alg}(\sigma_q\sigma'') \leq \rho \cdot \mathrm{Opt}_q(\sigma_q\sigma'')$. Because Alg incurs a cost at least w to serve a request in σ' or σ'' and a cost at least Dw to migrate between P and Q, it follows that

$$\mathrm{Alg}(\sigma_p\sigma'_p) \geq \mathrm{Alg}(\sigma_q) + (K_i + Di + L_{i-1} + D(i-1))w, \text{ and}$$
$$\mathrm{Alg}(\sigma_p\sigma''_q) \geq \mathrm{Alg}(\sigma_q) + (K_i + Di + L_i + Di)w,$$

where $K_j := \sum_{h=1}^{j} k_h$ and $L_j := \sum_{h=1}^{j} l_h$ for $1 \leq j \leq i$. Moreover, an offline algorithm that locates and keeps the server on p (resp. q) after processing σ_q can process $\sigma_q\sigma'$ (resp. $\sigma_q\sigma''$) with a cost of $\mathrm{Opt}_p + L_{i-1}w$ (resp. $\mathrm{Opt}_p + K_i w$). Therefore, it follows that $\mathrm{Opt}_p(\sigma_q\sigma'_p) \leq \mathrm{Opt}_p(\sigma_q) + L_{i-1}w$, and $\mathrm{Opt}_q(\sigma_q\sigma'_q) \leq \mathrm{Opt}_q(\sigma_q) + K_i w$. By the inequalities above, we have

$$\mathrm{Alg}(\sigma_q) + (K_i + Di + L_{i-1} + D(i-1))w \leq \rho(\mathrm{Opt}_p(\sigma_q) + L_{i-1}w), \text{ and}$$
$$\mathrm{Alg}(\sigma_q) + (K_i + Di + L_i + Di)w \leq \rho(\mathrm{Opt}_q(\sigma_q) + K_i w),$$

which yield the recurrences

$$K_i \leq (\rho-1)L_{i-1} - D(2i-1) + A, \text{ and}$$
$$L_i \leq (\rho-1)K_{i-1} - 2Di + B,$$

where $A := (\rho \cdot \mathrm{Opt}_p(\sigma_q) - \mathrm{Alg}(\sigma_q))/w$ and $B := (\rho \cdot \mathrm{Opt}_q(\sigma_q) - \mathrm{Alg}(\sigma_q))/w$. From the recurrences, we have

$$\begin{aligned} K_i &\leq (\rho-1)^2 K_{i-1} - 2\rho Di + (2\rho-1)D + A + (\rho-1)B \\ &\leq \left(-\tfrac{\rho(\rho-1)}{\rho-2}D + (\rho-1)A + B\right)\tfrac{(\rho-1)^{2i-1}}{\rho(\rho-2)} + \tfrac{2Di}{\rho-2} + \tfrac{\frac{\rho}{\rho-2}D - A - (\rho-1)B}{\rho(\rho-2)} \\ &= \left(-\tfrac{\rho(\rho-1)}{\rho-2}D + (\rho-1)A + B\right) \cdot \Theta((\rho-1)^{2i}) + O(i). \end{aligned}$$

The factor of $\Theta((\rho-1)^{2i})$ can be estimated as follows:

$$-\tfrac{\rho(\rho-1)}{\rho-2}D + (\rho-1)A + B = \tfrac{\rho}{w}\left((\rho-1)\mathrm{Opt}_p(\sigma_q) + \mathrm{Opt}_q(\sigma_q) - \mathrm{Alg}(\sigma_q) - \tfrac{\rho-1}{\rho-2}Dw\right),$$

which is negative by $-\frac{\rho-1}{\rho-2} \leq \rho - 5$ for $\rho \geq 3$ and by the assumption of the lemma. Therefore, K_i decreases as i grows sufficiently large, but it is impossible by definition. □

Lemma 8 Let $p := a$ and $q := b$, or $p := b$ and $q := c$. Let w be the weight of the edge (p,q). If there exist $\rho > 3$, $\beta > 0$, and a sequence σ_q of clients such that $\mathrm{Alg}(\sigma_q) > \rho \cdot \mathrm{Opt}_q(\sigma_q)$ and $\mathrm{Opt}_q(\sigma_q) \geq \beta Dw$, then there exists a sequence σ' such that $\sigma' = \sigma'_p$ and $\mathrm{Alg}(\sigma_q\sigma') > \rho' \cdot \mathrm{Opt}_p(\sigma_q\sigma')$, or that σ' is an arbitrarily long sequence with $\textsc{Alg}(\sigma_q\sigma') > \rho' \cdot \textsc{Opt}(\sigma_q\sigma')$, where $\rho' := \frac{\beta}{\beta+4}(\rho-3)+3$.

Proof We define σ' as follows:

(1) Let τ^0 be an empty sequence and $j := 1$.

(2) $\textsc{Alg}$ have processed $\sigma_q\tau^0\cdots\tau^{j-1}$ and locates the server on q. Then, we generate requests from p repeatedly until $\textsc{Alg}$ locates the server on p. Let i be the number of the requests from p.

(3) If $i \geq ((\beta+1)\rho' - \beta\rho - 1)D$, then set $\sigma' := \tau^0\cdots\tau^{j-1}p^i$, and quit the procedure.

(4) Otherwise, we consider costs of $\textsc{Alg}$ and $\textsc{Opt}$ for the clients p^i with the initial server q. Wherever $\textsc{Alg}$ moves the server between q and $u \notin \{p,q\}$ during the requests, $\textsc{Alg}$ incurs a cost at least $(i+D)w$. This is because w is at most the weight of (p,u) by $y \geq x \geq z$. An offline algorithm that keeps the server on q can process p^i with a cost of iw. Moreover, an offline algorithm that moves the server from q to p first and keeps the server on p can process p^i with a cost of Dw. Thus, we have

$$\begin{aligned}
&(\rho'-1)\textsc{Opt}_q(p^i) + \textsc{Opt}_p(p^i) - \textsc{Alg}(p^i) + (\rho'-5)Dw \\
&\leq (\rho'-1)iw + Dw - (i+D)w + (\rho'-5)Dw \\
&< \{(\rho'-2)((\beta+1)\rho' - \beta\rho - 1) + \rho' - 5\}Dw \qquad (12)\\
&= \{(\beta+1)\rho'^2 - (\beta\rho + 2(\beta+1))\rho' + 2\beta\rho - 3\}Dw \\
&= (\beta+1)(\rho' - A(\rho))(\rho' - B(\rho))Dw < 0,
\end{aligned}$$

where

$$A(\rho) := 1 + \frac{\beta\rho + \sqrt{\beta^2\rho^2 - 4(\beta+1)(\beta\rho-\beta-4)}}{2(\beta+1)}, \text{ and}$$
$$B(\rho) := 1 + \frac{\beta\rho - \sqrt{\beta^2\rho^2 - 4(\beta+1)(\beta\rho-\beta-4)}}{2(\beta+1)}.$$

The last inequality of (12) holds because $B(\rho) < \rho' < A(\rho)$ for $\rho > 3$, which can be verified by $A''(\rho) > 0$, $\rho' = A'(3)\cdot(\rho-3) + A(3) > 3$, and $B(\rho) < 2$. Therefore, by applying Lemma 7 with $P := \{p\}$ and $Q := \{q,u\}$, we can obtain a sequence τ^j beginning with p^i such that $\tau^j = \tau^j_p$ and $\textsc{Alg}(\tau^j) > \rho'\textsc{Opt}_p(\tau^j)$, or that $\tau^j = \tau^j_q$ and $\textsc{Alg}(\tau^j) > \rho'\textsc{Opt}_q(\tau^j)$.

(5) If $\tau^j = \tau^j_p$, then set $\sigma' := \tau^0\cdots\tau^j$, and quit the procedure. Otherwise, set $j := j+1$, and repeat the process from Step 2.

By definition, σ' is σ'_p or arbitrarily long. If the procedure ends in Step 3, then it follows that

$$\begin{aligned}
&\textsc{Alg}(\sigma_q\sigma') - \rho'\textsc{Opt}(\sigma_q\sigma') \\
&\geq \textsc{Alg}(\sigma_q) + \sum_{j\geq 0}\textsc{Alg}(\tau^j) + \textsc{Alg}(p^i) - \rho'\left\{\textsc{Opt}_q(\sigma_q) + \sum_{j\geq 0}\textsc{Opt}_q(\tau^j) + \textsc{Opt}_p(p^i)\right\} \\
&> (\rho-\rho')\textsc{Opt}_q(\sigma_q) + ((\beta+1)\rho' - \beta\rho)Dw - \rho' Dw \\
&= (\rho-\rho')(\textsc{Opt}_q(\sigma_q) - \beta Dw) \geq 0.
\end{aligned}$$

Otherwise, we can prove similarly that $\textsc{Alg}(\sigma_q\sigma') - \rho'\cdot\textsc{Opt}(\sigma_q\sigma') > (\rho-\rho')\textsc{Opt}_q(\sigma_q) > 0$. □

Lemma 9 Let $\{p,q\} := \{a,b\}$ and w be the weight of the edge (p,q). If there exist $\rho > 3$, $\beta > 0$, and a sequence σ_q of clients such that $(\rho-1)\textsc{Opt}_p(\sigma_q) + \textsc{Opt}_q(\sigma_q) - \textsc{Alg}(\sigma_q) + (\rho-5)Dw < 0$ and $\textsc{Opt}_q(\sigma_q) \geq \beta Dw$, then there exists a sequence σ' such that $\sigma' = \sigma'_a$ and $\textsc{Alg}(\sigma_q\sigma') > \rho'\cdot\textsc{Opt}_a(\sigma_q\sigma')$, or that σ' is an arbitrarily long sequence with $\textsc{Alg}(\sigma_q\sigma') > \rho'\cdot\textsc{Opt}(\sigma_q\sigma')$, where $\rho' = \frac{\beta}{\beta+4}(\rho-3)+3$.

Proof Let $P := \{a\}$ and $Q := \{b,c\}$ if $p = a$, $P := \{b,c\}$ and $Q := \{a\}$ otherwise. By applying Lemma 7 with such P and Q, we can obtain a sequence τ such that $\tau = \tau_a$ and $\textsc{Alg}(\sigma_q\tau) > \rho\cdot\textsc{Opt}_a(\sigma_q\tau)$, or that $\tau = \tau_b$ and $\textsc{Alg}(\sigma_q\tau) > \rho\cdot\textsc{Opt}_b(\sigma_q\tau)$. If $\tau = \tau_a$, then we have obtained a desired sequence. Otherwise, by Lemma 8, there exists a sequence τ' such that $\tau = \tau_a$ and $\textsc{Alg}(\sigma_q\tau_b\tau') > \rho'\cdot\textsc{Opt}_a(\sigma_q\tau_b\tau')$, or that τ' is an arbitrarily long sequence with $\textsc{Alg}(\sigma_q\tau_b\tau') > \rho'\cdot\textsc{Opt}(\sigma_q\tau_b\tau')$. Therefore, $\tau\tau'$ is a desired sequence. □

Now we prove Theorem 2. Suppose that $y = x+\delta$ and $z = \gamma\delta$ with $3 \leq \gamma \leq x/\delta$. We carefully choose γ and δ, and design a strategy to generate an arbitrarily long sequence σ with $\textsc{Alg}(\sigma) > \rho\cdot\textsc{Opt}(\sigma)$ for some $\rho = 3+\Omega(1/D)$. This proves that $\textsc{Alg}(\sigma) \geq \rho\cdot\textsc{Opt}(\sigma) + \alpha$ for any α independent of the number of clients.

We locate the initial server for σ on a. The strategy is defined using a state machine as shown in Fig. 2. In the state machine, a transition represents a server selected by $\textsc{Alg}$, together with optional conditions on the number of requests generated in the source state. A state with the form of u^k (i.e., b^h, a^j, and c^i) represents a sequence of requests from u until one of the outgoing arcs from the state meets the server of $\textsc{Alg}$ and the conditions on the number k of the requests. A state with the form of u^+ (i.e., a^+ and c^+) represents a sequence of requests from u until $\textsc{Alg}$ locates the

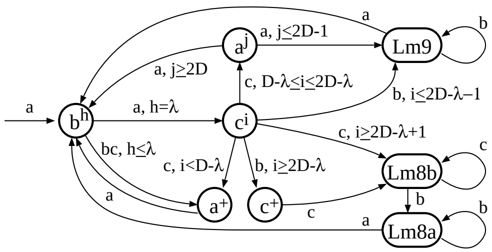

図 2 σ を生成する戦略.
Fig. 2 Strategy to generate σ.

server on u. The states Lm8b and Lm8a represent sequences of requests obtained by applying Lemma 8 with $p := b$ and $q := c$, and with $p := a$ and $q := b$, respectively. The state Lm9 represents a sequence of requests obtained by applying Lemma 9 with $p \in \{a, b\} \setminus \{s\}$ and $q := s$, where $s \in \{a, b\}$ is the server of Alg at the beginning of the state.

We divide σ into phases so that entering the state b^h begins a new phase. Alg locates the server on a at the beginning of each phase. Therefore, Theorem 2 is proved if for each phase $\phi = \phi_a$, $\textsc{Alg}(\phi) > \rho \cdot \textsc{Opt}_a(\phi)$ with the initial server on a, and if for a phase $\phi \neq \phi_a$ (i.e., an arbitrarily long sequence), $\textsc{Alg}(\phi) > \rho \cdot \textsc{Opt}(\phi)$ with the initial server on a.

Case 1: $\phi = b_{bc}^h a^+$ with $h \leq \lambda$. It follows that $\textsc{Alg}(\phi) > (h+2D)x$ and $\textsc{Opt}_a(\phi) \leq hx$ (cost of keeping the server on a). Thus, we have $\frac{\textsc{Alg}(\phi)}{\textsc{Opt}_a(\phi)} > \frac{h+2D}{h} \geq 1 + \frac{2D}{\lambda} > 4$.

Case 2: $\phi = \tau\tau'$, where $\tau := b_a^\lambda c_b^i$ with $i \leq 2D-\lambda-1$, and τ' is the sequence of clients generated in State Lm3. It follows that $\textsc{Alg}(\tau) = (\lambda + D)x + iy$, $\textsc{Opt}_a(\tau) = \lambda x + iy$ (cost of keeping the server on a), and $Dx \leq \textsc{Opt}_b(\tau) \leq Dx + iz$ (cost of moving the server to b first and keeping it on b). Thus, we have

$$\begin{aligned}
&(\rho-1)\textsc{Opt}_a(\tau) + \textsc{Opt}_b(\tau) - \textsc{Alg}(\tau) + (\rho-5)Dx \\
&\quad \leq (\rho-1)(\lambda x + iy) + Dx + iz - ((\lambda + D)x + iy) + (\rho-5)Dx \\
&\quad \leq \rho\{(3D-1)x + (2D-\lambda-1)\delta\} - \{(9D-2)x + (2D-\lambda-1)(2-\gamma)\delta\}.
\end{aligned}$$

Therefore, if $(\rho-1)\textsc{Opt}_a(\tau) + \textsc{Opt}_b(\tau) - \textsc{Alg}(\tau) + (\rho-5)Dx \geq 0$, then we obtain

$$\rho \geq 3 + \frac{x - (2D-\lambda-1)(1+\gamma)\delta}{(3D-1)x + (2D-\lambda-1)\delta},$$

which is $3 + \frac{\epsilon}{O(D)}$ with $0 < \epsilon < 1$ by setting $\gamma = O(1)$ and

$$\delta \leq \frac{(1-\epsilon)x}{(2D-\lambda-1)(\gamma+1)} = O\left(\frac{x}{D}\right). \tag{13}$$

This means that there exists $\rho = 3 + \Omega(1/D)$ such that $(\rho-1)\textsc{Opt}_a(\tau) + \textsc{Opt}_b(\tau) - \textsc{Alg}(\tau) + (\rho-5)Dx < 0$. Therefore, by Lemma 9, there exists $\rho' = 3 + \Omega(1/D)$ such that $\phi = \phi_a$ and $\textsc{Alg}(\phi) > \rho' \cdot \textsc{Opt}_a(\phi)$, or that ϕ is an arbitrarily long sequence with $\textsc{Alg}(\phi) > \rho' \cdot \textsc{Opt}(\phi)$.

Case 3: $\phi = \tau\tau'$, where $\tau = b_a^\lambda c_b^i c^+$ with $i \geq 2D - \lambda$, and τ' is the sequence of clients generated in States Lm2b and Lm2a. It follows that $\textsc{Alg}(\tau) \geq (\lambda+D)x + iy + (1+D)z$ and $Dy \leq \textsc{Opt}_c(\tau) \leq Dy + \lambda z$ (cost of moving the server to c first and keeping it on c). Thus, we have

$$\begin{aligned}
\frac{\textsc{Alg}(\tau)}{\textsc{Opt}_c(\tau)} &\geq \frac{(\lambda+D)x + iy + (1+D)z}{Dy + \lambda z} \geq \frac{3Dx + \{(2D-\lambda) + (1+D)\gamma\}\delta}{Dx + (D+\lambda\gamma)\delta} \\
&= 3 + \frac{\{(\gamma-1)D + \gamma - \lambda(3\gamma+1)\}\delta}{Dx + (D+\lambda\gamma)\delta},
\end{aligned}$$

which is $3 + \frac{\epsilon}{O(D)}$ with $0 < \epsilon < 1$ by setting

$$\gamma := 4 + 3\epsilon = O(1), \tag{14}$$

$$\lambda := \left\lfloor \frac{(\gamma - 1 - \epsilon)D + \gamma}{3\gamma + 1} \right\rfloor = \Theta(D), \text{ and} \tag{15}$$

$\delta = O(\frac{x}{D})$. It should be noted that $1 \leq \lambda < \frac{2D}{3}$ for $D \geq 3$. Therefore, by Lemma 8, there exists $\rho' = 3 + \Omega(1/D)$ such that $\phi = \phi_a$ and $\textsc{Alg}(\phi) > \rho' \cdot \textsc{Opt}_a(\phi)$, or that ϕ is an arbitrarily long sequence with $\textsc{Alg}(\phi) > \rho' \cdot \textsc{Opt}(\phi)$.

Case 4: $\phi = b_a^\lambda c_c^i a^+$ with $i < D - \lambda$. It follows that $\textsc{Alg}(\phi) \geq \lambda x + (i+D+1+D)y = \lambda x + (i+2D+1)y$ and $\textsc{Opt}_a(\phi) \leq \lambda x + iy$ (cost of keeping the server on a). Thus, we have

$$\frac{\textsc{Alg}(\phi)}{\textsc{Opt}_a(\phi)} \geq \frac{\lambda x + (i+2D+1)y}{\lambda x + iy} \geq 1 + \frac{(2D+1)y}{\lambda x + iy} > 1 + \frac{2D+1}{D} = 3 + \frac{1}{D}.$$

Case 5: $\phi = \tau\tau'$ where $\tau = b_a^\lambda c_c^i a_a^j$ with $D - \lambda \leq i \leq 2D - \lambda$ and $j \leq 2D - 1$, and τ' is the sequence of clients generated in State Lm3. If Alg keeps the server on c during a^j, then the cost for a^j is $(j + D)y$. If Alg moves the server from c to b after the j'th request of a^j, then the cost for a^j is at least $j'y + Dz + (j - j' + D)x = jy + D(\gamma\delta + x) - (j - j')\delta$. Because $\gamma \geq 3$ and $j - j' < 2D$, this is at least $jy + D(3\delta + x) - 2D\delta = jy + D(\delta + x) = (j + D)y$. Therefore, it follows that $\textsc{Alg}(\tau) \geq \lambda x + (i+D+j+D)y = \lambda x + (i+j+2D)y$. Moreover, $Dx < \textsc{Opt}_a(\tau) \leq \lambda x + iy$

(cost of keeping the server on a), and $\mathrm{OPT}_b(\tau) \le Dx + iz + jx = (j + D)x + iz$ (cost of moving the server to b and keeping it on b). Thus, we have

$$\begin{aligned}&(\rho-1)\mathrm{OPT}_b(\tau) + \mathrm{OPT}_a(\tau) - \mathrm{ALG}(\tau) + (\rho-5)Dx\\ &\quad\le (\rho-1)((j+D)x+iz) + \lambda x + iy - (\lambda x + (i+j+2D)y) + (\rho-5)Dx\\ &\quad\le \rho\{(4D-1)x + (2D-\lambda)\gamma\delta\} - \{(12D-2)x + (4D-1+(2D-\lambda)\gamma)\delta\}.\end{aligned}$$

Therefore, if $(\rho-1)\mathrm{OPT}_b(\tau) + \mathrm{OPT}_a(\tau) - \mathrm{ALG}(\tau) + (\rho-5)Dx \ge 0$, then we obtain

$$\rho \ge 3 + \frac{x + ((4D-1) - 2(2D-\lambda)\gamma)\delta}{(4D-1)x + (2D-\lambda)\gamma\delta},$$

which is $3 + \frac{\epsilon}{O(D)}$ with $0 < \epsilon < 1$ by setting $\gamma = O(1)$ and

$$\delta \le \frac{(1-\epsilon)x}{2(2D-\lambda)\gamma - (4D-1)} = O\left(\frac{x}{D}\right). \tag{16}$$

Therefore, we can prove as in Case 2 that there exists $\rho' = 3 + \Omega(1/D)$ such that $\phi = \phi_a$ and $\mathrm{ALG}(\phi) > \rho' \cdot \mathrm{OPT}_a(\phi)$, or that ϕ is an arbitrarily long sequence with $\mathrm{ALG}(\phi) > \rho' \cdot \mathrm{OPT}(\phi)$.

Case 6: $\phi = b_a^\lambda c_c^i a_a^j$ with $D - \lambda \le i \le 2D - \lambda$ and $j \ge 2D$. If ALG keeps the server on c during a^j, then the cost for a^j is $(j + D)y \ge 3Dy$. If ALG moves the server from c to b after the j'th request of a^j, then the cost for a^j is at least $j'y + Dz + (j - j' + D)x \ge jx + D(\gamma\delta + x)$. Because $\gamma \ge 3$ and $j \ge 2D$, this is at least $3D(\delta+x) = 3Dy$. Therefore, it follows that $\mathrm{ALG}(\tau) \ge \lambda x + (i+D+3D)y = \lambda x + (i+4D)y$ and $\mathrm{OPT}_a(\tau) \le \lambda x + iy$ (cost of keeping the server on a). Thus, we have

$$\begin{aligned}\frac{\mathrm{ALG}(\tau)}{\mathrm{OPT}_a(\tau)} &\ge \frac{\lambda x + (i+4D)y}{\lambda x + iy} = 1 + \frac{4Dy}{\lambda x + iy}\\ &\ge 1 + \frac{4D(x+\delta)}{2Dx + (2D-\lambda)\delta} = 3 + \frac{2\lambda\delta}{2Dx + (2D-\lambda)\delta},\end{aligned}$$

which is $3 + \Omega(1/D)$ by setting $\lambda = \Theta(D)$ and $\delta = \Theta(x/D)$.

Case 7: $\phi = \tau\tau'$, where $\tau = b_a^\lambda c_c^i$ with $i \ge 2D - \lambda + 1$, and τ' is the sequence of clients generated in States Lm2b and Lm2a. It follows that $\mathrm{ALG}(\tau) \ge \lambda x + (i + D)y$ and $Dy \le \mathrm{OPT}_c(\tau) \le Dy + \lambda z$ (cost of moving the server to c first and keeping it on c). Thus, we have

$$\begin{aligned}\frac{\mathrm{ALG}(\tau)}{\mathrm{OPT}_c(\tau)} &\ge \frac{\lambda x + (i+D)y}{Dy + \lambda z} \ge \frac{(3D+1)x + (3D-\lambda+1)\delta}{Dx + (D+\lambda\gamma)\delta}\\ &= 3 + \frac{x - ((3\gamma+1)\lambda - 1)\delta}{Dx + (D+\lambda\gamma)\delta},\end{aligned}$$

which is $3 + \frac{\epsilon}{O(D)}$ with $0 < \epsilon < 1$ by setting $\gamma = O(1)$ and

$$\delta \le \frac{(1-\epsilon)x}{(3\gamma+1)\lambda - 1} = O\left(\frac{x}{D}\right). \tag{17}$$

Therefore, by Lemma 8, there exists $\rho' = 3 + \Omega(1/D)$ such that $\phi = \phi_a$ and $\mathrm{ALG}(\phi) > \rho' \cdot \mathrm{OPT}_a(\phi)$, or that ϕ is an arbitrarily long sequence with $\mathrm{ALG}(\phi) > \rho' \cdot \mathrm{OPT}(\phi)$.

By setting γ as in (14), λ as in (15), and δ so that (13), (16), (17), and $\delta \le x/\gamma$ are satisfied, we can obtain a desired sequence ϕ. Thus, the proof of Theorem 2 is completed.

参 考 文 献

1) Awerbuch, B., Bartal, Y. and Fiat, A.: Distributed Paging for General Networks, *J.Algorithms*, Vol.28, No.1, pp.67–104 (1998).
2) Awerbuch, B., Bartal, Y. and Fiat, A.: Competitive Distributed File Allocation, *Information and Computation*, Vol.185, No.1, pp.1–40 (2003).
3) Bartal, Y., Charikar, M. and Indyk, P.: On Page Migration and Other Relaxed Task Systems, *Theoretical Computer Science*, Vol.268, No.1, pp.43–66 (2001).
4) Bartal, Y., Fiat, A. and Rabani, Y.: Competitive Algorithms for Distributed Data Management, *J.Computer and System Sciences*, Vol.51, No.3, pp.341–358 (1995).
5) Bienkowski, M., Byrka, J., Korzeniowski, M. and Meyer aufder Heide, F.: Optimal Algorithms for Page Migration in Dynamic Networks, *J.Discrete Algorithms*, Vol.7, No.4, pp.545–569 (2009).
6) Black, D.L. and Sleator, D.D.: Competitive Algorithms for Replication and Migration Problems, Technical Report CMU-CS-89-201, Department of Computer Science, Carnegie Mellon University (1989).
7) Borodin, A. and El-Yaniv, R.: *Online Computation and Competitive Analysis*, Cambridge University Press (1998).
8) Chrobak, M., Larmore, L.L., Reingold, N. and Westbrook, J.: Page Migration Algorithms Using Work Functions, *J.Algorithms*, Vol.24, No.1, pp.124–157 (1997).
9) Lund, C., Reingold, N., Westbrook, J. and Yan, D.: Competitive On-Line Algorithms for Distributed Data Management, *SIAM J.Comput.*, Vol.28, No.3, pp. 1086–1111 (1999).
10) Matsubayashi, A.: Uniform Page Migration on General Networks, *International Journal of Pure and Applied Mathematics*, Vol.42, No.2, pp.161–168 (2007).
11) Matsubayashi, A.: New Bounds for Online Page Migration on Three Points, Technical Report 2008-AL-120, IPSJ SIG (2008).
12) Westbrook, J.: Randomized Algorithms for Multiprocessor Page Migration, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, Vol.7, pp.135–150 (1992).