

Stochastic Analysis of the k-Server Problem on the Circle

Aris Anagnostopoulos, Clément Dombry, Nadine Guillotin-Plantard, Ioannis Kontoyiannis, Eli Upfal

▶ **To cite this version:**

Aris Anagnostopoulos, Clément Dombry, Nadine Guillotin-Plantard, Ioannis Kontoyiannis, Eli Upfal. Stochastic Analysis of the k-Server Problem on the Circle. 21st International Meeting on Probabilistic, Combinatorial, and Asymptotic Methods in the Analysis of Algorithms (AofA'10), 2010, Vienna, Austria. pp.21-34, 10.46298/dmtcs.2791 . hal-01185590

HAL Id: hal-01185590
https://inria.hal.science/hal-01185590

Submitted on 20 Aug 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Stochastic Analysis of the k-Server Problem on the Circle

Aris Anagnostopoulos[1] Clément Dombry[2†]
Nadine Guillotin-Plantard[3†] Ioannis Kontoyiannis[4‡] Eli Upfal[5]

[1]*Department of Informatics and System Sciences, Sapienza University of Rome,* aris@dis.uniroma1.it
[2]*Lab. de Mathématiques et Applications, Univ. de Poitiers,* clement.dombry@math.univ-poitiers.fr
[3]*Université de Lyon, Institut Camille Jordan,* nadine.guillotin@univ-lyon1.fr
[4]*Department of Informatics, Athens University of Economics & Business,* yiannis@aueb.gr
[5]*Computer Science Department, Brown University,* eli@cs.brown.edu

We consider a stochastic version of the k-server problem in which k servers move on a circle to satisfy stochastically generated requests. The requests are independent and identically distributed according to an arbitrary distribution on a circle, which is either discrete or continuous. The cost of serving a request is the distance that a server needs to move to reach the request. The goal is to minimize the steady-state expected cost induced by the requests. We study the performance of a greedy strategy, focusing, in particular, on its convergence properties and the interplay between the discrete and continuous versions of the process.

Keywords: k-server, probabilistic analysis, on-line algorithms

1 Introduction

The *k-server problem* first posed by Manasse, McGeoch, and Sleator [6, 7] is central to the study of on-line algorithms. Let $k \geq 2$ be an integer and consider a metric space $\mathbb{M} = (M, d)$ where $|M| > k$ is a set of points and d is the metric over M. A k-server problem is defined by a sequence of *requests*, $r_1, r_2, \ldots$, where each $r_i \in M$. A k-server algorithm controls k *servers* that reside in, and move between, points of M. A request r_i is satisfied when the algorithm moves a server to that point. The algorithm must satisfy request i before it sees any future request. The cost of serving a sequence of requests is the sum of the distances that servers have to traverse in order to satisfy these requests. The k-server problem models a number of important on-line computation problems such as paging and disk-scheduling [2]. So far, most studies of the k-server problem have focused on the *competitive analysis* of the problem, culminating in Koutsoupias and Papadimitriou's [5] seminal results, establishing a worst-case competitive ratio of $\Theta(k)$ for the problem.

†Supported by CNRS grant DREI 21514.
‡Supported in part by a Marie Curie International Outgoing Fellowship, PIOF-GA-2009-235837.

1365–8050 © 2010 Discrete Mathematics and Theoretical Computer Science (DMTCS), Nancy, France

An alternative approach to the study of on-line algorithms is *probabilistic analysis.* Instead of taking a worst-case approach, we assume that the input (in this case, the sequence of requests) follows some distribution, and we analyze the expected cost of a given algorithm over this input distribution. As in the worst-case scenario, the algorithm satisfies each request without information about the sequence of future requests, and we are interested in algorithms that are not tuned to a particular input distribution.

1.1 The Stochastic Model

In this work we study a stochastic version of the k-server problem in which the k mobile servers move on the circle of circumference k, which we denote by $\mathbb{T}$ and identify with the interval $[0, k)$. Throughout, we focus on the long-term behavior of the process induced by the following *greedy* algorithm: "at each step i, send the server that is closest to the request r_i." We analyze and compare two different models; in the continuous model, requests arrive anywhere on the circle $\mathbb{T} = [0, k)$ according to some distribution with a continuous density; in the discrete case, requests may arrive only at ℓ discrete stations, where $k < \ell < \infty$, and the positions of the stations are in $\mathbb{T}_\ell = \{ik/\ell\ ;\ 0 \le i \le \ell - 1\} \subset \mathbb{T}$. The circle $\mathbb{T}$ is endowed with the distance,

$$d_0(x, y) = \min\{|x - y|\,, |x - y - k|\,, |x - y + k|\},$$

and $\mathbb{T}_\ell$ with the induced distance.

We assume that the requests are independent and identically distributed (i.i.d.) according to a distribution supported either on $\mathbb{T}$ (in the continuous case) or on $\mathbb{T}_\ell$ (in the discrete case); their common distribution is denoted either by μ or μ_ℓ, respectively. For the sake of simplicity, we will always assume that μ has a strictly positive density with respect to Lebesgue measure on $\mathbb{T}$ and that μ_ℓ has positive mass on all points $x \in \mathbb{T}_\ell$.

In the continuous setting, we can model the process as a Markov chain $(S_n)_{n\ge 0}$ evolving on the state space,

$$\Sigma = \{\mathbf{p} = (p_0, p_1, \ldots, p_{k-1}) \in \mathbb{T}^k;\, p_j \le p_{j+1} \le \cdots \le p_{k-1} \le p_0 \le p_1 \le \cdots \le p_{j-1},\ \text{for some } j\}.$$

Here p_i corresponds to the position of the ith server. In the discrete setting, the servers move only within the ℓ stations, and accordingly the state space is $\Sigma_\ell = \Sigma \cap \mathbb{T}_\ell^k$. To avoid ambiguity, in the case when the two closest servers to a given request r_i are equidistant from r_i, we specify that the greedy algorithm moves the server which is to the left of the request (where the circle is oriented in the usual way). In addition, if two or more servers are on the same location (i.e., we have $p_i = p_{i+1} = \cdots = p_j = x$ for $i < j$, or $p_i = p_{i+1} = \cdots = p_{k-1} = p_0 = \cdots = p_j = x$ for $i > j$), and the new request is closer to x than to any other server, we choose server j if the request is on the left of x and server i if it is on the right of x. Note that, under these assumptions, the relative order of the servers on the circle does not change, and that, if at some point all the servers are on different locations, then they will never be on the same location in the future of the process.

The performance of the greedy algorithm is measured in terms of the total distance travelled by the servers to fulfill the requests. Let d be the distance on $\mathbb{T}^k$ defined by

$$d(\mathbf{p}, \mathbf{p}') = \sum_{i=0}^{k-1} d_0(p_i, p_i').$$

Recalling that S_n denotes the position of the servers after satisfying the nth request, the *cost* of the nth request is given by $C_n = d(S_{n-1}, S_n)$. We are interested in the asymptotic behavior of *the average cost over a run of N requests*, namely,

$$\frac{1}{N}\sum_{n=1}^{N} C_n = \frac{1}{N}\sum_{n=1}^{N} d(S_{n-1}, S_n).$$

2 Results

2.1 Convergence to a Stationary Distribution

In the discrete setting, since requests may arrive at all stations with positive probability, the finite state space Markov chain $(S_n)_{n\geq 1}$ has the following structure; let,

$$\tilde{\Sigma} = \{\mathbf{p} = (p_0, p_1, \ldots, p_{k-1}) \in \mathbb{T}^k;\, p_j < p_{j+1} < \cdots < p_{k-1} < p_0 < p_1 < \cdots < p_{j-1}, \text{ for some } j\},$$

and $\tilde{\Sigma}_\ell = \tilde{\Sigma} \cap \mathbb{T}^k_\ell$; then every state in $\Sigma_\ell \setminus \tilde{\Sigma}_\ell$ is transient and $\tilde{\Sigma}_\ell$ is the unique closed communication class and it is aperiodic. As a consequence (see, for example, [9]), the Markov chain (S_n) has a unique stationary distribution, say π_ℓ, supported on $\tilde{\Sigma}_\ell$, the distribution of the chain (S_n) converges to π_ℓ exponentially fast, and the law of large numbers holds. In particular, the average cost over N steps converges almost surely to the mean cost under π_ℓ, as $N \to \infty$.

The main contribution of this work is to obtain similar results in the continuous case. Under the assumption that μ has a positive density, we show in Theorem 1 that the process has a unique stationary distribution π, to which it converges exponentially fast.

Theorem 1 *Assume that μ has a density with respect to the Lebesgue measure on $\mathbb{T} = [0, k)$ that is uniformly bounded away from 0. Then the Markov chain $(S_n)_{n\geq 0}$ on Σ induced by the greedy algorithm has a unique stationary distribution π, and:*

- *for all $n \geq 1$, $\sup_{A, s_0} |P(S_n \in A | S_0 = s_0) - \pi(A)| \leq c_1 e^{-c_2 n}$, the supremum being taken over all measurable sets A of Σ and all $s_0 \in \Sigma$;*
- $\frac{1}{N}\sum_{n=1}^{N} C_n \to \mathrm{E}_\pi(C)$ *with probability 1, as $N \to \infty$;*
- $\Pr\left(\frac{1}{N}\sum_{n=1}^{N} C_n > \mathrm{E}_\pi(C) + \epsilon\right) \leq c_3 e^{-c_4(\epsilon) n}$ *for all $\epsilon > 0,\ n \geq 1$,*

where the finite constants $c_1, c_2, c_3, c_4(\epsilon)$ may depend on k but not on n; $c_2 > 0$ and $c_4 = c_4(\epsilon)$ is strictly positive for all $\epsilon > 0$; $C_i = d(S_{i-1}, S_i)$ is the cost of the ith request; and $\mathrm{E}_\pi(C)$ is the expected value of the one-step cost for a stationary version of the chain.

In view of the well-known convergence results in [8] and the more recent large deviations estimates in [4], in order to prove the theorem it suffices to verify that the Markov chain fulfills Doeblin's condition; see [8, page 391]. The proof is given in Appendix A.

In addition to the results of the theorem, since the cost function is bounded by definition, the Doeblin (or uniform ergodicity) property implies finer asymptotic results for the convergence of the averaged cost $\frac{1}{n}\sum_{n=1}^{N} C_i$. In particular, it is asymptotically normal, and the convergence to normality can be further refined via an Edgeworth expansion. Moreover, the exponential bound on the excess-cost probability can be sharpened to a precise asymptotic formula along the lines of the Bahadur-Rao refinement to the classical large deviations principle [3]. See [4] for details.

2.2 Convergence of the Discrete Model to the Continuous One

When the number of stations ℓ in the discrete model is large, we naturally expect that it should be possible to approximate the continuous model by an appropriately defined version of the discrete one. A precise asymptotic version of the above statement is established in the following proposition.

Proposition 1 *Suppose that $\{\mu_\ell\}$ is a sequence of request distributions on $\mathbb{T}_\ell$, $\ell \geq 1$, such that $\mu_\ell \to \mu$ weakly, as ℓ goes to infinity, for some continuous request distribution μ on $\mathbb{T}$. Then, the corresponding sequence $\{\pi_\ell\}$ of stationary distributions for the discrete chains induced by each of the μ_ℓ also satisfies $\pi_\ell \to \pi$ weakly, as $\ell \to \infty$, where π is the stationary distribution of the continuous chain induced by μ.*

Proof: If the servers are in position $\mathbf{p} \in \Sigma$ and a request arrives at site $x \in \mathbb{T}$, then the new position of the servers after service is

$$Q(\mathbf{p}, x) = (p_0, \ldots, p_{i-1}, x, p_{i+1}, \ldots, p_{k-1}) \tag{2.1}$$

where p_i is the nearest point of x (and the left such point in case of equality). The continuity set of the map Q is exactly the set of points $(\mathbf{p}, x)$ such that there is a unique point p_i nearest to x.

Now suppose the sequences $\{\mu_\ell\}$ and $\{\pi_\ell\}$ are as in the statement of the proposition. According to Theorem 2 below (proof omitted from this version), the stationary measures π_ℓ converge weakly to π, and so do the corresponding expected costs. Note that the assumptions of the theorem are always satisfied when μ is diffuse and, in particular, when μ has a density. □

Theorem 2 *Let $C_Q \subset \Sigma \times \mathbb{T}$ denote the continuity set of the map Q defined in equation* (2.1)*. Assume that the Markov chain $(S_n)_{n\geq 0}$ associated with the request distribution μ has a unique invariant probability measure π, and that, for every probability measure π' on $\mathbb{T}^k$,*

$$(\pi' \times \mu)(C_Q) = 1.$$

Then $\pi_\ell \to \pi$ weakly and $\mathrm{E}_{\pi_\ell}(C) \to \mathrm{E}_\pi(C)$, as $\ell \to \infty$, where $\mathrm{E}_{\pi_\ell}(C)$ and $\mathrm{E}_\pi(C)$, denote the expected costs under the stationary versions of the corresponding chains, respectively.

We provide further insight into the correspondence between the discrete and the continuous models by constructing a *coupling* between the corresponding chains. Consider the continuous model with a sequence of requests $(r_n)_{n\geq 1}$ that are i.i.d. with continuous distribution μ on $\mathbb{T}$. Let $(S_n)_{n\geq 0}$ denote the associated Markov chain on Σ and let $\Theta_\ell : \mathbb{T} \to \mathbb{T}_\ell$ be the discretization map defined by,

$$\Theta_\ell(x) = \frac{k}{\ell} \left\lfloor \frac{\ell}{k} x \right\rfloor .$$

Similarly, let $(r_n^\ell)_{n\geq 1} = (\Theta_\ell(r_n))_{n\geq 1}$ denote the corresponding discretized requests, so that the sequence (r_n^ℓ) is also i.i.d., with discrete distribution μ_ℓ given by $\mu_\ell(x) = \mu([x, x + k/\ell))$ for $x \in \mathbb{T}_\ell$. We consider the Markov chain (S_n^ℓ) defined by the initial state $S_0^\ell = \Theta_\ell(S_0)$ and the sequence of requests $(r_n^\ell)_{n\geq 1}$. Note that S_n^ℓ is in general *not* the same as $\Theta_\ell(S_n)$. We assume that ℓ is large enough so the image under Θ_ℓ of the initial positions of the k servers is a set of k distinct points.

As the number of stations ℓ goes to infinity, Θ_ℓ converges to the identity map and μ_ℓ converges weakly to μ, so we naturally expect that the Markov chains (S_n^ℓ) and $(\Theta_\ell(S_n))$ will also be "close" for large ℓ. In order to quantify this closeness, we examine the *(de-)coupling time*,

$$T_\ell = \inf\{n \geq 1; S_n^\ell \neq \Theta_\ell(S_n)\}.$$

Our next result, whose proof will appear in the full version of this work, is the following:

Proposition 2 *Let $\delta(\epsilon) = \sup\{\mu([x, x + \epsilon]) \; ; \; x \in \mathbb{T}\}$ for $\epsilon > 0$, and assume that $\delta(k/\ell)$ is strictly positive. Then the distribution of T_ℓ stochastically dominates a geometric distribution with parameter $k\delta(k/\ell)$, i.e.,*

$$\Pr(T_\ell > n) \geq \left(1 - k\delta\left(\frac{k}{\ell}\right)\right)^n.$$

As a consequence, $\mathrm{E}(T_\ell) \geq (k\ \delta(k/\ell))^{-1}$. Moreover, if density of μ is uniformly bounded above by some constant $c' > 0$, then $\delta(\varepsilon) \leq c'\varepsilon$ for all $\varepsilon > 0$, and $\mathrm{E}(T_\ell) \geq \ell/(c'k^2)$.

2.3 The Case of Two Servers and Uniform Distribution

In the case of uniformly distributed requests for $k = 2$ servers, we show that the greedy policy is optimal and that the steady-state (expected) cost is $5/18 = 0.2777\ldots$.

Consider the greedy algorithm G and an arbitrary other algorithm A. Let C_n^G (C_n^A) be the cost incurred by algorithm G (resp., algorithm A) when serving the nth request, and let Z_n^G (resp., Z_n^A) be the minimum distance between the servers immediately after serving the nth request (with $Z_0 = Z_0^A = Z_0^G = d(S_0(1), S_0(2))$).

Theorem 3 *If the requests are independent and uniformly distributed on $\mathbb{T}$, then the greedy algorithm is optimal in that, for any algorithm A,*

$$\mathrm{E}\left[\sum_n C_n^G \;\middle|\; Z_0\right] \leq \mathrm{E}\left[\sum_n C_n^A \;\middle|\; Z_0\right],$$

regardless of the initial position Z_0 of the two servers.

The optimality of the greedy policy for $k = 2$ follows from the fact that serving each new request with the closest server minimizes the cost to serve the request, while at the same time maximizing the distance between the servers, resulting in a better covering of the space. The proof will appear in the full version of this work.

Finally, we examine the cost distribution induced by the greedy algorithm. Let $(Y_n)_{n\geq 0}$ denote the Markov chain defined by the clockwise distance between the two servers. Due to the rotation-invariance of the uniform distribution, the following simple recurrence formula holds for the process $(Y_n)_{n\geq 0}$,

$$Y_{n+1} = \frac{1}{2}Y_n + \varepsilon_{n+1}, \quad \text{all } n \geq 1,$$

where the $(\varepsilon_n)_{n\geq 1}$ are i.i.d. Uniform$[0,1]$. This is exactly the statistical setting of an auto-regressive AR(1) process; see, e.g., [8]. In particular, under appropriate assumptions on the innovation sequence $(\varepsilon_n)_{n\geq 1}$, the Markov chain $(Y_n)_{n\geq 0}$ has a unique invariant distribution and it can be shown to satisfy an array of classical limit theorems, including the strong law of large numbers, the central limit theorem, the law of iterated logarithm, and so on. Here we establish some simple properties of the limiting distance $\lim_n Y_n$, and we compute the steady-state expected cost.

By induction, for every $n \geq 1$,

$$Y_n = \frac{Y_0}{2^n} + \sum_{k=0}^{n-1} \frac{\varepsilon_{n-k}}{2^k},$$

and, therefore,

$$Z = \lim_{n\to\infty} Y_n = \sum_{k=0}^{\infty} \frac{\varepsilon_{n-k}}{2^k}.$$

Hence, the characteristic function of Z satisfies the functional equation,

$$\phi(\theta) = \phi\left(\frac{\theta}{2}\right)\phi_U(\theta) = \prod_{k=0}^{\infty} \phi_U\Big(\frac{\theta}{2^k}\Big),$$

where $\phi_U(\theta) = \frac{e^{i\theta}-1}{i\theta}$ is the characteristic function of the uniform distribution in $[0,1]$.

The expected cost in this case can be computed explicitly as,

$$\mathrm{E}[C] = \mathrm{E}[\mathrm{E}[C|Z]] = \mathrm{E}\left[\int_0^{Z/2} x\,\mathrm{d}x + \int_0^{(2-Z)/2} x\,\mathrm{d}x\right] = \frac{1}{4}\mathrm{E}[Z^2] - \frac{1}{2}\mathrm{E}[Z] + \frac{1}{2},$$

and substituting the values of the first and second moments of Z gives the expected cost as,

$$E(C) = \frac{5}{18} = 0.27777\ldots.$$

From the characteristic function of Z we can also obtain additional information. Noting that, for all $k \geq 1$, $\phi(\theta)$ can be rewritten as,

$$\phi(\theta) = e^{i\theta} \prod_{k=0}^{\infty} \frac{\sin(\theta/2^{k+1})}{\theta/2^{k+1}},$$

it is easily verified that $|\theta|^k\phi(\theta)$ is integrable and that then Z admits a density f which is infinitely differentiable and satisfies,

$$f(x) = \frac{1}{\pi}\int_0^{+\infty} \cos(\theta(1-x)) \prod_{k=0}^{\infty} \frac{\sin(\theta/2^{k+1})}{\theta/2^{k+1}} d\theta$$

Moreover, the derivative of f equals,

$$f'(x) = \frac{1}{\pi}\int_0^{+\infty} \theta\sin(\theta(1-x)) \prod_{k=1}^{\infty} \frac{\sin(\theta/2^{k+1})}{\theta/2^{k+1}} d\theta.$$

From these expressions we deduce that, for every $x \in [0,2]$, $f(x) = f(2-x)$ and that the function f' vanishes at $x = 1$. A closed formula for the density distribution appears difficult to obtain.

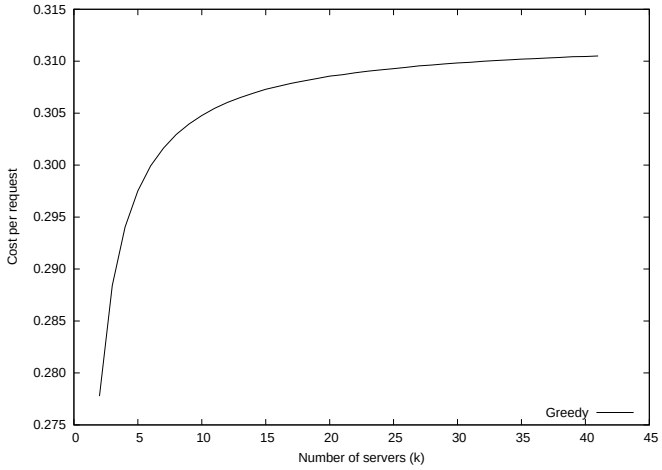


Fig. 1: The expected cost of serving a request as a function of k.

3 Open Problems

We are interested in the mean cost to serve a request when it arrives with distribution μ. When the k servers are in position $\mathbf{p} = (p_1, \ldots, p_k)$, the mean cost to serve a new request is given by,

$$\int_{\mathbb{T}} \left(\min_{i=1,\ldots,k} d_0(x, p_i) \right) d\mu(x),$$

and the asymptotic cost is,

$$C(\mu, k) = \int_{\mathbb{T}^k} \int_{\mathbb{T}} \left(\min_{i=1,\ldots,k} d_0(x, p_i) \right) d\mu(x) d\pi^k(p_1, \ldots, p_k).$$

- Scaling the problem with respect to k, so that k servers move on a circle of circumference k, how does $C(\mu, k)$ behave as a function on k? Figure 1 gives simulation results for the uniform distribution, showing that $C(\text{uniform}, k)$ is monotonically increasing with k.
- When $k = 2$, is it possible to characterize the steady-state distribution π when μ is not the uniform distribution?
- Which parts of the above analysis extend to the case of servers moving on the surface of a ball?

A Proof of Theorem 1

Recall that a Markov chain with state space Σ and transition kernel $\mathrm{P}(\cdot, \cdot)$ satisfies *Doeblin's condition* [8, page 391] if there exists a probability measure λ on Σ with the property that for some $m \geq 1$ and $\varepsilon \in (0, 1)$,

$$\mathrm{P}^m(\mathbf{p}, S) \geq \varepsilon \lambda(S),$$

for every $\mathbf{p} \in \Sigma$ and every (measurable) $S \subset \Sigma$.

We define now the probability measure λ that we will use. For $\delta < 1/8$ arbitrary but fixed, define the intervals $I_0 = [k-\delta, k) \cup [0, \delta]$ and $I_i = [i-\delta, i+\delta]$ for $1 \leq i \leq k-1$. Let $(Y_i)_{0 \leq i \leq k-1}$ be independent random variables, with each Y_i distributed according to $\mu(\cdot|I_i)$. Then λ is the probability measure on Σ induced by the joint distribution of $Y_0, Y_1, Y_2, \ldots, Y_{k-1}$, so that, in particular, for any set S of the form $S = S_0 \times S_1 \times \cdots \times S_{k-1} \subset \Sigma$, where each $S_i \subset \mathbb{T}$:

$$\lambda(S) = \prod_{i=0}^{k-1} \frac{\mu(S_i \cap I_i)}{\mu(I_i)}. \tag{A.1}$$

Notice that for the verification of the Doeblin condition it suffices to consider sets of the form $S = S_0 \times S_1 \times \cdots \times S_{k-1} \subset \Sigma$, where each $S_i \subset I_i$.

To that end we we need to show that, for any initial position $\mathbf{p}$ of the servers, there is a sequence of requests that has probability at least $\varepsilon\lambda(S)$, such that the greedy algorithm sends every server to a final position $\mathbf{p}' \in S$ in exactly m moves. We will establish this by describing a procedure (defined by a sequence of requests) that leads the greedy algorithm to move the servers to S.

The basic steps of the procedure are the following:

1. Send each server i to the corresponding interval I_i;
2. Perform some additional moves until $m - k$ moves have been made;
3. Send each server to its final position in k moves.

Step 2 is necessary, since Doeblin's condition requires that the greedy algorithm reaches the final position in *exactly* m moves. Below, each step is described and analyzed in detail; first recall that $\mathbf{p} = (p_0, p_1, \ldots, p_{k-1})$, where each p_i is the position of the ith server, and define the clockwise "distance" function,

$$\bar{d}(x, y) = \begin{cases} y - x & \text{if } x \leq y \\ k + y - x & \text{otherwise.} \end{cases}$$

We also define:

- $\bar{d}_i = \bar{d}(p_i, p_{i+1})$, the distance between servers i and $i+1$.
- A function A that maps two points of the circle to the interval between them,

$$A(x, y) = \begin{cases} [x, y) & \text{if } x \leq y \\ [x, k) \cup [0, y) & \text{otherwise.} \end{cases}$$

- The point-to-interval distances,

$$\bar{d}(x, I) = \inf_{y \in I} \bar{d}(x, y)$$
$$\bar{d}(I, x) = \inf_{y \in I} \bar{d}(y, x).$$

Finally, let $c > 0$ be a constant such that the density f of μ with respect to the Lebesgue measure on $\mathbb{T}$ satisfies, $\inf_{x\in\mathbb{T}} f(x) \geq c$.

Step 1.

Lemma 1 *For any initial position of the servers, the greedy algorithm will send each server i to I_i in at most $12k^2 - 8k$ moves with probability at least $\left(\frac{c}{4k}\right)^{12k^2-9k}\left(\frac{2\delta c}{k}\right)^k$.*

Proof: We describe a procedure defined by a sequence of requests for which the greedy algorithm sends each server i to I_i. The probability that this procedure will be followed is at least $\left(\frac{c}{4k}\right)^{12k^2-9k}\left(\frac{2\delta c}{k}\right)^k$.

Procedure $\mathcal{P}$:

1.1. Send server 0 to I_0
1.2. Send each server i to I_i

We describe and analyze each step separately.

Step 1.1

loop
  if $p_0 \in I_0$ **then**
    end
  else if $I_0 \subset A(p_0, p_0 + \bar{d}_0/2)$ **then** {move of type 1 (see Figure 2(a))}
    send server 0 to I_0 {probability $\geq (2\delta c)/k$}
  else {move of type 2 (see Figure 2(b))}
    $\ell \leftarrow \arg\max_i \bar{d}_i$ {therefore $\bar{d}_\ell \geq 1$}
    send server ℓ to $A(p_\ell + \bar{d}_\ell/4, p_\ell + \bar{d}_\ell/2)$ {probability $\geq c/4k$}
  end if
end loop

Lemma 2 *In at most $8k^2 - 4k$ moves and with probability at least $\left(\frac{c}{4k}\right)^{8k^2-4k-1} \cdot \frac{(2\delta c)}{k}$, step 1.1 will send server 0 to I_0.*

Proof:
We define the following potential function:

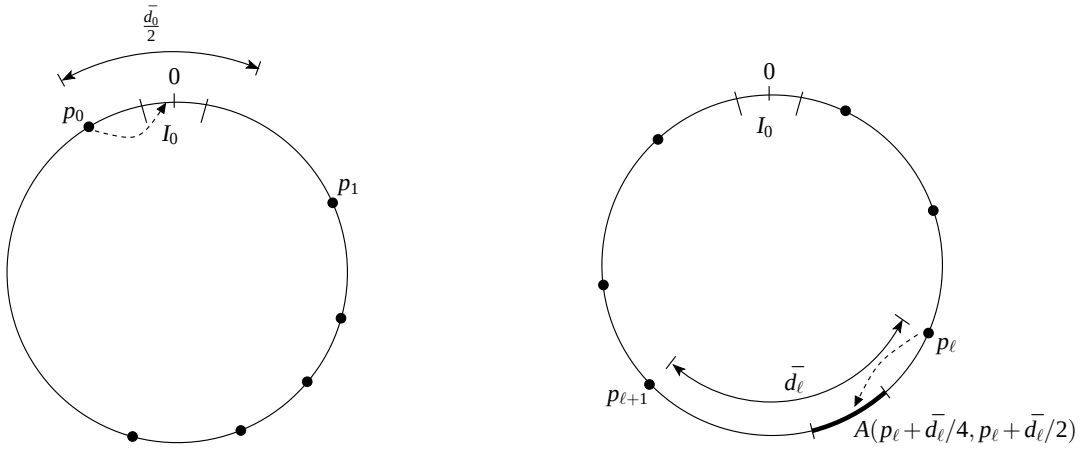
$$\phi(\mathbf{p}) = \mathbf{1}_{\{p_0 \notin I_0\}} \cdot \left(k \cdot \bar{d}(p_0, 0) + \sum_{i=1}^{k-1} \bar{d}(p_i, p_0) \right),$$

with $\mathbf{1}_A$ being the indicator function of event A. We will show the following facts:

Fact 1. $0 \leq \phi(\mathbf{p}) \leq 2k^2 - k$.

Fact 2. $\phi(\mathbf{p}) = 0$ if and only if $p_0 \in I_0$.

Fact 3. At each move, $\phi(\mathbf{p})$ either becomes 0 with probability at least $(2\delta c)/k$ and step 1.1 ends, or decreases by at least $1/4$ with probability at least $c/4k$.



(a) Move of type 1. The interval I_0 is entirely contained in the section of the circle served by server 0 according to the greedy algorithm, so server 0 will move inside I_0 with probability at least $(2\delta c)/k$.

(b) Move of type 2. The gap $\bar{d}_\ell$ is the maximum gap between two consecutive servers, so $\bar{d}_\ell \geq 1$. Thus, there is probability at least $(\bar{d}_\ell\, c)/4k \geq c/4k$ that a request will fall on the highlighted area $A(p_\ell+\bar{d}_\ell/4, p_\ell+\bar{d}_\ell/2)$, decreasing the potential ϕ by at least $\bar{d}_\ell/4 \geq 1/4$.

Fig. 2: Examples of moves in Step 1.1

Fact 1 holds since $\mathbf{1}_{\{p_0 \notin I_0\}} \in \{0,1\}$ and $0 \leq \bar{d}(\cdot,\cdot) \leq k$.

Fact 2 is obvious.

In order to show fact 3, we consider the two types of moves of the procedure described earlier. In the case of a type-1 move, the probability to be executed is $(2\delta c)/k$ and step 1.1 terminates. For a type-2 move we consider the following two cases:

- $\ell \neq 0$: In this case, notice that the total length of the circle is k and the total number of gaps is k. Therefore the length of the largest gap $\bar{d}_\ell$ is at least 1. Hence we have that the probability of the move is at least $c/4k$, and the distance $\bar{d}(p_\ell, p_0)$ decreases by at least $1/4$ (with the rest of the distances remaining the same), resulting to a decrease of $\phi(\mathbf{p})$ by at least $1/4$.
- $\ell = 0$: Again using the previous reasoning, the move is performed with probability at least $c/4k$. If server 0 moves into I_0, $\phi(\mathbf{p})$ becomes zero (and notice that the probability $c/4k \geq (2\delta c)/k$ for $\delta \leq 1/8$) and step 1.1 ends. Otherwise, assume that server 0 moves by $t \geq 1/4$. Then $\bar{d}(p_0, 0)$ decreases by t and every other distance $\bar{d}(p_i, p_0)$ increases by t. So the value of $\phi(\mathbf{p})$ decreases by:

$$k \cdot t - \sum_{i=1}^{k-1} t = kt - (k-1)t = t \geq 1/4.$$

From the above facts we deduce that the total number of moves that will be performed cannot be more than

$$\frac{2k^2 - k}{\frac{1}{4}} = 8k^2 - 4k,$$

and they all have probability at least $c/4k$ except possibly for the last one, which has probability at least $(2\delta c)/k$. Hence the probability of the first server moving inside I_0 is at least:

$$\Big(\frac{c}{4k}\Big)^{8k^2-4k-1} \cdot \frac{2\delta c}{k}.$$

□

Step 1.2.
To send each server i to I_i, we define the following procedure:

Procedure $\mathcal{P}_2(p_s, p_{s+1}, \ldots, p_t)$:
Input: $p_s, p_{s+1}, \ldots, p_t$ $\quad p_s \in I_s, p_t \in I_t$
Output: $p'_s, p'_{s+1}, \ldots, p'_t$ $\quad p_i \in I_i, i = s, s+1, \ldots, t$
if $t = s$ **then**
$p'_s \leftarrow p_s$
end
else if $t = s + 1$ **then**
$p'_s \leftarrow p_s$
$p'_t \leftarrow p_t$
end
else if there exists a $j, s < j < t$ such that $p_j \in I_j$ **then**
$\mathcal{P}_2(p_s, p_{s+1}, \ldots, p_j)$
$\mathcal{P}_2(p_j, p_{j+1}, \ldots, p_t)$
end
else
find a *good* server j {to be defined next}
send server j to I_j {using the procedure we describe next}
$\mathcal{P}_2(p_s, p_{s+1}, \ldots, p_j)$
$\mathcal{P}_2(p_j, p_{j+1}, \ldots, p_t)$
end
end if

We can now define what we mean by a "good" server; we say server j is good if there are no other servers between server j and the corresponding interval I_j, and the rest of the servers are all sufficiently far from I_j. An example is shown in Figure 3. Formally, we define:

A server j is *good* if, for every $i \neq j$, either $p_i \notin A(p_j, j)$ and $\bar{d}(I_j, p_i) > 1 - 2\delta$ (left type), or $p_i \notin A(j, p_j)$ and $\bar{d}(p_i, I_j) > 1 - 2\delta$ (right type).

Lemma 3 *Let $0 \leq s < t \leq k$ $(\equiv 0 \pmod k)$, with $p_s \in A(k - \delta, s + \delta)$ and $p_t \in A(t - \delta, \delta)$. Then there exists a good server $j \in \{s + 1, \ldots, t - 1\}$.*

Proof: The proof is by induction on the distance $t - s$. The base cases $t - s = 2$ and 3 are easy to verify by enumerating all possible position situations of the server(s) between s and t.

Assume now that the lemma holds for all $r < t - s$. We will show that it also holds for $t - s$. We consider the following three cases:

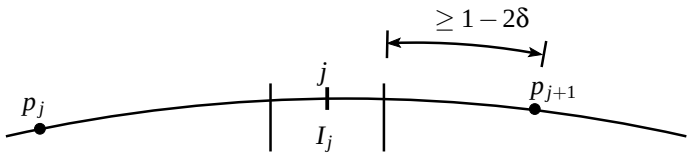


Fig. 3: An example of a *good* server of left type. There are no servers between server j (at position p_j) and interval I_j, and the distance (to the right) of all the other servers from I_j is higher than $1-2\delta$.

1. $p_{s+1} \notin A(k-\delta, s+1+\delta)$: Then notice that there is no server between $s+1$ and p_{s+1}, and we have $s+1-\delta-p_s > s+1-\delta-(s+\delta) = 1-2\delta$, and thus, $\bar{d}(p_s, I_{s+1}) > 1-2\delta$ (the same holds trivially for the rest of the servers), which means that server $s+1$ is a good server (of right type), and so we select $j = s+1$.

2. $p_{t-1} \notin A(t-1-\delta, \delta)$: Similarly to the first case, we have that server $t-1$ is a good server (of left type) and we select $j = t-1$.

3. $p_{s+1} \in A(k-\delta, s+1+\delta)$ and $p_{t-1} \in A(t-1-\delta, \delta)$: Then by the induction hypothesis we know that there exists a $j' \in \{s+2, \ldots, t-2\}$ that satisfies the conditions and hence we can select $j = j'$.

□

Having proven Lemma 3, we can now describe the procedure that sends the good server j to I_j:

if j is a left-type good server **then**
  loop
    if $p_j \in I_j$ **then**
      end
    else if $I_j \subset A(p_j, p_j + \bar{d}_j/2)$ **then**
      send server j to I_j {probability $\geq (2\delta c)/k$}
    else
      send server j to $A(p_j + \bar{d}_j/4, p_j + \bar{d}_j/2)$ {probability $\geq c/4k$}
    end if
  end loop
else {j is a right-type good server}
  loop
    if $p_j \in I_j$ **then**
      end
    else if $I_j \subset A(p_j - \bar{d}_{j-1}/2, p_j)$ **then**
      send server j to I_j {probability $(2\delta c)/k$}
    else
      send server j to $A(p_j - \bar{d}_{j-1}/2, p_j - \bar{d}_{j-1}/4)$ {probability $\geq c/4k$}
    end if
  end loop
end if

Notice that, at each move, either j enters I_j (with probability at least $(2\delta c)/k$) or moves by a distance of at least $\bar{d}_j/4 \geq 1/4$ (or $\bar{d}_{j-1}/4 \geq 1/4$) with probability at least $c/4k$ and remains a good server. Since the distance between j and I_j cannot be more than k, the total number of moves required to move j into I_j is bounded by $4k$ and their probability is at least:

$$\left(\frac{c}{4k}\right)^{4k-1} \cdot \frac{2\delta c}{k}.$$

Also note that we can execute the procedure $\mathcal{P}_2$ for $s = 0$, $t = k\ (\equiv 0 \pmod{k})$ in order to send all the servers to their corresponding intervals.

Lemma 4 *In at most $4k^2 - 4k$ moves and with probability at least $\left(\frac{c}{4k}\right)^{(4k-1)(k-1)} \left(\frac{2\delta c}{k}\right)^{k-1}$, step 1.2 will send each server $i \neq 0$ to I_i.*

Proof: The procedure sending the good server j to I_j will be executed at most $k-1$ times. Each time requires at most $4k$ moves and takes place with probability at most $\left(\frac{c}{4k}\right)^{4k-1} \cdot \frac{2\delta c}{k}$. □

Combining Lemmas 2 and 4 completes the proof of Lemma 1. □

Step 2.

For any possible initial configuration of the servers and any sequence of requests complying with step 1, the k moves of step 2 will be performed. However, as mentioned above, in order to satisfy Doeblin's condition we must show that we can reach the final configuration in exactly m moves, but for step 1 we only gave and upper bound of $12k^2 - 8k$ for the total number of moves. Hence, additional moves may be required in order to reach exactly that bound. The additional moves that we allow, are induced by requests that are close to the intervals I_i. In particular, they may fall in

$$\left(k - \frac{1}{4}, k\right) \bigcup \left[0, \frac{1}{4}\right) \bigcup \left[\bigcup_{i=1}^{k-1} \left(i - \frac{1}{4}, i + \frac{1}{4}\right)\right]$$

It is easy to see that for any sequence of requests that falls in the above set, none of the servers moves far away from its interval, which will allow them at the end to move close to their final positions.

Each request takes place with probability $c/2$, which is higher than any single request of step 1, and consequently the lower bound on the probability that we gave at step 1, holds even when fewer than $12k^2 - 8k$ moves are performed during step 1 and the rest are performed during the current step.

Step 3.

The last k requests will send the servers to a final configuration in $S = S_0 \times S_1 \times \cdots \times S_{k-1}$. Specifically, every server i will enter $S_i \subset I_i$, as we argued in step 2.

Consider now the probability that the last k requests send each server i to S_i. The total number of permutations for the order to send the servers is $k!$ and for each of them the probability that it happens is, $\prod_{i=0}^{k-1} \mu(S_i)$, so the total probability is

$$k! \prod_{i=0}^{k-1} \mu(S_i) \geq k! \prod_{i=0}^{k-1} \mu(S_i \cup I_i) \geq k! \left(\frac{2\delta c}{k}\right)^k \cdot \prod_{i=0}^{k-1} \frac{\mu(S_i \cup I_i)}{\mu(I_i)} = k! \left(\frac{2\delta c}{k}\right)^k \lambda(S),$$

using (A.1).

Taking into account all the moves of the three steps and the corresponding probabilities, we have that Doeblin's condition holds for

$$m = 12k^2 - 7k$$

and

$$\varepsilon = k! \left(\frac{c}{4k}\right)^{12k^2-9k} \left(\frac{2\delta c}{k}\right)^{2k}.$$

Acknowledgements

The authors are grateful to Guillaume Aubrun and Francis Comets for helpful discussions, as well as to the anonymous reviewers for their insightful comments.

References

[1] Billingsley, P. *Convergence of Probability Measures.* Wiley, New York, 1968.

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

[3] A. Dembo and O. Zeitouni. *Large Deviations Techniques and Applications*. Second edition. Springer-Verlag, New York, 1998.

[4] I. Kontoyiannis and S. P. Meyn. Spectral theory and limit theorems for geometrically ergodic Markov processes. *Annals of Applied Probability*, 13:304-362, February 2003.

[5] E. Koutsoupias and C. H. Papadimitriou. On the k-server conjecture. *Journal of the ACM*, 42(5):971–983, Sept. 1995.

[6] M. Manasse, L. McGeoch, and D. Sleator. Competitive algorithms for on-line problems. In R. Cole, editor, *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing*, pages 322–333, Chicago, IL, May 1988. ACM Press.

[7] M. S. Manasse, L. A. McGeoch, and D. D. Sleator. Competitive algorithms for server problems. *Journal of Algorithms*, 11(2):208–230, June 1990.

[8] S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Springer-Verlag, London, New York, 1993.

[9] Norris J.R. *Markov Chains.* Cambridge Series in Statistical and Probabilistic Mathematics, 1998.