Proceedings of the 2016 Winter Simulation Conference
T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, eds.

SPEEDING UP PAIRWISE COMPARISONS FOR LARGE SCALE RANKING AND SELECTION

L. Jeff Hong

Department of Economics and Finance and Department of Management Sciences
City University of Hong Kong, Kowloon, HONG KONG

Jun Luo

Antai College of Economics and Management
Shanghai Jiao Tong University, Shanghai, CHINA, 200052

Ying Zhong

Department of Management Sciences
City University of Hong Kong, Kowloon, HONG KONG

ABSTRACT

Classical sequential ranking-and-selection (R&S) procedures require all pairwise comparisons after collecting one additional observation from each surviving system, which is typically an $O(k^2)$ operation where k is the number of systems. When the number of systems is large (e.g., millions), these comparisons can be very costly and may significantly slow down the R&S procedures. In this paper we revise KN procedure slightly and show that one may reduce the computational complexity of all pairwise comparisons to an $O(k)$ operation, thus significantly reducing the computational burden. Numerical experiments show that the computational time reduces by orders of magnitude even for moderate numbers of systems.

1 INTRODUCTION

Using simulation experiments to select the best system with the largest mean performance from a finite set of alternatives is known ranking-and-selection (R&S) problems in the simulation literature. Many procedures have been designed to solve this type of problems, which in general can be classified in two approaches: the Bayesian approach and the frequentist approach. Interested readers may refer to Chick (2006) and Kim and Nelson (2006b) for a comprehensive introduction to both Bayesian and frequentist procedures.

Many R&S procedures are developed to handle small-scale problems where the number of systems is typically less than $1,000$. For instance, the well-known KN and KN++ procedures in Kim and Nelson (2001) and Kim and Nelson (2006a), hereinafter called KN family procedures, are used to solve less than 500 systems in their papers. As pointed by Luo et al. (2015), this limitation is mainly due to the limited simulation power as implemented in a single-processor computing environment. As various parallel computing environments are ready for simulation experiments, it turns out that KN family procedures are capable of handling large-scale R&S problems with up to 10^7 systems as in Ni et al. (2015). Simulation studies in both Luo et al. (2015) and Ni et al. (2015) reveal that the pairwise comparison work in KN family procedures become a bottleneck when the number of systems is large and simulating one observation from each system is relatively fast. Taking the Master/Slave parallel framework in Luo et al. (2015) as an example, because all pairwise comparisons are done on the master, it is highly likely that observations generated by the slaves are queued in front of the master waiting for comparison operations while many slaves are idling for the master to assign the next simulation task.

The time for pairwise comparisons did not get much attention when fully sequential procedures were implemented in a single-processor computing environment. Most of these procedures care more on their effectiveness and efficiency, i.e., trying to make the right selection decision with fewer total number of samples while achieving the targeted probability of correct selection (PCS) criteria. This is reasonable since they often deal with small-scale problems and the simulation time of obtaining one observation is relatively longer than the computing time of performing a comparison. In Section 2, we will derive the computational complexity to see the detailed explanation.

In order to deal with the inefficiency caused by all pairwise comparisons, Ni et al. (2015) proposed an divide-and-conquer approach in which they divide the all surviving systems into m subsets, where m is the number of slaves (called workers in their paper), and make pairwise comparisons within each subset and conduct additional comparisons among the current best systems in each subset. To further improve the efficiency by reducing the communications between the master and slaves, they allow each slave to simulates a batch of samples from each system. Of course, they need to run a preliminary stage to estimate the simulation completion time of generating one observation from different systems to balance the workload on each slave. The divide-and-conquer approach using the Master/Slave structure is straightforward and easily implemented to solve large-scale R&S problems in a parallel computing environment, especially for independent systems without considering the use of common random numbers (CRN). This divide-and-conquer approach has also been investigated in Ni, Hunter, and Henderson (2013). However, different from their approach, in this paper, we propose another type of approach to re-investigating the pairwise comparison formula of the KN family procedures and to breaking the formula by introducing two ordering sequences. In fact, fully sequential procedures designed on this approach can not only be implemented in a parallel computing environment, but also be suitable for a single-processor environment.

The rest of this paper is organized as follows. In Section 2, we show the computational complexity of sampling and comparisons for KN procedure we demonstrate how to reduce the complexity of comparisons by reforming the screening formula. This allows us to design, in Section 3, a modified KN procedure that provides a higher PCS than the corresponding original procedure.. Section 4 presents some preliminary numerical results, followed by conclusion and future research in Section 5.

2 THE COMPUTATIONAL COMPLEXITY OF KN PROCEDURE

The KN family procedures belongs to fully sequential procedures in the indifference-zone framework which can be traced back at least to Bechhofer (1954) and Paulson (1964). Here we use KN procedure to illustrate our key points. It is worthwhile pointing out that KN procedure can be viewed as a generalization of Paulson's procedure in Paulson (1964) for the unknown and unequal variance case and KN procedure tends to be more efficient since it is incorporated with a tight bound given by Fabian (1974) rather than the large deviation results in Paulson's procedure.

We first introduce necessary notations. Suppose that there are k independent systems. Let $X_{i\ell}$ denote the ℓth output from system i, which follows a normal distribution with unknown mean $\mu_i = \mathbf{E}[X_{i\ell}]$ and unknown variance $\sigma_i^2 = \mathbf{Var}[X_{i\ell}]$. We assume that $X_{i\ell}$ are independent and identically distributed (i.i.d.) for $\ell = 1,2,\ldots$, and $X_{i\ell}$ and X_{jn} are independent for $i \neq j$. We consider that there is only one truly best system under the indifference-zone formulation. Without loss of generality, we let $\mu_1 - \delta \geq \mu_2 \geq \ldots \geq \mu_k$ where $\delta > 0$ is the indifference-zone parameter. Our goal is to select system 1 as the best with a PCS, $1-\alpha$.

The KN procedure requires first-stage samples to estimate the unknown variances. In order to take the benefit of using CRN, KN procedure compute the pairwise sample variance between systems i and j,

$$S_{ij}^2 = \frac{1}{n_0 - 1} \sum_{\ell=1}^{n_0} \left(X_{i\ell} - X_{j\ell} - [\bar{X}_i(n_0) - \bar{X}_j(n_0)] \right)^2,$$

where $n_0 \geq 2$ is the first-stage sample size.

At stage $r \geq n_0$, KN procedure will eliminate system j from I, where I is the set containing all surviving systems in contention, if

$$\bar{X}_j(r) < \bar{X}_i(r) - \max\left\{0, \frac{h^2 S_{ij}^2}{2\delta r} - \frac{\delta}{2}\right\}, \text{ for some } i \in I,\ i \neq j, \tag{1}$$

where $h^2 = (n_0 - 1)\left[\left(\frac{2\alpha}{k-1}\right)^{-2/(n_0-1)} - 1\right]$. Then, the maximum expected number of samples to make the elimination decision on system j in the worst-case scenario is $N_{\max} = \frac{h^2 S_{ij}^2}{\delta^2}$. When the first-stage sample size n_0 is relatively large, then by taking the Taylor's expansion on the term $\left(\frac{2\alpha}{k-1}\right)^{-2/(n_0-1)}$, which means the computational complexity of sampling one system is approximately $O(\log k)$.

Investigating elimination formula in Equation (1), we have several important findings. First, the use of CRN helps to shrink the triangular region (thus reducing the total sample size) in KN procedure because $\mathbf{E}[S_{ij}^2] = \sigma_{ij}^2 < \sigma_i^2 + \sigma_j^2$, when $\mathbf{Cov}[X_i, X_j] > 0$. However, it does not improve the worst-case sample size of each system, which is still $O(\log k)$ when n_0 is large.

Second, for each stage of the KN procedure (after the first stage), an observation is collected for each surviving alternative, which represents a worst-case computational complexity of $O(k)$. However, the pairwise comparisons among k systems require a total of $k(k-1)/2$ comparisons, which has a worst-case computational complexity of $O(k^2)$. Therefore, from the view of computational complexity of the KN procedure as $k \to \infty$, it is clear that the part of pairwise comparisons dominates the part of sampling, and the total worst-case computational complexity of the KN procedure is $O(k^2 \log k)$.

Even the computational complexity of pairwise comparison is higher than that of sampling, the computational cost of comparison has not been viewed as a critical problem in traditional R&S literature, which is often assumed to be negligible compared with the computational cost of simulating observations. One of the reasons is that the number of systems is typically quite small and simulation time of generating one observation often takes orders of magnitude longer than the time of comparison operations required. Until recently using parallel computing environments to handle large-scale R&S problems, this turns out to be a very critical issue, as discussed in Luo et al. (2015) and Ni et al. (2015). Suppose there are tens of thousands systems that will be solved in a parallel simulation environment with hundreds of processors in the Master/Slave structure, a small amount of comparison time in each round on the master could lead to a large total amount of idling time for all slaves waiting for next simulation job. As mentioned in both Luo et al. (2015) and Ni et al. (2015), this kind of waste could dramatically reduce the benefits from a parallel implementation.

In Ni et al. (2015), they proposed a divide-and-conquer method to handle this issue. In this paper, we take a different approach to dealing with this problem. It is worthwhile pointing out that our approach can improve the efficiency of the original KN procedure even implemented on a single-processor.

3 THE MODIFIED KN PROCEDURE

Before introducing the modified KN procedure, we first revisit the screening formula in Equation (1). Without considering CRN, we have $\sigma_{ij}^2 = \sigma_i^2 + \sigma_j^2$. Therefore, it motivates us to replace S_{ij}^2 in Equation (1) of the KN procedure by $S_i^2 + S_j^2$, where S_i^2 and S_j^2 denote the sample variances of systems i and j calculated using the first stage sample with a sample size of n_0. For $i \in I$, define

$$\begin{aligned} W_i^+(r) &= \bar{X}_i(r) + \frac{h^2 S_i^2}{2\delta r}, \\ W_i^-(r) &= \bar{X}_i(r) - \frac{h^2 S_i^2}{2\delta r}. \end{aligned}$$

Let $i_-^\star = \arg\max_{i\in I} W_i^-(r)$ at stage r. Then, it is clear that system j will be eliminated before reaching the maximum point of the triangular region if and only if

$$W_j^+(r) < W_{i_-^\star}^-(r) + \frac{\delta}{2}. \tag{2}$$

By doing so, we only compare all surviving systems with the current "best" $i_-^\star$ instead of all pairwise comparisons between any two surviving systems. Notice that, after collecting all stage r samples, update $i_-^\star$ is an $O(k)$ operation and use $i_-^\star$ to compare to all other alternatives is again an $O(k)$ operation. Therefore, we reduce the computational complexity of all pairwise comparisons from $O(k^2)$ to $O(k)$, and reduce the overall computational complexity of the KN procedure of $O(k^2\log k)$ to $O(k\log k)$. Notice that this cannot be further reduced under the KN framework, because it is also the order of the total sample size as $k\to\infty$.

Because the new procedure does not allow the use of CRN, we call it independent KN procedure and iKN procedure in short. The following is the more detailed procedure.

Procedure 1 (The iKN Procedure)

Step 0. **Setup:** Select confidence level $1-\alpha$, indifference zone δ and first-stage sample size $n_0 \geq 2$. Let

$$h^2 = (n_0-1)\left[\left(\frac{2\alpha}{k-1}\right)^{-2/(n_0-1)} - 1\right]. \tag{3}$$

Step 1. **Initialization:** Let $I = \{1,2,\ldots,k\}$ be the set of alternatives still in contention. Obtain n_0 observations $X_{i\ell}$, $\ell = 1,2,\ldots,n_0$, from system i, $i = 1,2,\ldots,k$. Compute that

$$S_i^2 = \frac{1}{n_0-1}\sum_{\ell=1}^{n_0}\left(X_{i\ell} - \bar{X}_i(n_0)\right)^2,$$

where $\bar{X}_i(n_0)$ is the first-stage sample mean of system i with n_0 observations. Let

$$N_i = \frac{h^2 S_i^2}{\delta^2},$$

If $n_0 > \max_{i\neq j}(N_i+N_j)$, then stop and select the system with the largest $\bar{X}_i(n_0)$ as the best. Otherwise, set the observation counter $r = n_0$ and go to **Step 2**.

Step 2. **Pre-screening:** Let

$$W_i^+(r) = \bar{X}_i(r) + \frac{N_i\delta}{2r}, \text{ and } W_i^-(r) = \bar{X}_i(r) - \frac{N_i\delta}{2r}$$

and

$$i_-^\star = \arg\max_{i\in I} W_i^-(r).$$

Step 3. **Screening:** Let $I^{\text{old}} = I$. For any $j \in I^{\text{old}}$, $j \neq i_-^\star$, if $r \leq N_j + N_{i_-^\star}$, then set

$$I^{d_1} = \left\{ j \in I^{\text{old}} : W_j^+(r) < W_{i_-^\star}^-(r) + \frac{\delta}{2} \right\}, \tag{4}$$

else if $r > N_j + N_{i_-^\star}$, then set

$$I^{d_2} = \left\{ i \in I^{\text{old}} : i = \arg\min_{j,i_-^\star}\left(\bar{X}_j(r), \bar{X}_{i_-^\star}(r)\right) \right\}. \tag{5}$$

Update the contention set $I = I^{\text{old}} \setminus (I^{d_1}\cup I^{d_2})$, and go to **Step 4**.

Step 4. **Stopping Rule:** If $|I| = 1$, then stop and select the system whose index is in I as the best. Otherwise, take one additional observation $X_{i,r+1}$ from each system $i \in I$ and update

$$\bar{X}_i(r) = \frac{r\bar{X}_i(r) + X_{i,r+1}}{r+1},$$

and set $r = r+1$. If $r > \max_{i \neq j}(N_i + N_j)$, then stop and select the system with the largest $\bar{X}_i(r)$ as the best. Otherwise, go to **Step 2**.

Remark 1 System $i^\star_-$ serves as a current "best" system at each stage after introducing the new screening formula in Equation (2). It is usually used to eliminate other inferior systems as shown in Formula (4). However, it does not mean that it will never be eliminated by other systems, as shown in Formula (5). In fact, Formula (5) mean the discrete process $\left\{\bar{X}_j(r) - \bar{X}_{i^\star_-}(r), r = n_0, n_0+1, \ldots\right\}$ has already exited the triangular region after passing the maximum point. In that case, we cannot use Formula (4) anymore, but simply compare sample mean values between $\bar{X}_j(r)$ and $\bar{X}_{i^\star_-}(r)$.

Remark 2 The trick can be applied to KN++ procedure with variance updating in a similar fashion by defining

$$W_i^+(r) = \bar{X}_i(r) + \frac{h^2 S_i^2(r)}{2\delta r}, \quad \text{and} \quad W_i^-(r) = \bar{X}_i(r) - \frac{h^2 S_i^2(r)}{2\delta r},$$

where $S_i^2(r) = \frac{1}{r-1}\sum_{\ell=1}^r \left(X_{i\ell} - \bar{X}_i(r)]\right)^2$.

Remark 3 The statistical validity of iKN procedure can be shown use the same way as for the original KN procedure. However, since we break the sample variance of difference between two systems into the sum of two individual sample variances, the PCS actually become a little bit higher than that in KN procedure. To see why, we recall that Equation (5) in Kim and Nelson (2001) with $c = 1$ as follows,

$$\mathbf{E}\left[\frac{1}{2}\exp\left\{\frac{-h^2}{2(n_0-1)} \cdot \frac{(n_0-1)S_{ik}^2}{\sigma_{ik}^2}\right\}\right] = \frac{\alpha}{k-1}. \tag{6}$$

Since that $\frac{(n_0-1)S_{ik}^2}{\sigma_{ik}^2}$ is a chi-squared random variable χ^2 with (n_0-1) degrees of freedom and $\mathbf{E}[\exp\{t\chi^2\}] = (1-2t)^{-(n_0-1)/2}$, we can solve h^2 as in Equation (3). In iKN procedure, we need to show that

$$\mathbf{E}\left[\exp\left\{\frac{-h^2}{2(n_0-1)} \cdot \frac{(n_0-1)(S_i^2+S_k^2)}{(\sigma_i^2+\sigma_k^2)}\right\}\right] \geq \frac{2\alpha}{k-1}.$$

Notice that

$$\begin{aligned}
&\mathbf{E}\left[\exp\left\{\frac{-h^2}{2(n_0-1)}\frac{\sigma_i^2}{\sigma_i^2+\sigma_k^2} \cdot \frac{(n_0-1)S_i^2}{\sigma_i^2} + \frac{-h^2}{2(n_0-1)}\frac{\sigma_k^2}{\sigma_i^2+\sigma_k^2} \cdot \frac{(n_0-1)S_k^2}{\sigma_k^2}\right\}\right] \\
&= \mathbf{E}\left[\exp\left\{\frac{-h^2}{2(n_0-1)}w \cdot \frac{(n_0-1)S_i^2}{\sigma_i^2}\right\}\right] \times \mathbf{E}\left[\exp\left\{\frac{-h^2}{2(n_0-1)}(1-w) \cdot \frac{(n_0-1)S_k^2}{\sigma_k^2}\right\}\right], \quad \left(w := \frac{\sigma_i^2}{\sigma_i^2+\sigma_k^2}\right) \\
&= \left(1 + \frac{h^2}{(n_0-1)}w\right)^{-(n_0-1)/2}\left(1 + \frac{h^2}{(n_0-1)}(1-w)\right)^{-(n_0-1)/2} \\
&= \left(1 + \frac{h^2}{(n_0-1)} + \frac{h^4}{(n_0-1)^2}w(1-w)\right)^{-(n_0-1)/2} \\
&\geq \left(1 + \frac{h^2}{(n_0-1)}\right)^{-(n_0-1)/2} = \frac{2\alpha}{k-1}.
\end{aligned}$$

Therefore, Procedure iKN is statistical valid. Indeed, its PCS is higher than that of the KN procedure.

4 PRELIMINARY NUMERICAL STUDIES

In the numerical experiments, we set the PCS $1-\alpha = 0.95$, the indifference-zone parameter $\delta = 0.01$ and the first-stage sample size $n_0 = 100$. We assume that all outputs from system i are i.i.d. with a normal distribution with mean μ_i and variance σ_i^2. The simulation output is simply taken by generating a normal random variable, which takes a magnitude of milliseconds in Java. The numerical experiment is conducted on a PC computer with Intel Core i7-3770 CPU and 8GB memory and the operating system is the 64-bit version of Windows 7. We consider the slippage configuration in which $\mu_1 = 0.01,\ \mu_2 = \cdots = \mu_k = 0$ and equal-variance $\sigma_i^2 = 0.03^2$ for all systems $i = 1,2,\ldots,k$.

We firstly compare KN procedure with iKN procedure with the number of systems varying over $k = a \times 10^3$ where $a = 1,2,\ldots,20$. We report the averaged CPU Time (measured in milliseconds) over 20 macro-replications. Since we only run 20 macro-replications, we do not report the PCS, which is always equal to 1 as both procedures select the true best system, system 1, in each macro-replication.

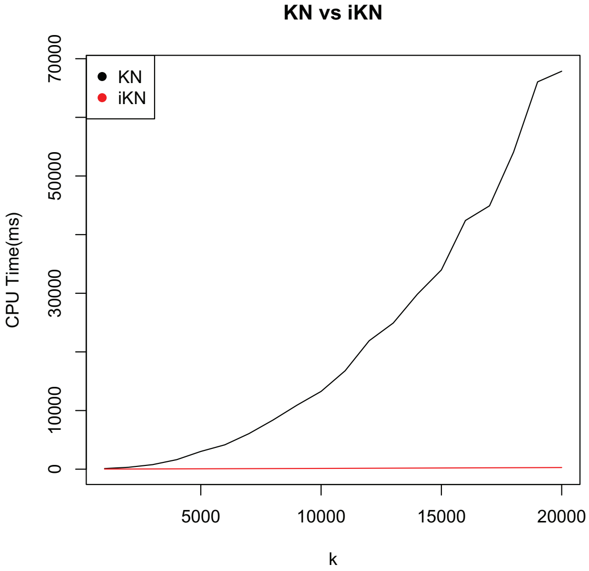

Figure 1: The average CPU Time for both KN and iKN.

Figure 1 plots the average CPU Time for both KN and iKN procedures with different k's, from which we find that the CPU Time for KN increases in a quadratic-like order as the number of systems k increases while the the CPU Time for iKN tends to be increased in a very flat slope. This is consistent with our theoretical analysis that the computational complexity of KN is $O(k^2 \log k)$. Even the computational complexity of iKN is $O(k \log k)$ (which is easier to see for larger values of k in Figure 2 later), Figure 1 shows that the improvement by iKN is quite significant. For instance, to solve a problem with $k = 2 \times 10^4$, KN needs about 68 seconds to select the best while iKN returns the result with less than one second (in fact, only about 0.28 second).

In order to obtain the relationship between the average CPU Time and the number of systems k for iKN, we test iKN with a larger number of systems varying over $k = a \times 10^4$ where $a = 1,2,\ldots,100$. Figure 2 shows that the CPU Time for iKN increases linearly with respect to the number of systems. Although the CPU Time is supposed to increase in the order of $O(k \log k)$, due to the fact that $k \log k$ increment will be eventually dominated by k rather than $\log k$ as k being large enough, that makes that the plot in Figure 2 looks like a linear increasing slope. It is also worthwhile pointing out that the CPU Time for iKN to solve

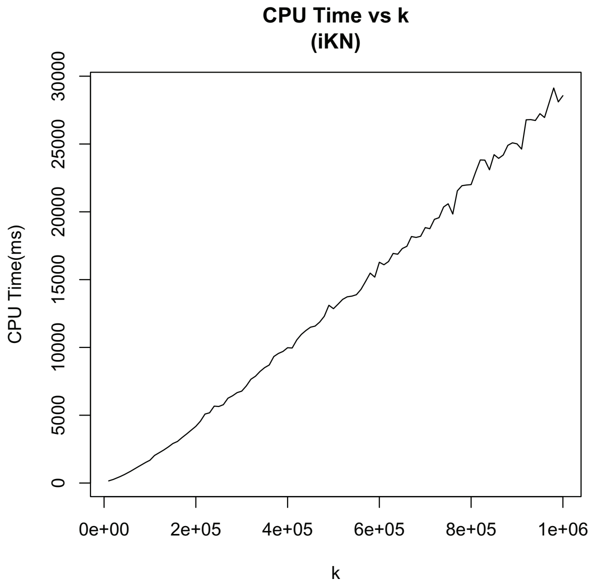

Figure 2: The average CPU Time for iKN.

a problem with $k = 10^6$ is about 30 seconds, which is still less than a half of that for KN to solve a problem with $k = 2 \times 10^4$.

As mentioned in Section 2, the computational complexity of sampling and that of comparison operations are in the same order of $O(k \log k)$, which results the computational complexity of iKN is $O(k \log k)$. To see that, we plot the CPU Time for iKN with respect to the total sample size in Figure 3. Figure 3 shows that the CPU Time for iKN increases linearly with respect to the total sample size. Since the total sample size increases in the order of $O(k \log k)$, an approximately linear relationship reveals that our analysis on the computational complexity of iKN is indeed $O(k \log k)$.

To sum up, we find that the computational complexity of KN is $O(k^2 \log k)$, which make KN become inefficient to solve large-scale R&S problems. However, this issue has been gently addressed as in iKN procedure, whose computational complexity is $O(k \log k)$. Preliminary numerical results show the significant advantage of iKN in solving large-scale problems.

5 CONCLUSIONS AND FUTURE RESEARCH

We studied the computational complexity of comparison operations and sampling of KN procedure, which become a critical issue when solving large-scale R&S problems with a parallel implementation. This is mainly caused by the $O(k^2)$ pairwise comparison operations in each stage. By modifying the screening formula, we require only comparisons between a current “best” system with all other inferior systems, which in return reduces the complexity of comparisons to at most $O(k)$ in each stage. In fact, it is highly possible that the modified procedure, iKN procedure, conducts no comparison operations during early stages. A preliminary numerical study shows the advantage of the iKN procedure compared with the original KN procedure.

We are also working on modifying iKN procedure so that it may be implemented on parallel computing environment and testing its efficiency with more simulation experiments.

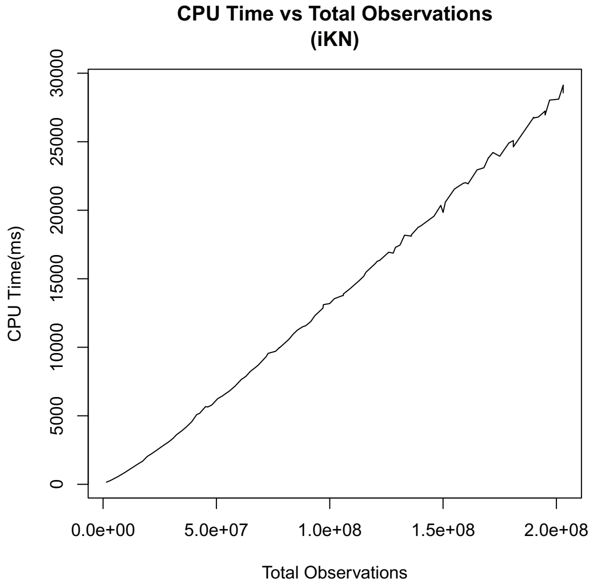

Figure 3: The average CPU Time v.s. the total sample size for iKN.

ACKNOWLEDGMENTS

The research of L. Jeff Hong was partially supported by Hong Kong Research Grants Council under project GRF 613012. The research of Jun Luo was partially supported by the Natural Science Foundation of China under Grant 71401104. The research of Ying Zhong was partially supported by Hong Kong PhD Fellowship Scheme.

REFERENCES

Bechhofer, R. E. 1954. “A single-sample multiple decision procedure for ranking means of normal populations with known variances”. *The Annals of Mathematical Statistics* 25:16–39.

Chick, S. E. 2006. “Subjective Probability and Bayesian Methodology”. In *Elsevier Handbooks in Operations Research and Management Science: Simulation*, edited by S. G. Henderson and B. L. Nelson, 225–257. Elsevier.

Fabian, V. 1974. “Note on Anderson’s sequential procedures with triangular boundary”. *The Annals of Statistics* 2:170–176.

Kim, S.-H., and B. L. Nelson. 2001. “A fully sequential procedure for indifference-zone selection in simulation”. *ACM Transactions on Modeling and Computer Simulation* 11:251–273.

Kim, S.-H., and B. L. Nelson. 2006a. “On the asymptotic validity of fully sequential selection procedures for steady-state simulation”. *Operations Research* 54:475–488.

Kim, S.-H., and B. L. Nelson. 2006b. “Selecting the best system”. In *Elsevier Handbooks in Operations Research and Management Science: Simulation*, edited by S. G. Henderson and B. L. Nelson, 501–534. Elsevier.

Luo, J., L. J. Hong, B. L. Nelson, and Y. Wu. 2015. “Fully sequential procedures for large-scale ranking-and-selection problems in parallel computing environments”. *Operations Research* 63 (5): 1177–1194.

Ni, E. C., D. F. Ciocan, S. G. Henderson, and S. R. Hunter. 2015. “Efficient Ranking and Selection in Parallel Computing Environments”. Technical report, Cornell University.

Ni, E. C., S. R. Hunter, and S. G. Henderson. 2013. “Ranking and selection in a high performance computing environment”. In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S.-H.

Kim, A. Tolk, R. Hill, and M. E. Kuhl, 833–845. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Paulson, E. 1964. "A Sequential Procedure for Selecting the Population with the Largest Mean from *k* Normal Populations". *The Annals of Mathematical Statistics* 35:174–180.

AUTHOR BIOGRAPHIES

L. JEFF HONG is a chair professor in the Department of Economics and Finance, and the Department of Management Sciences, at the City University of Hong Kong. His research interests include Monte Carlo methods, financial engineering, and stochastic optimization. He is currently an associate editor for *Operations Research, Naval Research Logistics and ACM Transactions on Modeling and Computer Simulation*. His email address is jeffhong@cityu.edu.hk.

JUN LUO is an assistant professor of Antai College of Economics and Management at Shanghai Jiao Tong University. His research interests include stochastic modeling and simulation, with their applications in service operations management and healthcare management. His email address is jluo_ms@sjtu.edu.cn.

YING ZHONG is a Ph.D student in Department of Management Science in the College of Business at City University of Hong Kong. His research interests include ranking & selection, discrete optimization via simulation and multi-armed bandit problems. His email address is yzhong4-c@my.cityu.edu.hk.