Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

OPTIMAL COMPUTING BUDGET ALLOCATION VIA SAMPLING BASED UNBIASED APPROXIMATION

Xiao Jin

Department of Industrial and Systems Engineering
National University of Singapore
10 Kent Ridge Crescent
119260, SINGAPORE

Haobin Li

Department of Computing Science
Institute of High Performance Computing
A*STAR Singapore
1 Fusionopolis Way #16-16 Connexis
138632, SINGAPORE

Loo Hay Lee
Ek Peng Chew

Department of Industrial and Systems Engineering
National University of Singapore
10 Kent Ridge Crescent
119260, SINGAPORE

ABSTRACT

In a Ranking and Selection problem, the objective of allocation is vital in deriving the rule. However, most of these objectives do not have a closed form. Due to the high cost of a direct approximation, several cheap but biased substitutes were applied to simplify the problem. These simplifications however could potentially affect the optimality of efficiency and therefore influence its finite performance. Fortunately, due to the increasing accessibility of parallel hardware (e.g. GPU), a direct approximation is becoming more tractable. Thus, we want to test the performance of an allocation rule based on an unbiased and direct approximation, expecting an acceleration on the performance. In this paper, we target on one of the famous objectives, the Probability of Correct Selection (PCS). Numerical experiments were done, showing a considerable improvement in finite performance of our algorithm comparing to a traditional one.

1 INTRODUCTION

Simulation optimization (sim-opt) seeks an efficient scheme to allocate limited computing budget to a finite or infinite number of solutions in order to find the best set of solution. Due to the stochastic feature of simulation, multiple trials are deployed to estimate the performance of each system. However, only a limited amount of computing budgets is at hand. How to allocate the total budget effectively becomes the core issue. Specifically, when the selection pool is a finite set and applying simulations to all solutions is possible, the problem becomes a Ranking and Selection (R&S) problem. Usually, due to the complex structure of its mathematical form, people are forced to apply approximation technique to solve the problem. These approximated forms are often cheaper to calculate. However, due to the existence of biases, these approximations may not approach the true optimality of efficiency. A direct approximation was considered before (Chen and Lee 2011) but was regarded as intractable in sequential processing. Fortunately, due to the increasing accessibility and decreasing cost of parallel hardware, such as GPU, applying Monte Carlo Method (MCM) to directly approximate the original problem becomes possible (Xu et al. 2015). With the

power of parallelizing, one can now rely on MCM to achieve an unbiased approximation (e.g. the Sample Average Approximation (SAA)) .

In this paper, we limit our scope within one of the allocation objective, namely, the Probability of Correct Selection (*PCS*). It is one of the most important objectives of budget allocation but was often automatically converted to Approximated PCS (*APCS*) (Chen, He, and Fu 2006, and LaPorte, Branke, and Chen 2012, for Optimal Computing Budget Allocation method; Chick et al. 2001 and Chick, Branke, and Schmidt 2007 for Expected Value of Improvement method; Kim and Nelson 2001 and Luo et al. 2015, for Indifference Zone method) derived by Bonferroni inequality for simplification. Frazier (2014) pushed forward this limit in Indifference Zone (IZ) method and decreased the total budget needed by applying a Bayes-inspired IZ to avoid using such a simplification. Similarly, we want to develop a new allocation rule based on a direct approximation avoiding using Bonferroni inequality. The efficiency gap between using *APCS* and a direct approximated *PCS* is our main stress.

2 BACKGROUND INFORMATION

2.1 Problem statement

Suppose we have k systems indexed by $i = 1, 2, \ldots, k$ and a measure J to describe their performance. The task is to select the system b with the optimal (e.g. the smallest) J_i. However, for each system, its J_i can only be observed by multi-times simulation. The simulation result $\tilde{J}_i$ can be regarded as a random variable. Here we assume that $\tilde{J}_i \sim \mathcal{N}(J_i, \sigma_i)$. We thus have the definition of the optimal system:

$$b \triangleq \arg\min_i \{J_i\} = \arg\min_i \left\{E[\tilde{J}_i]\right\}$$

with b unknown ahead. A reasonable selecting strategy is to select the solution with the best average performance after multiple simulations. For i^{th} system, its j^{th} independent observation is denoted as $J_{i,j}$. Therefore, when simulating i^{th} system for N_i times, the sample mean of simulation results $\bar{J}_i \sim \mathcal{N}(J_i, \frac{\sigma_i}{\sqrt{N_i}})$. The solution with the optimal sample mean is indexed as:

$$s = \arg\min_i \{\bar{J}_i\}.$$

Based on the above setting, a Correct Selection is defined as:

$$CS \triangleq \{\cap_{i \neq b} \bar{J}_b < \bar{J}_i\} = \{\bar{J}_b < \min_{i \neq b} \{\bar{J}_i\}\} = \{s = b\}.$$

Our goal is to maximize the probability of achieving Correct Selection. Let us first suppose that J_i and σ_i are known ahead. In this circumstances, when a limited amount of total budget is considered, we need an optimal allocation of $\vec{N} = (N_1, N_2, \ldots, N_k)$ that can maximize the Probability of Correct Selection (*PCS*) as possible. Therefore, the problem is formulated as:

$$\begin{aligned} \max_{\vec{N} \succeq 0} \; & PCS \triangleq P\{\bar{J}_b < \min_{i \neq b} \{\bar{J}_i\}\} \\ s.t. \; & \sum_i^k N_i = T. \end{aligned} \tag{1}$$

2.2 Limitation of APCS

We shall briefly introduce how *APCS* is derived and then illustrate its limitation.

The difficulty of optimizing *PCS* is that *PCS* does not have a closed form. Even with the hypothesis that $J_{i,j}$ are $i.i.d.$ normal, still, it is hard to calculate its value and gradient. This solid obstacle makes a direct manipulation on *PCS* difficult. The usual practice to tackle this is to instead optimize its lower bound, *APCS*. I is derived by Bonferroni inequality and is a lower bound to the real *PCS*:

$$APCS\text{-}B = 1 - \sum_{i \neq b} P\{\bar{J}_b \geq \bar{J}_i\} \leq 1 - P\{\cap_{i \neq b} \bar{J}_b \geq \bar{J}_i\} = PCS.$$

If we directly optimize this *APCS-B* by solving:

$$\begin{aligned} &\max_{\vec{N} \succeq 0} \ APCS\text{-}B \\ &s.t. \ \sum_{i=0}^{k} N_i = T \end{aligned} \tag{2}$$

then we get the Optimal Computing Budget Allocation (OCBA) (Chick, Branke, and Schmidt 2010). There is a technical simplification which makes the rule an asymptotic one: $N_b \gg N_i \ \forall i$. On this asymptotic basis, the programming system has an explicit solution:

$$\delta_{b,i} = J_b - J_i$$

$$\frac{N_i}{N_j} = \left(\frac{\sigma_i/\delta_{b,i}}{\sigma_j/\delta_{b,j}}\right)^2 \quad \forall i, j \neq b$$

$$N_b = \sigma_b \sqrt{\sum_{i \neq b} \frac{N_i^2}{\sigma_i^2}}.$$

The asymptotic feature points out that when the total budget T approaches infinity, the above allocation will be the best static ratio to increase the *PCS*. Its asymptotic optimality can also be proved by the theorem of Large Deviation (*LD*) (Glynn and Juneja 2004).

OCBA is inexpensive to calculate. However, certain limitations of such a rule cannot be ignored. First of all *APCS-B* ignores the correlation among multiple comparisons. It is clear that

$$P\{\bar{J}_b \geq \bar{J}_i | \bar{J}_b \geq \bar{J}_j\} P\{\bar{J}_b \geq \bar{J}_j\} \neq P\{\bar{J}_b \geq \bar{J}_i\} P\{\bar{J}_b \geq \bar{J}_j\}.$$

Although the impact caused by ignoring the correlation between the event $\{\bar{J}_b \geq \bar{J}_i\}$ and the event $\{\bar{J}_b \geq \bar{J}_j\}$ will be eased when more observations are received. However, in reality, the computing budget is usually scarce and making the early stage decision more important than the late stage. Such a distortion in the early stage may affect the increasing rate of inexplicit *PCS*, which makes an asymptotic rule far from enough. Secondly, since the approximation is biased and not tight on early stage, it can not be taken as an indicator of the current progress when the total budget is limited. Especially at the very beginning, *APCS-B* can be a negative number which has no meaning.

In general, the *APCS* has its limitation in representing the current progress. It is a biased approximation, which causes the information lost and could potentially deteriorate the finite performance when T is small.

3 METHODOLOGY

As mentioned, *APCS* has limitations in both measuring and optimizing the budget allocation. To push the limit forward, we apply a new Monte Carlo method. In this method, the *PCS* will be estimated by an unbiased approximation. When the approximation is accurate, the allocation that maximizes the approximated *PCS* is the one that maximizes the real *PCS*. Due to the computational intensity, parallel power will be adopted to make computation tractable.

3.1 Sample Average Approximation

First of all, we want to establish an unbiased approximation to the theoretical *PCS*. Be noticed that *CS* is an event. Since for any event $\mathscr{A}$

$$P(\mathscr{A}) = E[\mathbb{1}\{\mathscr{A}\}]$$

meaning that the probability equals the expected frequency of the event occurrence, this frequency can be treated as an unbiased approximation of the probability. Multiple independent experiments will be executed to observed the frequency of occurrence and decrease the estimation error. According to the Law of Large Number, this frequency will converge to $P(\mathscr{A})$ *a.s.*. In an practical sense, this means when the Monte Carlo sample size is huge, this portion can be regarded as an accurate representation of the real $P(\mathscr{A})$. By this rationale, such a frequency can be taken as a valid unbiased approximation to the probability. Since the average number is used, it is called Sample Average Approximation (SAA).

Since $CS = \{\bar{J}_b < \min_{i\neq b}\{\bar{J}_i\}\}$ and the distribution of $\bar{J}_i$ is known, we can execute M number of independent experiments and calculate the frequency of *CS* occurrence to derive our approximation. For example, in j^{th} experiment, we sample $m_{i,j} \sim \mathscr{N}(J_i, \frac{\sigma_i}{\sqrt{N_i}})$ once for each i. According to the definition of *CS*, the *CS* of the j^{th} experiment is defined as:

$$CS_j = \left\{ m_{b,j} < \min_{i\neq b}\{m_{i,j}\} \right\}.$$

Then we derive our SAA approximation:

$$\tilde{S} \triangleq \frac{\sum_{j=1}^{M} \mathbb{1}\{CS_j\}}{M}.$$

$\mathbb{1}(\cdot)$ is the indicator function. When *CS* occurs, it takes the value of 1. Otherwise, it is zero.

The calculation of the above quantity is time consuming. However, samplings are mutually independent. This is quite different from the simulation model where each observation is computed by multi-stage dependent calculation. This characteristic leads us to the application of parallel to bring the calculation tractable.

3.2 Optimal Computing Budget Allocation based on SAA

Now that the approximation is established, an optimal allocation can be derived by maximizing this approximation. For each valid allocation $\vec{N}$, its correspondent *PCS* can be estimated by $\tilde{S}$. Thus for each $\vec{N}$, we first estimate their respective *PCS*, then selecting the allocation that has the maximum $\tilde{S}$ value. If we denote $S_{\vec{N}}$ as the realized SAA of *PCS* under the allocation of $\vec{N}$, accordingly, we have the following optimization problem:

$$\begin{aligned} \max_{\vec{N}} \ & S_{\vec{N}} \\ s.t. \ & \langle \vec{N}, \mathbf{1} \rangle = T \\ & \vec{N} \in \mathbb{N}^k. \end{aligned} \tag{3}$$

Our goal is to select the optimal $\vec{N}^*$ with the biggest SAA value. As we see, this problem is itself a Ranking and Selection problem:

$$\vec{N}^* = \arg\max_{\vec{N}} E[\tilde{S}_{\vec{N}}].$$

Yet, it is different from a traditional R&S in that the budget limit here is no longer a fixed number of total observations. As we plan to apply MCM on a GPU, each sampling of a specific configuration is equivalent

to one observation of the original problem. When $\tilde{S}_{\vec{N}}$ for different $\vec{N}$ are computed in parallel, the total amount of time depends on the $\tilde{S}_{\vec{N}}$ that costs the longest time. Usually, given the same size of sampling for each $\tilde{S}_{\vec{N}}$ (e.g. the same M), the time cost should not vary much. This means that for calculation of $\tilde{S}_{\vec{N}}$ for different $\vec{N}$, it is optimal to execute the same M number of experiments in $\tilde{S}_{\vec{N}}$ calculation. Due to this unique feature of the parallel computing, we only need to keep the time cost of finishing M samplings within an acceptable range while letting M be as huge as possible.

Applying common random number is suggested. For each experiment of estimating a specific $\vec{N}$, we can first generate multiple independent standard normal results and do affine transformation to create a realization value of Monte Carlo sampling. We denote $w_{i,j}$ as our standard normal realization:

$$m_{i,j} = J_i + \frac{w_{i,j}\sigma_i}{\sqrt{N_i}}.$$

Thus, the CS_j is specified as

$$CS_j = \Big\{ J_b + \frac{w_{b,j}\sigma_b}{\sqrt{N_b}} < \min_{i \neq b} \{ J_i + \frac{w_{i,j}\sigma_i}{\sqrt{N_i}} \} \Big\}.$$

We shall launch M (large, e.g.1000) number of this independent trial and calculate the frequency of the occurrence of CS. We have the following form of SAA by plugging in the redefined CS:

$$\tilde{S}_{\vec{N}} = \frac{1}{M} \sum_{j=1}^{M} \mathbb{1}\Big\{ J_b + \frac{w_{b,j}\sigma_b}{\sqrt{N_b}} < \min_{i \neq b} \{ J_i + \frac{w_{i,j}\sigma_i}{\sqrt{N_i}} \} \Big\}.$$

Since M is the same, $\mathbf{W}^{k \times M} = \{w_{i,j}\}$ are reused for different $\vec{N}$. This application of common random number can not only save the total computation, but also increase the probability of selecting the optimal allocation. Therefore, we derive the following form:

$$\begin{aligned} \max_{\vec{N} \succeq 0} \ & \sum_{j=1}^{M} \mathbb{1}\Big\{ \min_{i \neq b} \{ J_i + \frac{w_{i,j}\sigma_i}{\sqrt{N_i}} \} > J_b + \frac{w_{b,j}\sigma_b}{\sqrt{N_b}} \Big\} \\ s.t. \ & \langle \vec{N}, \mathbf{1} \rangle = T \\ & \vec{N} \in \mathbb{N}^k. \end{aligned} \tag{4}$$

The last issue is the estimation of parameters. In reality, J_i, σ_i and b are all unknown. Instead, we rely on their estimators to derive the allocation rule. The b index is unknown, we take s as its representation. Such substitution is valid because as N_i approaches infinity, $\bar{J}_i$ will converge to J_i $a.s.$ (according to the Law of Large Number), making $s = b$ eventually. Similarly, J_i and σ_i are unknown, we use $\bar{J}_i$ and S_i instead. They will be arbitrarily accurate when we have a decent number of simulations.

A frequentist view of inference is adopted. We suppose that at a specific time point, the i^{th} solution has been simulated for N_{Ei} times. The distribution of $\tilde{\bar{J}}_i$, which is the sample mean after allocating further ΔN_i budget to i^{th} solution, can be derived. Based on a frequentist inference, each newly planed to be observed $J_{i,j} \sim \mathcal{N}(\bar{J}_i, S_i)$. Therefore, $\tilde{\bar{J}}_i \sim \mathcal{N}(\bar{J}_i, \frac{S_i}{\sqrt{N_{Ei}+\Delta N_i}})$. The final optimal allocation adopted in our algorithm is the solution of the following programming system:

$$\begin{aligned} \max_{\vec{\Delta N}} \ & \sum_{j=1}^{M} \mathbb{1}\Big\{ \min_{i \neq s} \{ \bar{J}_i + \frac{w_{i,j}S_i}{\sqrt{N_{Ei}+\Delta N_i}} \} > \bar{J}_b + \frac{w_{b,j}S_b}{\sqrt{N_{Eb}+\Delta N_b}} \Big\} \\ s.t. \ & \langle \vec{\Delta N}, \mathbf{1} \rangle = \Delta T \\ & \vec{\Delta N} \in \mathbb{N}^k \end{aligned} \tag{5}$$

where ΔT denotes a portion of the total budget T and N_{Ei} denotes the already allocated budgets to the solution i. After allocation, we simulate according to it and update $\tilde{J}_i$ and S_i after getting new simulation results. The detail pseudo code is as follows:

Initialization:
  Set M, ΔT, Generate $\mathbf{W}$;
  Allocate N_0 size of equal allocation;
  Calculate the initial $\bar{J}_i$ and S_i for all i;
while *T is not exhausted* **do**
  Allocation:
    Subtract ΔT from T;
    Solve (5) to get the optimal allocation;
  Simulation:
    Simulate according to the allocation;
    Update $\tilde{J}_i$, S_i and $\vec{N}_E$;
end

Algorithm 1: OCBA-SAA.

So, the basic idea is that after an initial stage, in each iteration, we take s as b and generate $\mathbf{W}$. Then we do affine transformation based on sample mean $\bar{J}_i$ and standard variance S_i to create each $m_{i,j}$. We then solve the optimization problem to get $\vec{N}$ for the following simulation session.

4 NUMERICAL EXPERIMENT

Numerical experiment is done to show the advantage of using an unbiased estimation in the situation where the total budget is scarce (e.g. T small). In this particular case, N_0 and ΔT are usually set small. Extremely, ΔT is set to be 1. In purpose of showing the differences, we need to compare our method with the one that maximize the *APCS*.

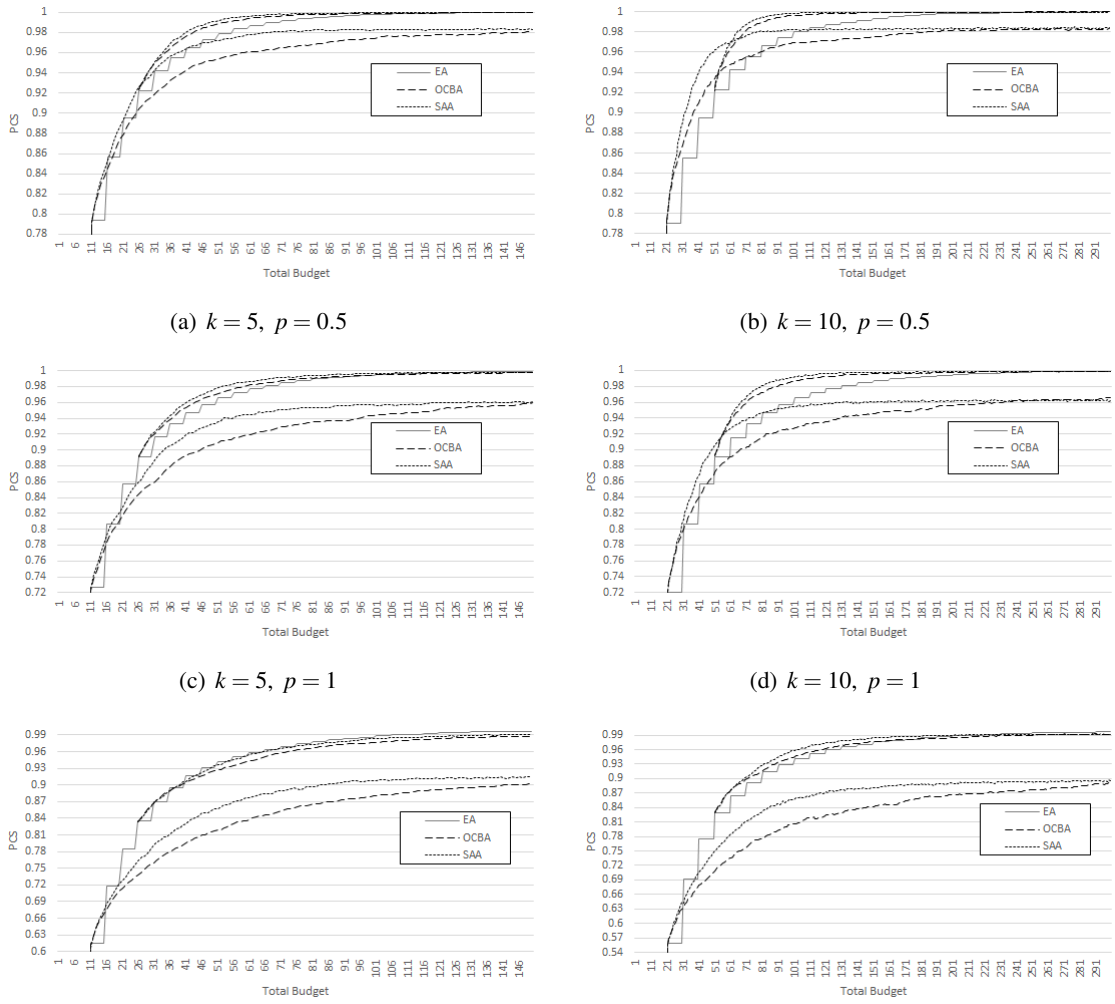
When the batch size $\Delta T = 1$, the classical OCBA turns to SOCBA (Chen, He, and Fu 2006), which intends to solve the following problem in each iteration:

$$\begin{aligned} &\max_{\vec{\Delta N}} \ \textit{APCS-B} \\ &s.t. \ \left\langle \vec{\Delta N}, \mathbf{1} \right\rangle = 1 \\ &\qquad \vec{\Delta N} \in \{0,1\}^k. \end{aligned}$$

This algorithm is exactly what we need for comparison since the other parts of the algorithm are identical to our method except its optimizing over the biased *APCS*.

The effectiveness of SAA and SOCBA are both heavily relying on how close $\bar{J}_i$ and S_i are to J_i and σ_i. In this sense, the N_0 is especially important. If SAA can dominate SOCBA in different scenario of N_0 (indicating different estimation accuracy), then we can conclude that applying more computation in *PCS* estimation is meaningful.

We use monotone-decreasing-means (Chick, Branke, and Schmidt 2007) to test the performance of our algorithm against SOCBA. It creates systems with the same interval between true means. The variance decreases from the worst system to the best system. Here, we set the variance as i^p, where i is the true rank of the system from the best to the worst and p a controllable parameter. When $p = 1$, the variance will be linear to the rank difference. When $p > 1$, the variance increases faster than the rank, *vice versa*. This configuration is especially designed to test the ability of utilizing both the mean and variance information.



(a) $k = 5,\ p = 0.5$ (b) $k = 10,\ p = 0.5$

(c) $k = 5,\ p = 1$ (d) $k = 10,\ p = 1$

(e) $k = 5,\ p = 1.5$ (f) $k = 10,\ p = 1.5$

Figure 1: The *PCS* according to the budget consumed under Monotone configuration.

Each trial runs until the *PCS* under Equal Allocation (EA) turns to 1. Since a sequential EA allocates one budget for all solution in each iteration, the trajectory of it is a step function. Since the initial stage is exactly EA, we compare SAA and OCBA rule in scenarios where N_0 is 2 and 5 respectively.

The *PCS* shown in the graphs are estimated by 1000 Monte Carlo Sampling knowing the true mean and variance. All results shown are averaged among 100 trials, meaning that each curve on the graph is the average of 100 results.

Through numerical results, we can generally conclude that SAA performs robustly better than OCBA when batch size is 1. Trends can be drawn from the Figure 1 that the smaller N_0 is, the larger the performance gap between using unbiased estimation versus a biased one be. Further, as a sized 10 problem performs better than a sized 5 in all configurations, one could expect that the gap is even larger when problem size increases.

5 CONCLUSION

In this work, we apply MCM to create an SAA to approximate *PCS* in an unbiased way. Then we derive an optimal allocation rule based on this approximation. The numerical experiments are conducted to examine

its performance against the algorithm relying on *APCS* when the batch size is 1. The result shows consistent advantages of our rule against the SOCBA rule in the presented scenarios. Through this result, we can partially conclude that our allocation based on estimated *PCS* is robust in performance. It can combine mean and variance information effectively, and leads to a more efficient finite time performance.

Still, more work needs to be done in effectively solving the problem (5). When problem size goes large or batch size increases, the total number of valid allocation will go faster than the exponential. This issue is essential for a practical application.

Other than solving the programming problem, another potential of this work is to broaden the simulation output distribution to a general one. When the observation results do not follow a normal distribution, SAA will be different leading probably to a new allocation rule.

ACKNOWLEDGMENTS

This work is supported by Ministry of Education, Singapore [Grant MOE2015-T2-2-005], dedicated to the study of sim-opt in multi-fidelity and parallel computing.

REFERENCES

Chen, C.-H., D. He, and M. Fu. 2006. "Efficient Dynamic Simulation Allocation In Ordinal Optimization". In *Proceedings of the 2006 Winter Simulation Conference*, edited by L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, 2005–2009. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Chen, C.-h., and L. H. Lee. 2011. "Stochastic Simulation Optimization: An Optimal Computing Budget Allocation". In *Proceedings of the 2011 Winter Simulation Conference*, edited by S. Jain, R. Creasey, J. Himmelspach, K. White, and M. Fu. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Chick, S. E., J. Branke, and C. Schmidt. 2007. "New Greedy Myopic And Existing Asymptotic Sequential Selection Procedures: Preliminary Empirical Results". In *Proceedings of the 2007 Winter Simulation Conference*, edited by S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 289–296. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Chick, S. E., J. Branke, and C. Schmidt. 2010. "Sequential Sampling To Myopically Maximize The Expected Value Of Information". In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Ycesan, 71–80. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Chick, S. E., K. Inoue, K. Inoue, and K. Inoue. 2001. "New Two-stage And Sequential Procedures For Selecting The Best Simulated System". In *Proceedings of the 2001 Winter Simulation Conference*, edited by B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 732–743. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Frazier, P. I. 2014. "A Fully Sequential Elimination Procedure For Indifference-zone Ranking And Selection With Tight Bounds On Probability Of Correct Selection". In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 926–942. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Glynn, P., and S. Juneja. 2004. "A Large Deviations Perspective On Ordinal Optimization". In *Proceedings of the 2004 Winter Simulation Conference*, edited by R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Kim, S.-H., and B. L. Nelson. 2001. "A Fully Sequential Procedure For Indifference-zone Selection In Simulation". In *Proceedings of the 2001 Winter Simulation Conference*, edited by B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 251–273. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

LaPorte, G. J., J. Branke, and C.-H. Chen. 2012. "Optimal Computing Budget Allocation For Small Computing Budgets". In *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, J. Himmelspach, O. R. R. Pasupathy, and A. M. Uhrmacher, 292. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

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". In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 1177–1194. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Xu, J., E. Huang, C.-H. Chen, and L. H. Lee. 2015. "Simulation Optimization: A Review And Exploration In The New Era Of Cloud Computing And Big Data". In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 1550019. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

XIAO JIN is a Ph.D candidate in the Department of Industrial and Systems Engineering, National University of Singapore. He received his B.S. (Industrial Engineering) degree from the Nanjing University in 2015. His research interests include simulation-based optimization, parallel computing and heuristic optimization. His e-mail address is e0000925@u.nus.edu.

HAOBIN LI is Scientist for the Institute of High Performance Computing, A*STAR Singapore. He received his B.Eng. degree (1st Class Honors) in 2009 from the Department of Industrial and Systems Engineering at National University of Singapore, with minor in computer science; and Ph.D. degree from the same department in 2014. He has research interests in operations research, simulation optimization and designing high performance optimization tools with application on logistics and maritime studies. lihb@ihpc.a-star.edu.sg.

LOO HAY LEE is an Associate Professor and Deputy Head (Graduate Studies and Research) in the Department of Industrial and Systems Engineering, National University of Singapore. He received his B.S. (Electrical Engineering) degree from the National Taiwan University in 1992 and his S. M. and Ph.D. degrees in 1994 and 1997 from Harvard University. He is currently a senior member of IEEE, a committee member of ORSS, and a member of INFORMS. His research interests include production planning and control, logistics and vehicle routing, supply chain modeling, simulation-based optimization, and evolutionary computation. His email address is iseleelh@nus.edu.sg.

EK PENG CHEW is an Associate Professor and Deputy Head (Undergraduate Studies) in the Department of Industrial and Systems Engineering, National University of Singapore. He received his Ph.D. degree from the Georgia Institute of Technology. His research interests include logistics and inventory management, system modeling and simulation, and system optimization. His email address is isecep@nus.edu.sg.