Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

EPSILON OPTIMAL SAMPLING

Travis Goodwin
Jie Xu
Chun-Hung Chen

Dept. of Systems Engr. & Operations Research
George Mason University
4400 University Drive
Fairfax, VA, 22030, USA

Nurcin Celik

Dept. of Industrial Engineering
University of Miami
1320 S Dixie Hwy
Coral Gables, FL, 33146, USA

ABSTRACT

Epsilon Optimal Sampling (EOS) is a novel algorithm that seeks to reduce the computational complexity of selecting the best design using stochastic simulation. EOS is an Optimal Computing Budget Allocation (OCBA) type algorithm that reduces computational complexity by integrating machine learning (ML) models into the simulation optimization algorithm. EOS avoids the pitfall of trading computational overhead in simulation execution for computational overhead in ML model training by using a concept we call policy stability. In this paper, we present the concept of policy stability, how it can be used to improve dynamic sampling techniques, and how low-fidelity ML estimates can be integrated into the process. Numerical results are presented to provide evidence as to the improvement in computational efficiency that can be achieved when using EOS in conjunction with ML models over the standard OCBA algorithm.

1 INTRODUCTION

Stochastic simulations are desirable tools for use in modelling complex systems due to their ability to accommodate practically any parametric representation of real world systems, as well as complex interactions between systems and their environment, which otherwise would be impossible to express in a closed form manner. Some examples of such use cases are production planning in semiconductor manufacturing (Pickardy et al. 2010; Hsieh et al. 2007; Song et al. 2019; Zhang et al. 2020; Calverley et al. 2021), operational planning of power systems (Thanos et al. 2015; Xu et al. 2020; Yavuz et al. 2020), resource allocation in healthcare and other service systems (Kasaie and Kelton 2013; Chen and Wang 2016; Qiu and Song 2016), and transportation (Zhou et al. 2021). These simulations require a large amount of computational workload to execute, to a point where this limitation sometimes precludes the use of simulations for desired purposes, or severely limits the number of alternative systems or environmental variables considered in the simulation experimental design. The goal of simulation optimization is to reduce the computational overhead required to execute simulations to achieve a desired level of confidence in the statistical output when using simulations to select the best alternative from a finite number of designs or decisions. Epsilon Optimal Sampling (EOS) is a novel simulation optimization algorithm that seeks to provide a means to improve on existing algorithms by integrating low-fidelity machine learning (ML) estimates of the simulation output.

There are two traditional classes of simulation optimization algorithms, and a third unique class is emerging in recent published literature. The first class of traditional algorithms are the fixed-confidence approaches, often referred to as the frequentist approach to simulation optimization. These algorithms compute the required number of replications to achieve a fixed confidence level in the results from the output. Examples of such algorithms include Bechhofer (1954), the fully sequential Procedure KN (Kim and Nelson 2001), KN++ (Kim and Nelson 2006), and the two-stage procedure NSGS Algorithm (Nelson

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

et al. 2001). An example of a recent improvement to the fixed-confidence approach is seen in Chen (2011). Where-as the frequentist approach often leverages a "fixed-confidence" method (e.g., we compute the required replications to achieve a given level of confidence), Bayesian algorithms often leverage a "fixed-budget" method. A fixed-budget method is where the computational time is fixed, and we attempt to optimize an objective function about our uncertain decision. The expected value of information (EVI) presented in Chick and Inoue (2001) is one example, while another well known algorithm is the knowledge gradient (KG) policy from Frazier et al. (2008). Russo (2020) is a more recent publication which also falls under this category. Another fixed budget approach to sampling that is considered different from Bayesian approaches is the Optimal Computing Budget Allocation (OCBA) approach. Chen et al. (2000) first introduced OCBA type algorithms. Glynn and Juneja (2004) later proved the asymptotic optimality of such an approach to sampling in the context of optimizing the probability of correctly selecting the true best alternative with a fixed sampling budget. EOS is an OCBA-type algorithm, but adapted to incorporate low fidelity ML estimates to improve the rate at which this probability increases as the sampling budget is increased.

A third class of algorithms that has recently emerged is referred to as offline-simulation online-application (OSOA). This phrase was coined in Hong and Jiang (2019), and refers to the extensive use of simulations prior to the time at which a decision must be made (e.g., "offline"). The results of these simulations are used to to train machine learning models or heuristics to provide decision making recommendations at the time which a decision is required (e.g., "online"). The requirement for a timely decision recommendation for an "online" decision, vice using the simulation outcomes directly, is driven by the the existence of what is referred to as covariates; the problem is such that the decision maker or system is faced with a series of random covariates that change over time, and the decision maker or system must select the best decision at each moment when the state of the covariates change. The vector describing the state of all the pertinent covariates at some given time can also be referred to as the context of a decision. In these problems, a new decision is desired as the context changes because the utility of a decision is a function not just of the alternative selected, but also the state of the random covariates. Furthermore, the time between decisions is too small relative to the computational overhead of the simulation to run an extensive simulation based analysis of alternative decisions for each set of covariates. An example of this type of algorithm is given by Shen et al. (2021). An alternative approach to solving this type of problem is given in Goodwin et al. (2022). In this paper, SAMPLE is presented as an OCBA-type framework for integrating both simulation output and off-line ML models to provide decision recommendations in a dynamic environment using simulation optimization. Despite the significant improvements shown in the actual sampling process of the decision space, SAMPLE uses computationally intensive linear algebra operations that are unavoidable to fit a Gaussian mixture model to the ML estimates. This effectively prevents the practical implementation of what is referred to as "online learning"; the iterative training of an online ML model as more simulation data is gathered. EOS is an algorithm that was designed to overcome the hurdles presented in SAMPLE. Instead of a complex Gaussian mixture model, EOS uses a very simple and straightforward concept called policy stability to directly integrate offline ML estimates with simulation data, and updates how these estimates are used with a dynamic learning parameter that is computed at each iteration of sampling.

For the research presented here, we consider the case where the simulation budget is limited such that fixed-confidence methods are not permissible. Furthermore, we consider the case where low-fidelity ML estimates are available for use and integration with simulation output. We also assume that sampling occurs in a dynamic environment, where the total budget, ΔL, is incremented by one replication at each sampling iteration. In the event that $\Delta L \neq 1$, the required computations to approximate our exploration parameter, ε, become highly non-linear and intractable using the method presented here. Under these conditions, the goal of this paper is to present EOS as an alternative sampling algorithm for simulation optimization, and provide evidence that shows the potential computational gains that can be realized when using EOS either with or without low-fidelity ML estimates, as compared to traditional OCBA approaches. The structure of the paper is as follows; Section 2 discusses the logic of EOS and presents the algorithm itself. Section 3 present numerical results of the performance of EOS under typical test scenarios against traditional OCBA. We present our conclusions and a short discussion in Section 4.

2 EPSILON OPTIMAL SAMPLING

2.1 OCBA

Let $X_i,\ i = 1,2,...,n$ be a set of n design alternatives, and let $f(X_i)$ be the simulation output of some stochastic simulation. In the remainder of this paper, we will refer to $f(X_i)$ as f_i. Let $i^* = \arg\max_{i=1,2,...,n}(E(f_i))$; our goal is to select $\tilde{i}^*$ such that $\tilde{i}^* = i^*$, where $\tilde{i}^*$ is our estimate of the best alternative using the incomplete information that we observe from the simulation. Since the simulation output is random, our objective can only be achieved in probability. Assume that the simulation output is normally distributed; this is a canonical assumption in many algorithms, which can be guaranteed in most cases by using uncorrelated batch sampling. Furthermore, to obtain a closed-form implementation of the OCBA algorithm, we adopt a canonical assumption in the OCBA literature and assume $L^*_{\tilde{i}^*} \gg L^*_i \ \ \forall\ i \neq \tilde{i}^*$. This is equivalent to assuming that the difference $E(f_{i^*}) - E(f_i)$ is small enough relative to σ_i, σ_{i^*} for all i, i^* such that to differentiate i^* based on the observed mean, a majority of the computing budget must be allocated to this alternative. In practice, this presents a computationally efficient approximation to the true OCBA allocation ratio, which requires solving a set of nonlinear equations. The OCBA literature has provided extensive numerical evidence on the effectiveness of this approximation. Let $\bar{f}_i, \hat{\sigma}^2_i$ be the observed simulation output mean and variance, respectively, and let υ denote the uncertainty resulting from the use of a random number stream to generate the simulation results. For OCBA type algorithms, our goal is to maximize what is referred to as the probability of correct selection, or PCS, which is defined as:

$$PCS = P(\tilde{i}^* = i^* \mid \bar{f}_i, \hat{\sigma}_i, i = 1...n)$$

Traditional OCBA-type algorithms solve this problem by providing a static allocation policy derived from a non-linear optimization model that optimizes an approximation of the PCS using the Bonferroni inequality, after allowing the number of replications allocated to each alternative to be continuous. Let L_i be the number of replications allocated to alternative i, let L^*_i be the ratio of any given budget that should be allocated to alternative i, let $\delta_i = f_{i^*} - f_i$, and let L be the total computing budget. The optimality conditions derived in the OCBA literature are as follows:

$$\frac{L^*_{i_1}}{L^*_{i_2}} = \left(\frac{\sigma_{i_1}\delta_{i_2}}{\sigma_{i_2}\delta_{i_1}}\right)^2,\ i_1 \neq i_2 \neq i^* \tag{1}$$

$$L^*_{i^*} = \sigma_{i^*}\sqrt{\Sigma_{i\neq i^*}\frac{(L^*_i)^2}{\sigma^2_i}}$$

$$L = \sum_{i=1}^{n} L_i$$

$$L_i = L * \frac{L^*_i}{\Sigma^n_{i=1} L^*_i} \tag{2}$$

In implementation, since the true mean and variance are unknown, estimated values from observed simulation statistics are used, and the ratios are quickly computed at each iteration of sampling, such that L^*_i is approximated by $L^*_{i,k}$ using the sampling statistics which have been observed up to iteration k, and (2) is computed using L_k, where $L_k = L_{k-1} + \Delta L$. This work considers the case where $\Delta L = 1$. The resulting static policy computed by OCBA often results in large gaps between what has been actually allocated up to iteration k, which we denote as L_{ik}, and what we infer should have been allocated, L_i. To select the actual alternative which will be sampled at a given iteration following the OCBA ratio computations, the most-starving heuristic is often used. Let i' be the alternative which we will sample at a given iteration; then, after computing the OCBA ratios, the most-starving heuristic is:

$$i' = \arg\max_{i=1\ldots n} (L_i - L_{ik})$$

The OCBA algorithm has been extended in many unique ways, such as in Wang et al. (2020), which presents an implementation of OCBA when parallel computing can be effectively leveraged to observe a large number of samples simultaneously. The SAMPLE algorithm described in the introduction, which is based on the Multi-Fidelity Budget Allocation framework (Peng et al. 2018), presents an effective but computationally expensive method of integrating off-line ML estimates into the computation of the OCBA ratios to more effectively allocate replications. The algorithm we present here, EOS, is another extension of OCBA that seeks to improve the convergence rate of OCBA by integrating low-fidelity ML estimates into the budget allocation computations. It leverages the general framework presented by the SAMPLE algorithm for integration of ML estimates with simulation output. The framework has four major components; the simulation itself, the low-fidelity offline estimates, a simulation allocation engine (driven by a simulation optimization algorithm), and a statistical, machine learning, or heuristic model that integrates the offline estimates with the simulation data. EOS provides a computationally efficient method to integrate the low-fidelity estimates that can be updated dynamically as more simulation observations are gathered over time. We denote this as "online learning". In EOS, the online learning aspect is achieved through the use a learning parameter that we compute based on the certainty at which our current observed simulation data will yield accurate estimates of the simulation output mean and variance. To achieve this, we present a concept we call "policy stability", which is used to compute a parameter we denote as ε, to facilitate the direct integration of ML estimates into the selection of alternatives to simulate at each sampling iteration. Policy stability is an estimate of the probability that a certain sampling policy would result in no change in the estimated optimal alternative. In EOS, with probability ε, we sample using traditional OCBA; otherwise, we greedily sample the alternative which is associated with the lowest policy stability. A heuristic argument is presented to suggest that our computation of ε is highly correlated with the relative marginal impact that the OCBA sampling policy would have on PCS; eg, when OCBA sampling would best increase PCS, ε will be large. We show how a value we call the adjusted policy stability can be computed using any number of low-fidelity ML models, and how we can compute and apply a learning parameter β that effectively weights our adjusted policy stability computations in a way that allows the sampling algorithm to appropriately converge to traditional OCBA when we become "confident enough" in the parametric estimates of our simulation output.

2.2 Policy Stability

As previously described, policy stability is a concept used to estimate the impact that a particular sampling policy will have in challenging our current belief about $\tilde{i}^*$. The concept of policy stability is similar to the underlying concept that drives the KG algorithm, except that we treat this computation as a greedy, sub-optimal alternative to what should be the true optimal sampling policy (OCBA). Conceptually, the idea of policy stability and its use in our ε computations are that when many policies (from the set of all possible policies) provide evidence suggesting that the decision we would make with our current statistics may be sub-optimal, we should have a larger propensity to sample in a way that explores the decision space. In this context, “explore" would mean to sample alternatives in a way that best improves the statistical error of our parametric estimates of the simulation output. This is compared to "exploiting" current information, which here means trusting our parametric estimates and sampling to maximize PCS using OCBA.

To apply the idea of policy stability, we first fix our estimates of f_i, σ_i^2 using the observed statistics at a given iteration k. Given an assumption of normality, and denoting by Q the sampling policy which will be used to select alternatives for simulation, we will determine the impact that $l_i(Q)$ replications will have on the probability that $\bar{f}_{i,k+1} > \bar{f}_{\tilde{i}^*,k}$, where $l_i(Q)$ is the number of replications that will be allocated to alternative i under policy Q. Given $\bar{f}_{ik}, \hat{\sigma}^2_{ik}$, subsequent observations $l_i(Q)$ will be normally distributed with mean $\bar{f}_{ik}$ and standard deviation $\hat{\sigma}^2_{ik}$. Upon sampling, the expected value of $\bar{f}_{i,k+1}$ can be estimated as follows:

$$E(\bar{f}_{i,K+1}(l_i(Q))) = \frac{L_{ik}\bar{f}_{ik} + \Sigma_{l=1}^{l_i(Q)} f_l(X_i)}{L_{ik} + l_i(Q)}$$

Furthermore, we can also compute the variance of this estimate as:

$$\text{Var}(\bar{f}_{i,k+1}(Q)) = \frac{l_i(Q)\hat{\sigma}^2_{ik}}{(L_{ik} + l_i(Q))^2}.$$

We can express the probability that $\bar{f}_{i,k+1}(l_i(Q)) > \bar{f}_{\tilde{i}^*,k+1}$ given our current statistical observations by using a Student's t cumulative distribution having mean $M = \bar{f}_{ik} - \bar{f}_{\tilde{i}^*,k}$, and variance

$$S^2 = \frac{l_i(Q)\hat{\sigma}^2_{ik}}{(l_i(Q) + L_{ik})^2} + 1(l_{\tilde{i}^*}(Q) > 0)\frac{l_{\tilde{i}^*}(Q)\hat{\sigma}^2{}_{\tilde{i}^*k}}{(l_{\tilde{i}^*}(Q) + L_{\tilde{i}^*k})^2}$$

with $l_{ik} - 1$ degrees of freedom. The indicator function is required to consider when Q allocates replications to the current estimated optimal alternative, and in the environment considered in this paper where we only sample 1 alternative at each sampling iteration, is not used. We will denote by P_i the probability that, after sampling alternative i $l_i(Q)$ times, the posterior mean $\bar{f}_i$ will be larger than the estimated mean of the current estimated optimal alternative. Then $P_i = \Phi(M/S)$, where $\Phi(\cdot)$ is the cumulative distribution function of a random variable with a Student's t distribution. For $i = \tilde{i}^*$, the probability is computed by considering if sampling $\tilde{i}^*$ would lead to $\bar{f}_{\tilde{i}^*} < \bar{f}_{\tilde{i}^*_2}$, where $\tilde{i}^*_2$ is the current second best alternative, such that $P_{\tilde{i}^*}$ is the probability that after sampling $\tilde{i}^*$, $\tilde{i}^*_2$ becomes $\tilde{i}^*_1$. Under the assumption that $\Delta L = 1$ (and therefore only one alternative will be sampled at a time), this is the only event in the probability space that would result from our beliefs about the true optimal alternative changing.

We denote policy stability as $I(Q)$, and define it as follows:

$$I(Q) = \prod_{i:l_i(Q)>0} (1 - P_i). \tag{3}$$

2.3 Epsilon Computations

Given this approach to computing policy stability, we next consider how to compute ε for EOS. Let L^*_i be the OCBA ratio computed with perfect information (e.g., known mean and variance). Consider that if we knew L^*_i with certainty for all i, then for a given total sampling budget L_k, we would be capable of optimally allocating any additional budget with certainty. Knowing L^*_i with certainty therefore suffices to optimally allocate a given budget. Under OCBA procedures, we assume that the observed statistics about output distribution parameters are correct; therefore, we assume we know L^*_i with certainty, as OCBA computations have been shown to approximate the theoretically optimal allocation (under the aforementioned output assumptions) when variance about parameter estimates is minimized in the asymptotic case. However, for small k and the corresponding L_k replications (where small is relative to the variance of the simulation output and statistical precision required to discern i^* correctly), our estimates for the simulation output distribution parameters will be prone to large error. In the context of such large error, there would be a variance of corresponding magnitude in L^*_{ik}, our estimate of L^*_i. Let $V_k = [L^*_{i,k}(\bar{f}_{ik}, \hat{\sigma}^2_{ik}), i = 1...n]$, where $L^*_{i,k}(\bar{f}_{ik}, \hat{\sigma}^2_{ik})$ is an estimate of L^*_i given the observed statistics $\bar{f}_{ik}$, $\hat{\sigma}^2_{ik}$ at iteration k, and let $V = [L^*_i, i = 1...n]$. If $||[\bar{f}_i, \hat{\sigma}^2_i] - [f_i, \sigma^2_i]|| > 0 \; \forall \; i$, then it immediately follows that $||V_k - V|| > 0$, where $||\cdot||$ denotes any norm,. Observe that these differences are random variables that depend on the random number stream used to generate the simulation results. Also observe that some alternatives are more important to reduce $||V_k - V||$

than others; eg, if we can reduce our variance about the estimates of distribution parameters for the true best alternative to some level of confidence, we would have a larger reduction on $||V_k - V||$ than if we were to reduce the variance about the distribution parameter estimates for the worst possible alternative (since the majority of replications are allocated to the top performing alternatives).

The tension in optimal sampling when considering the variance of the estimates L_{ik}^* is that sampling to best reduce $||V_k - V||$ can be different than sampling to best improve $P_{\bar{f}_i, \hat{\sigma}_i^2, i=1...n}(CS)$, since $L_{\tilde{i}^*}^* \gg L_i^* \ \forall\, i \neq \tilde{i}^*$ by assumption, and $P(\tilde{i}^* = i^*)$ is almost always negligibly small following initialization when estimates are prone to large error. Here, we denote by $P_{\bar{f}_i, \hat{\sigma}_i^2, i=1...n}(CS)$ the *estimated* probability that we have correctly selected the true optimal alternative, which is a random variable subject to not just the magnitude of L_k and how those replications were allocated, but also the variance about $\bar{f}_{ik},\ \hat{\sigma}_{ik}^2$. To optimally reduce the variance about our estimate for i^* with imperfect information, it is therefore not enough to just estimate L_i^* at a given iteration k, we must also consider how and when to sample to reduce the error in our estimates of L_i^*, and balance this with the need to optimally sample with OCBA to best improve the $P_{\bar{f}_i, \hat{\sigma}_i^2, i=1...n}(CS)$. Since, in the general case, $||V_k - V||$ is best reduced by sampling true promising alternatives, and our best guess for which alternatives are promising will be based on the simulation data, it is reasonable to believe that, in the general case, OCBA is a good choice for sampling to reduce parameter estimate error. However, if we apply OCBA routinely, at some point we will have sampled estimated top performers with a large enough frequency such that the reduction in parameter estimate variance is marginal, but the OCBA heuristic would continue to sample these alternatives when L_{ik}^* is large for such promising i. In other words, if we only used OCBA, we would sample "promising" alternatives enough times based on simulation data such that, eventually, the possible reduction in parameter estimate error (and subsequent reduction in error for optimal budget allocation error) will be greater for alternatives which are not currently believed to be "promising".

Denote by Q^* the sampling policy which would theoretically dominate any other policy when maximizing PCS. Let

$$C(Q) = E_{X_i(l_{i(Q)})}\left(\sqrt{\Sigma_{i=1}^m (L_{ik}^*(\bar{f}_i, \hat{\sigma}_i^2) - L_i^*)^2} | \upsilon_{k-1}\right),$$

which represents the expected value of the sum of the square differences between the true optimal sampling policy and the estimated optimal sampling policy, taken with respect to the random observations that would be generated under policy Q, conditioned on the previously observed simulation outputs. Let Q' be the policy that minimizes $C(Q)$. Ideally, one would compute Q^* directly, but this is intractable. Suppose we assume that Q^* is either Q_{OCBA} or Q'; eg, we assume that our parametric estimates are good enough that Q_{OCBA} is truly optimal, or that sampling to best reduce parametric error would result in a greater increase in PCS. If one could both compute the policy Q' which minimizes $C(Q)$, and the marginal increase in $P(CS)$ as a function of selecting either Q_{OCBA} or Q' for sampling, our algorithm would provide the means to compute such values, and use the output to select one of the two policies to sample with at each iteration to maximize $P(CS)$. However, the computation of these values in reasonably large problems is computationally intensive. Instead, we consider the following. As previously described, Q_{OCBA} provides a mechanism for sampling that does result in at least a "good" reduction in error with regards to $C(Q)$ until we have reached a high level of parameter estimation precision for "promising" alternatives relative to the precision of seemingly "poor" alternative parameter estimates. As the precision by which "promising" alternatives are understood increases, then not only does it become more desirable to sample using Q' to maximize $P(CS)$, but it is also clear that the number of alternative sampling policies which would yield a greater reduction in $C(Q)$ would also increase. Therefore, one may conclude that when many alternative sampling policies yield a better opportunity to improve parameter estimates over currently optimal sampling, then we should have a greater propensity to minimize prediction error, as the probability that Q' will result in a larger $P(CS)$ is increasing. Furthermore, observe that for policies which yield some $C(Q)$ that is large relative to some given point, then either the true expected value of that alternative is near-optimal, or the precision by which the expected performance value is known is low, or some combination of the two. By (3), that policy must necessarily have a corresponding small $I(Q)$ relative to other alternatives which yield a smaller $C(Q)$. Our algorithm proposes to use policy stability to compare the relative magnitude of $C(Q)$ for any given set of policies based off of this argument. Without the ability to compute either $C(Q)$ or $\arg\max_{Q_{OCBA},Q'}(P(CS))$ in a computationally efficient way, we argue that one can effectively estimate the likelihood that $P_{Q_{OCBA}}(CS) \geq P_{Q'}(CS)$ by estimating $C(Q)$ using $I(Q)$, and determining the ratio of alternatives for which the statement $I(Q_{OCBA}) > I_Q$ is true. Mathematically, this approximation can be expressed as follows:

$$Q^* = \arg\max_{Q_{OCBA},Q'}(P(CS)) \Rightarrow \tag{4}$$

$$P(Q^* = Q_{OCBA}) \approx \frac{\Sigma_{B\in B} 1[P_{Q_{OCBA}}(CS) > P_Q(CS)]}{|B|} \approx$$

$$\frac{\Sigma_{i=1}^{n} 1(I(Q_i) \geq I(Q_{OCBA}))}{n} = \varepsilon \tag{5}$$

Here, Q_i is the policy that allocates all replications to alternative i, and we can express (5) as given under our assumption that $\Delta L = 1$. In other words, we compare the stability of OCBA with the stability of other alternative sampling policies. When the stability is about the same, or less, for OCBA, as compared to other policies, we sample using Q_{OCBA}. When stability is less for a greater number of policies, we will have a higher propensity for greedily sampling the alternative which we believe would uncover the most information. Since the context of this sampling is that we strictly increment the sample budget by 1 at each iteration, the computations given in (4) are straightforward and easy to carry out for all possible sampling policies. Furthermore, if one has access to low-fidelity ML estimates, we can amend our policy stability estimates to produce an adjusted policy stability estimate, given as follows:

$$I(Q) = \prod_{i:l_i(Q)>0}(1-P_i) + \beta * \sum_{j=1}^{m}\left(\prod_{i:l_i(Q)>0} 1 - U(g_j(X_i) > g_j(X_{\tilde{i}^*}))\right) \tag{6}$$

In the above equation, $g_j(\cdot)$ is the estimate of the j-th ML model, and $U(\cdot)$ is the probability that the performance estimate for alternative i is greater than the estimate for alternative $\tilde{i}^*$ in the j-th ML model, where a normal cumulative distribution is used to compute this probability using a variance estimate based on the mean squared error (MSE) of the j-th ML model. To estimate MSE, we randomly generate a mean for each i from the distribution given by $N(\bar{f}_{ik}, \hat{\sigma}^2_{ik}/L_{ik})$. The parameter β is given by:

$$\beta_k = \exp\left(-\frac{\bar{f}_{\tilde{i}_1^*} - \bar{f}_{\tilde{i}_2^*}}{\hat{\sigma}^2_{\tilde{i}_1^*}/L_{\tilde{i}_1^*,k}}\right) \tag{7}$$

The parameter β weights our offline ML estimates less and less as more and more information is gathered in our high fidelity simulations. For EOS, this discount/learning parameter is how online learning is achieved. Given these computations, we can now present the EOS algorithm in its entirety in the following algorithm.

3 NUMERICAL RESULTS

To provide evidence showing the potential computational efficiency improvement that can be achieved by EOS, we present the results from four numerical experiments comparing the following sampling algorithms: Equal Allocation (EQ), OCBA, SAMPLE, and EOS (both with and without off-line ML estimates incorporated). A set of 250 values were generated at equally spaced positions along the interval [20, 45]. A second set of 250 values were randomly generated from a continuous uniform distribution with parameters [2,3]. These values were consistent across all 4 experimental design points, and represented the true expected performance values and simulation output variances of alternatives $i = 1,2,...250$. To get

Algorithm 1 (EOS algorithm)

INPUT: A set of decision alternatives $\{X_1, X_2, ...X_n\}$, a total simulation budget L, a set number of initial replications l_0, and low-fidelity ML estimates $g_{i,j}, i = 1...n, j = 1...m$

INITIALIZE:

Set $l_i = l_0$, $i = 1, \ldots, n$; set iteration counter $k = 1$; set expended simulation budget $L_k = nl_0$; set total expended budget per alternative $L_{ik} = l_0$. Simulate $X_i, i = 1, \ldots, n$ l_i times, compute sample variance $\hat{\sigma}_i^2$ and sample mean $\bar{f}_i$. Set $\beta = 1$

LOOP: WHILE $L_k < L$ **DO**

1: Compute the MSE of each ML model for all alternatives given current sampling statistics.
2: Compute the OCBA sampling policy according to (1).
3: Compute P_j $\forall$ j according to (3).
4: Compute the adjusted policy stability according to (6).
5: Compute ε according to (4).
6: With probability ε, sample using the computed OCBA policy; else, sample the alternative with the smallest policy stability; update statistics.
7: Update β according to (7).
8: Set $k = k + 1$.

END OF LOOP

Decision: Return the decision with the highest $\bar{f}_i$.

the four design points, two parameters were varied; the number of initial replications allocated, l_0 and the output distribution (Gamma or Normal). To use the Gamma distribution for simulation output, the method of moments were used to fit the distribution parameters. For each of the four experiments, the same set of low-fidelity ML estimates were used. These estimates were generated by randomly perturbing the true performances by a random number observed on a continuous uniform distribution with parameters $[-.2, .2]$. The selection of these parameters to synthetically derive a set of ML estimates was based on the distribution of the simulation expected values and variances use for the experiments and the research objective to show the potential that could be achieved through the integration of ML models. The simulation parameters of the experiments are given in Table 1.

Table 1: Experimental design for comparing ROS efficiency; simulation parameters in vector are [Initial replications l_0, Simulation output distribution (G=Gamma, N=Normal)].

Experiment ID	Simulation Parameters
1	[3,N]
2	[3,G]
2	[10,N]
3	[10,G]

The results of these experiments are given in Figure 1. From these results, we can make several observations. EOS, when applied with or without ML models, performs better than OCBA in all experiments, and performs at a level of efficiency that is the equivalent to SAMPLE. Because EOS largely exploits the normality assumption to the compute the β and ε parameters, it does see a more significant drop in performance when a Gamma distribution is used for the output, as compared to SAMPLE. Furthermore, we can also observe that EOS exhibits preferable behavior when a smaller number of initial replications are used, as opposed to a large number of initial replications. This is a reflection of how the learning parameter β is computed; with a large initialization set, the model will not effectively integrate the information from the ML estimates, and will perform similarly to EOS. A final data point to note which is not depicted in the graphs is the computational difference between the SAMPLE algorithm and the EOS algorithms.

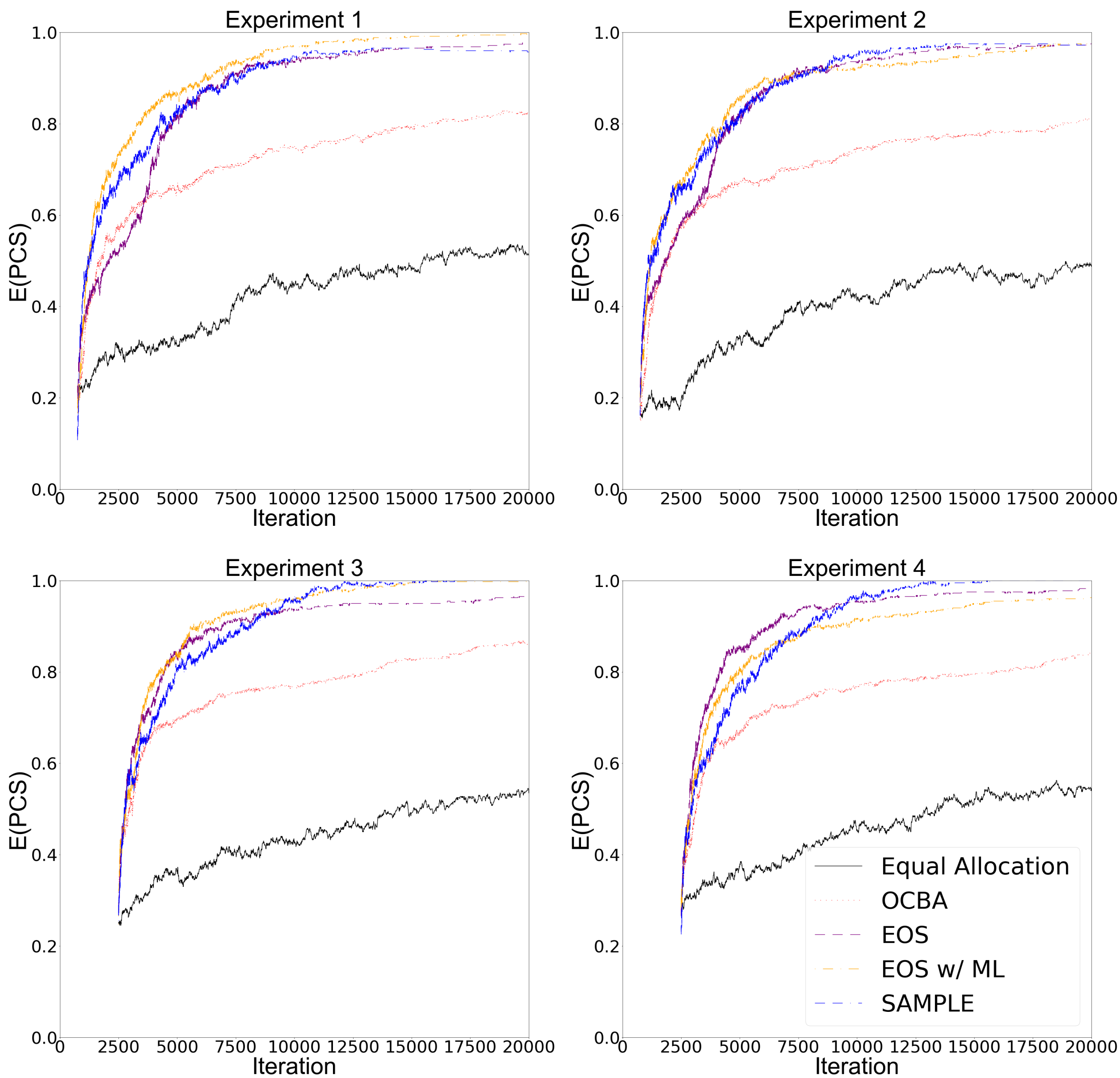

Figure 1: Results from numerical experiments.

The SAMPLE algorithm, even without online updating, requires a substantial amount of computational overhead to initialize, and requires and order of magnitude larger run-time at each iteration to compute the posterior probabilities. EOS and EOS with ML, on the other hand, run on at the same pace as OCBA; therefore, even in the cases where SAMPLE appears to perform the same as EOS with ML or better than EOS, the computational efficiency of both of the EOS algorithms is far less to achieve approximately the same results in terms of PCS.

4 CONCLUSION

In this paper, we presented EOS, a novel algorithm that integrates low-fidelity ML estimates into an OCBA-type simulation optimization algorithm. The concept of policy stability was presented, and the heuristic logic for the computation of an ε and β parameter are given. Since the use of policy-stability to integrate ML estimates and the computation of ε and β were simple, a limited version of online learning could be achieved. Numerical results were presented that showed this algorithm performs better than OCBA, with or without off-line ML models, and could generally perform as well as the SAMPLE algorithm when ML estimates are integrated into the adjusted policy stability computations. One important distinction that makes EOS preferable to SAMPLE given these results is the lack of computational overhead in EOS that SAMPLE requires to execute the EM algorithm and compute the posterior distributions after each iteration. Future research into this framework for simulation optimization should focus on developing a more robust mathematical framework for the heuristic argument presented for the derivation of ε, which would naturally lead to a more effective definition of the parameter. More research should also be done into the derivation and use of the β parameter, to include possible integration of dynamic and computationally efficient ML methods for its derivation and use for online learning. This research might include substantial concepts from the ML technique of boosting using decisions trees. Additionally, the context of this research, simulation optimization as an application of the digital twin concept, presents ample opportunity to address many of the assumptions required to derive EOS. In this research, we do not investigate how one could dynamically update the low-fidelity models used in EOS in a computationally efficient manner either while sampling on-line or between observed contexts. Our assumption that $\Delta L = 1$ is required to derive some of the parameters used in EOS; when $\Delta L \neq 1$, the computations used here become intractable. If this assumption is removed, additional work could be performed to solve the non-linear computations required to derive the associated parameters, and parallel computing could be leveraged to execute more than 1 replication per sampling iteration. Finally, this research does not address or directly consider the impact of low-fidelity ML model accuracy on performance. The fact that EOS performs well without any ML model addition would suggest that the algorithm could rebound quickly from even a disastrously inaccurate ML model, but adjustments in how the learning parameter β is computed might be possible in future work to accommodate for observed error rates in the low-fidelity models. All these lines of effort would serve as substantial research for future work.

ACKNOWLEDGEMENTS

T. Goodwin, J. Xu, N. Celik, and C.-H. Chen were supported in part by the Air Force Office of Scientific Research under grant FA9550-19-1-0383. T. Goodwin and J. Xu were also supported in part by the National Science Foundation under grant 1923145. J. Xu was also supported in part by the National Science Foundation under grant 2228603, and UChicago Argonne LLC under grant 1F-60250. C.-H. Chen was also supported in part by the National Science Foundation under grant FAIN-2123683. Numerical experiments were conducted using resources provided by the Office of Research Computing at George Mason University (URL: https://orc.gmu.edu) and funded in part by grants from the National Science Foundation (Awards Number 1625039 and 2018631).

REFERENCES

Bechhofer, R. 1954. “A Single-Sample Multiple Decision Procedure for Ranking Means of Normal Populations with Known Variances”. *Annals of Mathematical Statistics* 25:16–39.

Calverley, J., C. Currie, B. Onggo, T. Monks, and M. Higgins. 2021. “Simulation Optimisation for Improving the Efficiency of a Production Line”. In *Proceedings of the Operational Research Society Simulation Workshop 2021*, 137–144.

Chen, C.-H., J. Lin, E. Yücesan, and S. E. Chick. 2000. “Simulation Budget Allocation for Further Enhancing the Efficiency of Ordinal Optimization”. *Discrete Event Dynamic Systems* 10(3):251–270.

Chen, E. J. 2011. “A Revisit of Two Stage Selection Procedures”. *European Journal of Operations Research* 210:281–286.

Chen, T., and C. Wang. 2016. “Multi-Objective Simulation Optimization for Medical Capacity Allocation in Emergency Department”. *Journal of Simulation* 10(1):50–68.

Chick, S. E., and K. Inoue. 2001. “New Two-Stage and Sequential Procedures for Selecting the Best Simulated System”. *Operations Research* 49:732–743.

Frazier, P., W. Powell, and S. Daynik. 2008. "A Knowledge-Gradient Policy for Sequential Information Collection". *SIAM Journal on Control and Optimization* 47:2410–2439.

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. Peter, 577–585. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Goodwin, T., J. Xu, N. Celik, and C.-H. Chen. 2022. "Real-Time Digital Twin-based Optimization with Predictive Simulation Learning". *Journal of Simulation* 0(0):1–18.

Hong, L. J., and G. Jiang. 2019. "Offline Simulation Online Application: A New Framework of Simulation-Based Decision Making". *Asia-Pacific Journal of Operational Research* 36(06):1940015.

Hsieh, B.-W., C.-H. Chen, and S.-C. Chang. 2007. "Efficient Simulation-Based Composition of Scheduling Policies by Integrating Ordinal Optimization with Design of Experiment". *IEEE Transactions on Automation Science and Engineering* 4(4):553–568.

Kasaie, P., and W. D. Kelton. 2013. "Simulation Optimization for Allocation of Epidemic-Control Resources". *IIE Transactions on Healthcare Systems Engineering* 3(2):78–93.

Kim, S., 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. Nelson. 2006. "Selecting the Best System". In *Elsevier Handbooks in Operations Research and Management Science: Simulation*, edited by S. H. SG and B. Nelson, Chapter 17. New York: Elsevier.

Nelson, B. L., J. Swann, D. Goldsman, and W. Song. 2001. "Simple Procedures for Selecting the Best Simulated System when the Number of Alternatives is Large". *Operations Research* 49:950–963.

Peng, Y., J. Xu, L. H. Lee, J. Hu, and C.-H. Chen. 2018. "Efficient Simulation Sampling Allocation Using Multifidelity Models". *IEEE Transactions on Automatic Control* 64(8):3156–3169.

Pickardy, C., J. Branke, T. Hildebrandt, J. Heger, and B. Scholz-Reiter. 2010. "Generating Dispatching Rules for Semiconductor Manufacturing to Minimize Weighted Tardiness". In *Proceedings of the Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. R. Montoya-Torres, J. Hugan, and E. Yücesan, 2504–2515. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Qiu, Y., and J. Song. 2016. "A Multiobjective Simulation Optimization of the Macrolevel Patient Flow Distribution". *Healthcare Analytics*:303.

Russo, D. 2020. "Simple Bayesian Algorithms for Best Arm Identification". *Operations Research* 68(6):1625–1647.

Shen, H., L. J. Hong, and X. Zhang. 2021. "Ranking and Selection with Covariates for Personalized Decision Making". *INFORMS Journal on Computing* 33(4):1500–1519.

Song, J., Y. Qiu, J. Xu, and F. Yang. 2019. "Multi-Fidelity Sampling for Efficient Simulation-Based Decision Making in Manufacturing Management". *IISE Transactions* 51(7):792–805.

Thanos, A. E., M. Bastani, N. Celik, and C.-H. Chen. 2015. "Dynamic Data Driven Adaptive Simulation Framework for Automated Control in Microgrids". *IEEE Transactions on Smart Grid* 8(1):209–218.

Wang, J., S. C. Clark, E. Liu, and P. I. Frazier. 2020. "Parallel Bayesian Global Optimization of Expensive Functions". *Operations Research* 68(6):1850–1865.

Xu, J., R. Yao, and F. Qiu. 2020. "Mitigating Cascading Outages in Severe Weather Using Simulation-Based Optimization". *IEEE Transactions on Power Systems* 36(1):204–213.

Yavuz, A., J. Darville, N. Celik, J. Xu, C.-H. Chen, B. Langhals, and R. Engle. 2020. "Advancing Self-Healing Capabilities in Interconnected Microgrids via Dynamic Data Driven Applications System with Relational Database Management". In *Proceedings of the 2020 Winter Simulation Conference*, edited by K.-H. G. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, 2030–2041. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Zhang, F., J. Song, Y. Dai, and J. Xu. 2020. "Semiconductor Wafer Fabrication Production Planning using Multi-Fidelity Simulation Optimisation". *International Journal of Production Research* 58(21):6585–6600.

Zhou, C., J. Xu, E. Miller-Hooks, W. Zhou, C.-H. Chen, L. H. Lee, E. P. Chew, and H. Li. 2021. "Analytics with Digital-Twinning: A Decision Support System for Maintaining a Resilient Port". *Decision Support Systems* 143:113496.

AUTHOR BIOGRAPHIES

TRAVIS GOODWIN is an Operations Research Analyst for The MITRE Corporation. Travis received his bachelors in Mathematics from the University of Maine, Orono, his Masters in Operations Research from George Mason University, and his PhD in Systems Engineering and Operations Research from George Mason University in 2023. Travis' primary work includes implementing computational methods for improving timely access to data for decision makers and portfolio investment trade-offs for large scale, uncertain, and ill-defined decision spaces. His email address is tgoodwin@mitre.org.

JIE XU is an associate professor in the Department of Systems Engineering and Operations Research at George Mason University. He received the M.S. degree in computer science from the State University of New York, Buffalo, in 2004, and the Ph.D. degree in industrial engineering and management sciences from Northwestern University, Evanston, IL, in 2009. His research interests are data analytics, dynamic data driven application systems, stochastic simulation and optimization, with applications in cloud computing, energy systems, electric vehicles, health care, manufacturing, and transportation. His email address is jxu13@gmu.edu.

NURCIN CELIK is an associate professor at the Department of Industrial Engineering at the University of Miami. She received her M.S. and Ph.D. degrees in Systems and Industrial Engineering from the University of Arizona. Her research interests include architectural design and application of dynamic data-driven adaptive simulations for distributed systems. She received the 2017 Presidential Early Career Award for Scientists and Engineers. She received several other awards including the 2020 Provost's Award for Scholarly Activity, ICCS2015 Best Workshop Paper Award, ISERC2015 Best Paper Award, 2014 Eliahu I. Jury Early Career Research Award, and WSC2011 Best Paper Award. Her email address is celik@miami.edu.

CHUN-HUNG CHEN is a professor of Systems Engineering and Operations Research at George Mason University. He has led research projects in stochastic simulation and optimization. He served as Co-Editor of the Proceedings of the 2002 Winter Simulation Conference and Program Co-Chair for 2007 Informs Simulation Society Workshop. He has served on the editorial boards of IEEE Transactions on Automatic Control, IEEE Transactions on Automation Science and Engineering, IIE Transactions, Journal of Simulation Modeling Practice and Theory, and International Journal of Simulation and Process Modeling. He received his Ph.D. degree from Harvard University in 1994. His email address is cchen9@gmu.edu.