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

REPRESENTATIVE CALIBRATION USING BLACK-BOX OPTIMIZATION AND CLUSTERING

Serin Lee
Pariyakorn Maneekul
Zelda B. Zabinsky

Department of Industrial and Systems Engineering
University of Washington
Seattle WA, USA

ABSTRACT

Calibration is a crucial step for model validity, yet its representation is often disregarded. This paper proposes a two-stage approach to calibrate a model that represents target data by identifying multiple diverse parameter sets while remaining computationally efficient. The first stage employs a black-box optimization algorithm to generate near-optimal parameter sets, the second stage clusters the generated parameter sets. Five black-box optimization algorithms, namely, Latin Hypercube Sampling (LHS), Sequential Model-based Algorithm Configuration (SMAC), Optuna, Simulated Annealing (SA), and Genetic Algorithm (GA), are tested and compared using a disease-opinion compartmental model with predicted health outcomes. Results show that LHS and Optuna allow more exploration and capture more variety in possible future health outcomes. SMAC, SA, and GA, are better at finding the best parameter set but their sampling approach generates less diverse model outcomes. This two-stage approach can reduce computation time while producing robust and representative calibration.

1 INTRODUCTION

Calibration, or parameter estimation to fit a model to data, is essential for ensuring the validity of a model and model outcomes (e.g., simulation results). Model calibration typically involves four steps: identify the parameters to be calibrated, select target data to compare with model outcomes, determine a goodness-of-fit (GOF) measure between target data and model outcomes, and choose parameter search strategies (Vanni et al. 2011).

Previous research on model calibration considers two main approaches: i) identifying a single optimal parameter set, and ii) determining a large number of feasible parameter sets. Relying solely on a single parameter set may not account for uncertainty in the target data and may limit the range of future model predictions beyond the calibration period. A large number of parameter sets may be a better representation of uncertainty, but has a high computation cost.

In our previous study (Lee et al. 2021), we developed an agent-based model where we calibrated the model for 152 days and then predicted policy outcomes for 348 days. During the prediction period, we conducted a 4-way sensitivity analysis on policy interventions, resulting in over 300 policy scenarios for each calibration set. Due to the complexity and stochastic nature of the model, each policy run took around 500 minutes. To mitigate the computational resources needed, we employed a clustering approach, reducing the number of parameter sets to two. This enabled us to obtain valuable policy insights while avoiding excessive use of computer resources.

Inspired by our previous study, we propose a two-stage process that we call representative calibration. This approach aims to identify multiple diverse parameter sets that are both computationally efficient and

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

"good enough" to represent the target data. Our calibration approach involves a two-stage process. In the first stage, we leverage well-known black-box optimization algorithms for parameter search and identify good-enough parameter sets. In the second stage, we apply a clustering approach to the selected parameter sets to obtain representative parameter sets. This two-stage approach balances computation efficiency with sufficient coverage of potential future model outcomes.

In this study, we evaluate the effectiveness of our proposed calibration approach and compare the performance of several parameter search algorithms. In Section 2, we review relevant literature on model calibration. Section 3 explains the concept of representation calibration, including black-box optimization algorithms and clustering. We describe the numerical comparison plan in Section 4.1, and present a disease-opinion compartmental model (Lee et al. 2023) in Section 4.2 to illustrate the two-stage calibration process. Finally, we present the study's findings in Section 5 and provide a discussion in Section 6.

2 LITERATURE REVIEW

In recent years, various calibration techniques have been proposed across different fields such as epidemics, economics, engineering, and neuroscience. As models become more complex, data availability increases, and methodology advances, calibration techniques have become more effective and efficient. Most of these techniques focus on time efficiency and precision of the model to target data, with little attention given to the robustness or representativeness of the calibration.

Latin hypercube sampling has been commonly used in calibration due to its simplicity and coverage of a parameter space (Mckay et al. 2000), and is still in use today (Lee et al. 2021; Rao and Brandeau 2022). Black-box optimization methods, such as simulated annealing and genetic algorithms have also been applied to calibration (Cheng et al. 2006; Dahabreh et al. 2017). Bayesian optimization methods, such as the Sequential Model-based Algorithm Configuration (SMAC) (Hutter et al. 2011) and Optuna (Akiba et al. 2019) have been used for hyperparameter tuning and calibration (Kerr et al. 2021; Maurice et al. 2017). Our study explores these well-known calibration algorithms within the proposed two-stage representative calibration framework.

Several Bayesian calibration methods address parameter uncertainty by approximating a posterior distribution. Kerr et al. (2021) used Optuna to calibrate an agent-based epidemiological model. The study derived a posterior distribution of parameters by using more than 15,000 parameter sets. The top ten best-fitting parameter sets were used for scenario analyses. Jalal et al. (2021) combined Bayesian calibration with an artificial neural network as a surrogate model. By generating 10,000 parameter sets, they derived a posterior distribution of parameters while accounting for data uncertainty through the use of threshold values. The paper improved accuracy and computation time compared to an importance sampling algorithm.

A Gaussian process metamodel was used in Xie et al. (2017) to obtain the posterior distribution of parameters, and quantified a credible interval to account for parameter estimation uncertainty. Unlike our approach which aims to capture uncertainty of the model, the calibration performance was focused on reducing the width of credible intervals and prediction intervals. Nevertheless, it remains unclear whether these approaches represent diverse uncertainty in model outcomes, since the top parameter sets were selected solely based on GOF measure.

A clustering approach was adopted in Krauledat et al. (2006) to reduce the calibration process for Brain-Computer Interfaces. The calibration parameters consist of prototypes of a Common Spatial Pattern algorithm used to classify brain states. Although this study uses clustering for calibration, the focus of the study is to reuse previously calibrated clustering from previous data for new data. This is slightly different from our objective in that our main interest lies in efficiently calibrating a model to given data, and then predicting future trajectories, instead of reusing clustered results to new data.

Most of the algorithms focus on time efficiency and precision of the model to target data. The calibrated parameter sets are usually presented as each individual parameter's posterior distribution. Little attention has been given to the information or representativeness of the whole calibration parameter space on the model outcomes.

3 REPRESENTATIVE CALIBRATION

We propose a two-stage process, where the first stage is to optimize the GOF using an optimization algorithm and the second stage is to apply clustering to the results of the first stage optimization, obtaining representative parameter sets.

In the first stage, we use a black-box optimization algorithm to search for parameters and identify parameter sets that minimize the goodness-of-fit measure between model outcomes and target data. The first stage optimization problem is stated as,

$$\begin{aligned} \min_{x} \quad & \text{GOF}(f(x), y) \\ & l_i \leq x_i \leq u_i \qquad \text{for } i = 1, \dots, n \end{aligned} \tag{1}$$

where the target data y is a vector in m dimensions, $y = [y_1, \dots y_m]$, and the model outcome, $f(x)$, may also be m-dimensional to correspond to the target data. The calibration parameter vector $x = [x_1, \dots, x_n]$ denotes a calibration parameter set in n dimensions, and typically has lower and upper limits l_i and u_i, respectively, for $i = 1, \dots, n$. The goodness-of-fit measure is defined by the user to be appropriate to the model and data (e.g., Mean Square Error, Mean Absolute Error, or Total Sum of Squares).

In the second stage, we apply a clustering technique to the parameter sets obtained from the first-stage to identify representative parameter sets. After identifying the representative parameter sets, the model is run for a period longer than the calibration period to observe the diversity of predicted future model trajectories.

3.1 Stage 1: Parameter Search using Black-Box Optimization

This section describes the black-box optimization algorithms used in the first stage of representative calibration. The algorithms include Latin Hypercube Sampling, Sequential Model-based Algorithm Configuration, Optuna, Simulated Annealing, and Genetic Algorithm.

3.1.1 Latin Hypercube Sampling (LHS)

Latin hypercube sampling is a quasi-random sampling method that is often favored in computer experiments because of its simplicity and coverage of the parameter space (Mckay et al. 2000). LHS divides each parameter into equally probable intervals and samples once from each interval. Such even spacing of samples reduces sampling variance and can be applied to high-dimensional problems. However, when the sample size is not large enough, LHS may not be as effective as other algorithms at minimizing GOF.

3.1.2 Sequential Model-based Algorithm Configuration (SMAC)

SMAC is a Bayesian optimization framework that is applied in various areas, ranging from hyperparameter tuning in machine learning to global optimization of black-box functions (Hutter et al. 2011). SMAC creates a random forest surrogate model and updates it as the algorithm proceeds. SMAC combines a local search with random sampling to balance exploration and exploitation (Anastacio and Hoos 2020). Although SMAC's Bayesian approach may find near-optimal solutions with a small number of model runs, it may take a long time to build the random forest, making SMAC appropriate for computationally expensive models.

3.1.3 Optuna

Optuna is an optimization framework actively used in hyperparameter tuning that uses a dynamic approach to explore the search space (Akiba et al. 2019). It employs a combination of sampling and pruning algorithms to improve the efficiency of the optimization algorithm. The default sampling method for Optuna uses the Tree-structured Parzen estimator (TPE), which generates two probability density functions for "good" and "bad" subsets (Bergstra et al. 2011). As default, the median pruning technique is used, which terminates a model if its best performance is inferior to the median of all model outcomes (Golovin et al. 2017). Similar to SMAC, Optuna may require only a small number of model runs, but the additional effort required to build the TPE and median pruning may result in computational overhead.

3.1.4 Simulated Annealing (SA)

Simulated annealing is a metaheuristic global optimization algorithm that randomly samples from a domain and accepts a candidate point based on a "cooling schedule" that gradually decreases over time (Metropolis et al. 2004). This approach helps the algorithm escape local minima and find approximate global optima, which is difficult for other optimization techniques such as gradient descent. However, the performance of simulated annealing depends on the method for generating sequential points and the tuning of the cooling schedule.

3.1.5 Genetic Algorithm (GA)

Genetic algorithms are metaheuristic global optimization algorithms that mimic natural evolution by utilizing the survival of the fittest, selection, and mutation (Holland 1992). GA evaluates solutions based on a fitness function, selects the best ones, and generates new populations using genetic operators like crossover and mutation. GA can dynamically change the search process by varying crossover and mutation probabilities, but it can be computationally expensive for complex problems that require large population sizes and high numbers of generations. Results can also be sensitive to the initial population.

3.2 Stage 2: Clustering

In the second stage, we reduce the number of parameter sets by clustering a set of good-enough parameter sets that satisfy a GOF threshold. Our method for determining the optimal number of clusters, inspired by the elbow method introduced by (Ketchen and Shook 1996), involves identifying the value of K at which an additional cluster (i.e., increasing the number of clusters from K to $K+1$) does not lead to a reduction of more than 5% in the total within-cluster sum of squares (WSS).

Next, the Partition Around Medoids (PAM) algorithm (Kaufman and Rousseeuw 1990) is applied to identify K-medoid points as representative parameter sets. Unlike the centroid method that calculates the mean of all points within a cluster, the medoid is the actual point in the cluster that is most centrally located. It is less susceptible to extreme values that may skew the mean in the centroid approach. Additionally, since the medoid is an actual data point, it has a more straightforward interpretation and can be more easily related to the original data.

4 COMPUTATIONAL STUDY

4.1 Experimental Setup

Our computational study aims to explore a two-stage representative calibration process with five different first-stage optimization algorithms namely, LHS, SMAC, Optuna, SA and GA. We solve the calibration optimization problem (1) with mean absolute error as our goodness-of-fit measure. We run each algorithm for 50,000 function evaluations. For each algorithm, instead of executing 50,000 function evaluations in one run, we run 10 replications of the algorithm with 5,000 function evaluations in each replication. This scheme, from our observation from numerical experiment, balances exploration with exploitation to some extent. For simulated annealing, our numerical experience is that the solution plateaus after about 5,000 function evaluations. So repeating simulated annealing with 10 different starting points (associated with the initial random seed), we hope to identify more good points. Hence, our experiment runs each algorithm for 50,000 function evaluations total, but is the result of 10 repetitions with 5,000 function evaluations each. After the 50,000 function evaluations, we filter the 50,000 points to obtain those with MAE < 2, which we consider good-enough parameter sets.

In the second stage, we employ the Partition Around Medoids algorithm to identify medoid points that represent clusters of the good-enough parameter sets. We then simulate the model for 100 days (59 days for the calibration period and 41 days beyond the calibration period) with the clustered parameter sets and evaluate the result.

All experiments are performed on a high-performance computing cluster equipped with 10 CPU cores and 20GB of memory, using an Intel Xeon processor.

4.2 Model Description

For a numerical study, we calibrate a disease-opinion compartmental model. In this section, we define the parameters to be calibrated, the target data, and the goodness-of-fit measure to evaluate the similarity between target data and model outcomes.

We illustrate our calibration method using a disease-opinion compartmental model on COVID-19, as described in (Lee et al. 2023). The model is a deterministic model that uses ordinary differential equations to represent progression of a disease epidemic and vaccination opinion within a population.

We calibrate thirteen unknown parameters, denoted as $x = (x_1, \ldots, x_{13})$, where each parameter x_i has a lower bound and an upper bound, $l_i \leq x_i \leq u_i$, as presented in Table 1. The calibration parameters consist of four disease-related parameters (x_1, x_3, x_4, x_5) and nine opinion-related parameters $(x_2, x_6, \ldots x_{13})$.

Table 1: 13 calibration parameters (x_i) and their corresponding lower bounds (l_i) and upper bounds (u_i).

Parameter (x_i)	**Description**	**Lower bound** (l_i)	**Upper bound** (u_i)
x_1	Base transmission rate	1.5	2.5
x_2	Relative degree of opinion contact compared to physical contact	0.1	0.5
x_3	Base mortality rate of reference group (ages 18 to 29 years)	0.0002	0.0003
x_4	Average days to lose immunity and become susceptible	150	250
x_5	Proportion of initially susceptible individuals	0.4	0.8
x_6	Proportion of pro-vaccinators that actively share opinion	0.01	0.05
x_7	Emotional judgment importance for vaccination in age group 1	0.01	0.99
x_8	Emotional judgment importance for vaccination in age group 2	0.01	0.99
x_9	Emotional judgment importance for vaccination in age group 3	0.01	0.99
x_{10}	Emotional judgment importance for vaccination in age group 4	0.01	0.99
x_{11}	Emotional judgment importance for vaccination in age group 5	0.01	0.99
x_{12}	Sensitivity of rational judgement to vaccination probability	0.1	1.0
x_{13}	Sensitivity of emotional judgement to vaccination probability	0.1	1.0

Figure 1 displays the target data y which consists of eight weekly data points from January to February 2023 and comprises five types of data. We use the notation $y_{k,t}$ to represent the target data, where k refers to the type of health outcome $k \in \{1, \ldots, 5\}$ and t denotes each week during the period $t \in \{1, \ldots, 8\}$. Specifically, $k = 1, 2,$ and 3 correspond to the percentage of the population that is vaccinated in age group 0 to 17, 18 to 64, and over 65, respectively; $k = 4$ denotes the percentage of the population that is infectious; and $k = 5$ represents the running cumulative number of deaths. Additionally, we use $f_{k,t}(x)$ to represent the five model health outcomes at the eight weekly dates, that correspond to the calibration parameter set x.

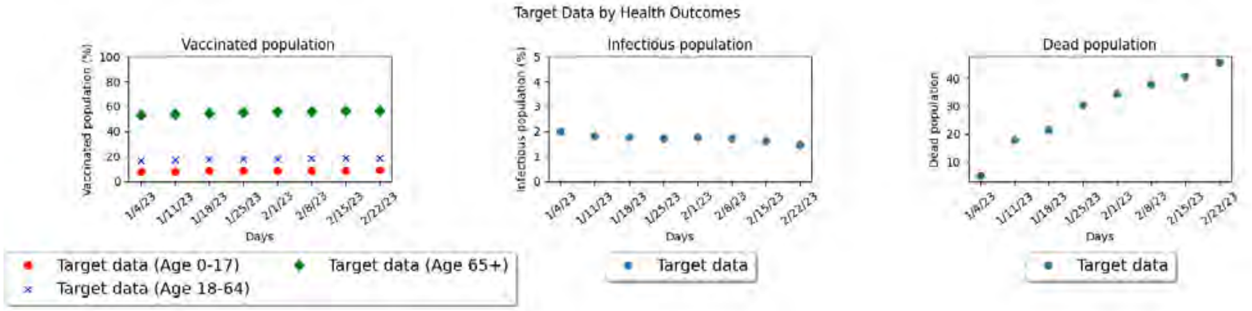


Figure 1: Target data consisting of eight weekly data points on vaccination, infectious, and dead population from January to February 2023.

Our GOF measure between target data $y_{k,t}$ and model outcomes $f_{k,t}(x)$ is calculated as the sum of mean absolute errors (MAE), which is then normalized by the average of each target data. We use MAE instead of mean squared error (MSE) because our normalized measure ranges between 0 to 1, so the absolute difference provides an estimate of the absolute error, whereas MSE tends to underestimate larger differences in the 0 to 1 range. The calibration optimization problem is formulated as follows:

$$\begin{aligned} \min_{x} \quad & \mathrm{GOF}(f(x),y) \\ & \mathrm{GOF}(f(x),y) = \sum_{k=1}^{5}\sum_{t=1}^{8} \frac{|f_{k,t}(x) - y_{k,t}|}{\bar{y}_k} \\ \text{s.t.} \quad & l_i \le x_i \le u_i \qquad \text{for } i = 1,\ldots,13 \end{aligned} \tag{2}$$

where $y_{k,t}, f_{k,t}(x) \in \mathbb{R}$ for $k = 1,\ldots,5$ and $t = 1,\ldots,8$, and $\bar{y}_k = (1/8)\sum_{t=1}^{8} y_{k,t}$ for $k = 1,\ldots,5$.

5 RESULTS

5.1 Parameter Search Results using Black-Box Optimization

This section presents the results from the first stage of representative calibration. As shown in Table 2, SMAC, SA, and GA have a higher number of good enough points (MAE < 2) out of 50,000 points (10 runs with 5,000 function evaluations each). This indicates that SMAC, SA, and GA can identify good solutions quickly and more than two-thirds of the solutions found have MAE < 2. The computation time of SMAC and Optuna is about twice that of LHS, SA, and GA, due to the overhead of computing a random forest and TPE, respectively.

Table 2: Performance measures for each of five algorithms including, the best incumbent MAE value of 50,000 points, number of good-enough points (MAE < 2) out of 50,000 points, and total computation time in seconds.

Algorithm	LHS	SMAC	OPTUNA	SA	GA
Best MAE	1.613	1.350	1.455	1.346	1.378
Number of good enough points (MAE < 2)	318	37,505	7,331	31,562	47,093
Total Time (seconds)	12,157	29,631	13,479	12,296	26,612

Figure 2 shows the incumbent function values averaged over 10 runs of each algorithm, as well as the maximum and minimum value over 10 replications represented by vertical bars at selected numbers of function evaluations. Observe that the result of this plot supports why we did 10 replications of 5,000 function evaluations in each run instead of 50,000 sequential iterations. The plot clearly demonstrates that the algorithms progress in the initial iterations, followed by a slowdown and eventual plateauing, signifying a diminishing return on subsequent iterations. Observe that, in Figure 2, SMAC and SA outperform Optuna and GA in terms of minimizing MAE. Table 2 also shows that the best incumbent MAE value for SMAC and SA is lower than the others.

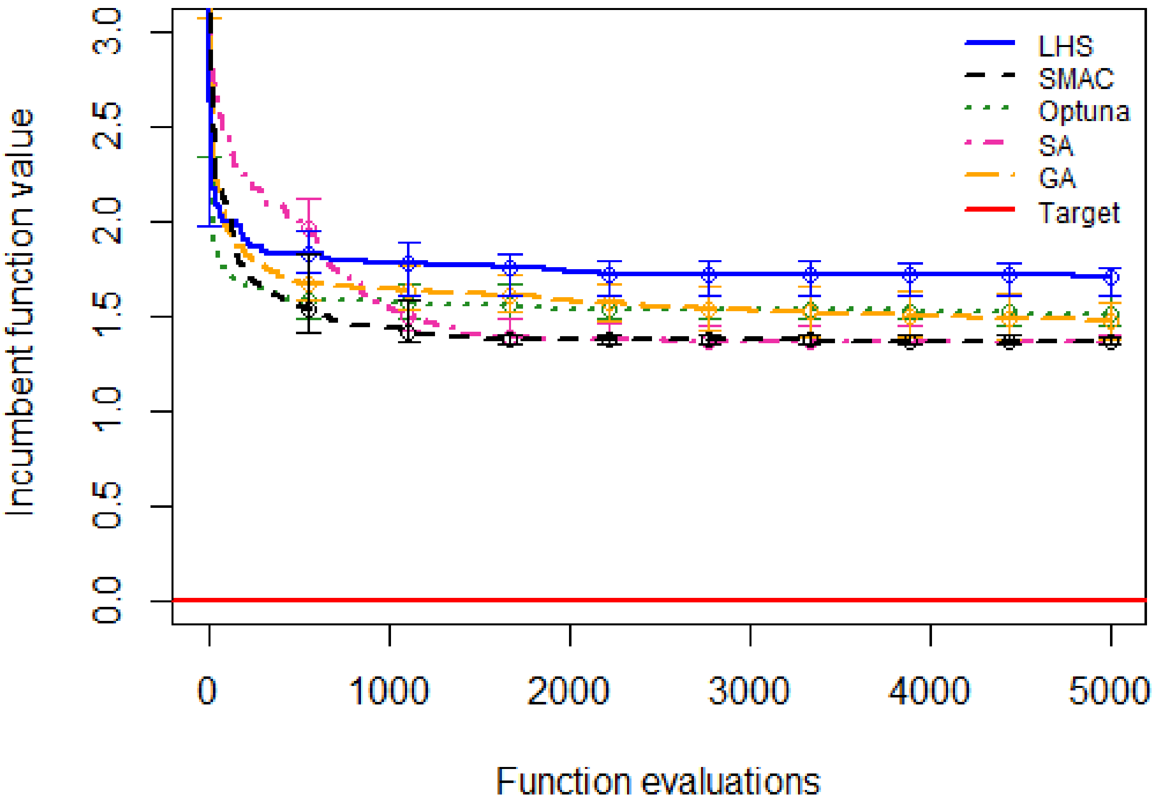
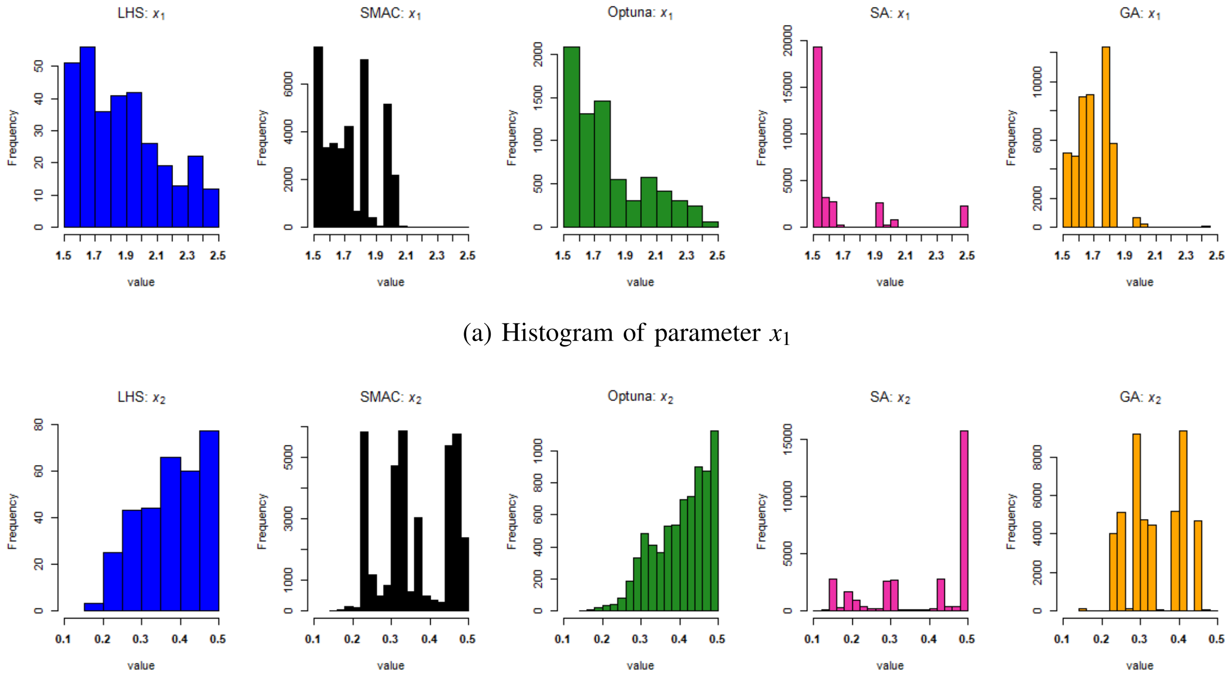


Figure 2: Incumbent function value plots for the average, maximum and minimum MAE over 10 runs of the five algorithms (LHS, Optuna, SMAC, SA, and GA) applied to the disease-opinion compartmental model.

Figure 3 shows histograms of the two most important parameters, x_1 and x_2, from the filtered solutions (MAE < 2) over 50,000 function evaluations from the five algorithms. The distribution of parameter values within their lower and upper bounds differs by algorithm. LHS provides a spread of possible values over the lower and upper bound range, whereas SA and GA are highly concentrated around the near-optimal values. The parameter values from Optuna are nearly as spread out as LHS, but still concentrate on near-optimal values. The distribution is one aspect of representative parameter sets.



(a) Histogram of parameter x_1

(b) Histogram of parameter x_2

Figure 3: Histograms of parameters x_1 and x_2 illustrate the distribution of good enough solutions (MAE < 2) from all the five algorithms (LHS, Optuna, SMAC, SA, and GA) over all 50,000 function evaluations.

5.2 Clustering Results

The Partition Around Medoids algorithm is used to identify K-medoid points as representative parameter sets. For each algorithm, the PAM algorithm is applied to the filtered set of good-enough parameter sets over 50,000 points. We plot the cluster results on the two most important parameters, namely, x_1 and x_2. Figure 4 illustrates the clustering of good enough parameter sets from LHS, SMAC, Optuna, SA, and GA. Each dot in the 2-dimension plot represents a good-enough parameter set, and the large circle represents the medoid point of a cluster. Since the number of clusters K is determined by observing changes in the total within-cluster sum of squares, each algorithm has a different number of clusters, as a result, a different number of medoid points, i.e., LHS has 4 clusters, SMAC has 10 clusters, Optuna has 7 clusters, SA has 12 clusters, and GA has 7 clusters, as shown in Figure 4. In Figure 4, the good-enough points of LHS and Optuna were scattered over both parameter ranges. On the other hand, points from SMAC, SA, and GA focus on a narrower range. Additionally, Optuna was able to provide a higher density of good enough solutions than LHS, with a wide range of representative medoid points. Note that the clustering results from Figure 4 are plotted for two parameters, x_1 and x_2, out of thirteen parameters. The visualization of the nearest medoid in two dimensions does not illustrate the full thirteen dimension space.

5.3 Model Trajectories for 100 Days

Finally, for each black-box optimization algorithm, we run the disease-opinion compartmental model, using the K-medoid parameter sets for 100 days (59 days used for calibration and 41 days for prediction). We then evaluate the diversity of model trajectories by plotting health outcomes for each representative parameter set. Figure 5 presents the five health outcomes from the compartmental model: the vaccinated population by age group (Figure 5a), infectious population (Figure 5b), and dead population (Figure 5c). In each

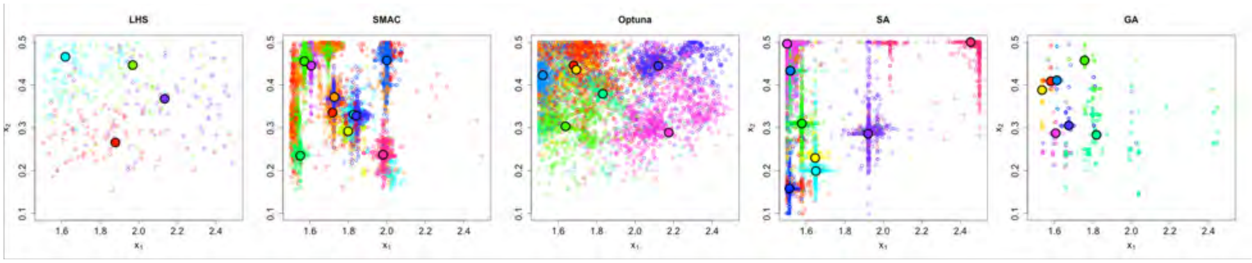


Figure 4: Cluster results on parameters x_1 and x_2 out of 13 parameters, with medoid points for each algorithm.

plot, a single line represents the model trajectory using a medoid point for the parameter set. As such, the number of lines corresponds to the number of clusters from each algorithm. Therefore, LHS, SMAC, Optuna, SA, and GA have 4, 10, 7, 12, and 7 lines, respectively.

LHS (with 4 medoid points) and Optuna (with 7 medoid points) show diverse results in infectious population and dead population from the span of the trajectories. The trajectories for the vaccinated population in the 65+ age group match the target data and are more diverse with SMAC, SA, and GA than for Optuna and LHS. This is helpful in representing the uncertainty of the target data and impact on health outcomes. On the other hand, SMAC, SA, and GA, with multiple concentrated medoid points (10, 12, and 7, respectively) tend to show trajectories that lie close to each other.

6 DISCUSSION

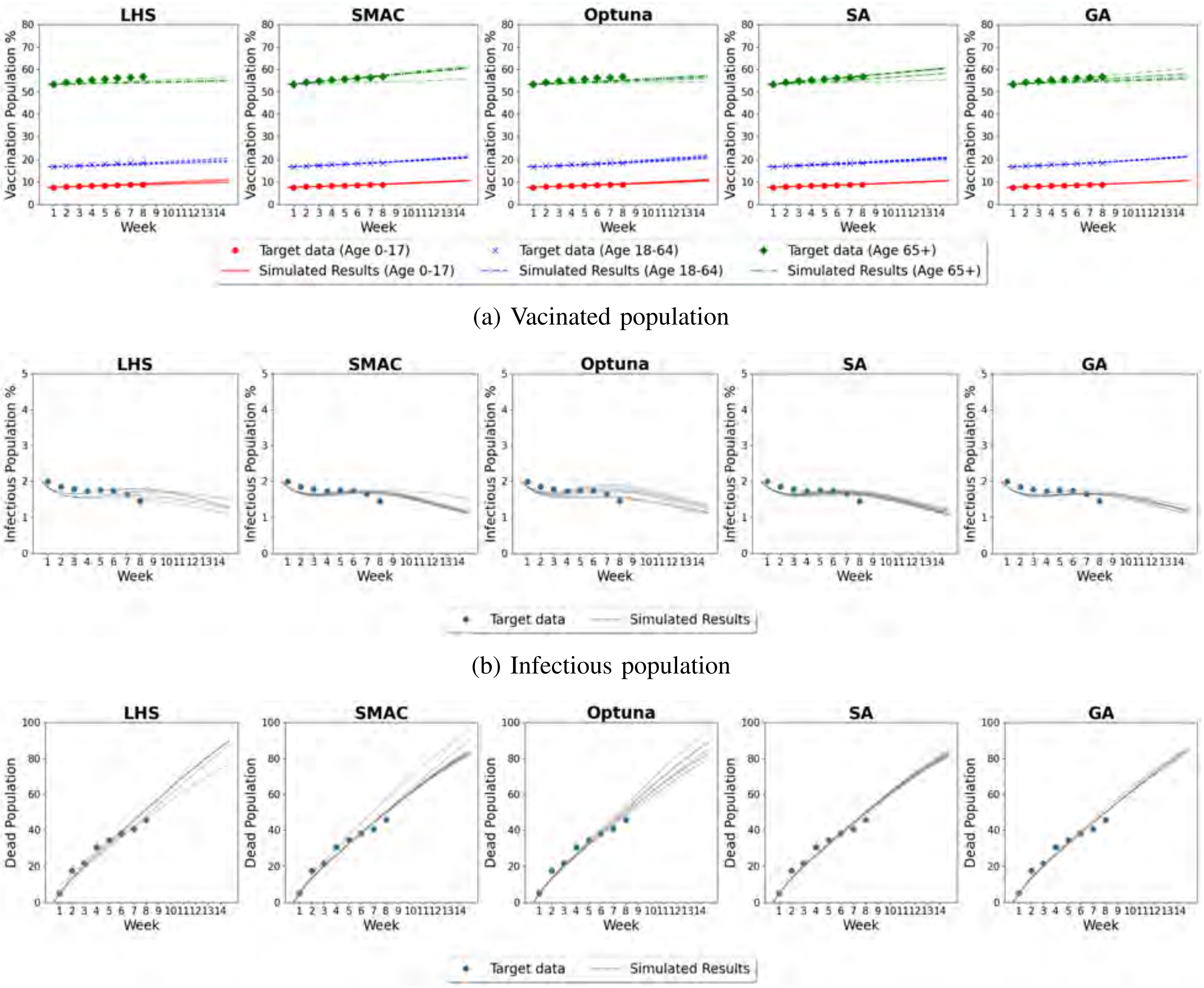
In this paper, we evaluate a representative calibration approach that identifies multiple diverse parameter sets in order to represent the uncertainty of predicted model outcomes while being computationally efficient. The approach involves a two-stage process, (1) apply black-box optimization algorithms to search for good parameter sets, then (2) apply a clustering approach to obtain representative parameter sets.

We observe that in the first stage, most algorithms find good-enough solutions that have relatively low MAE. Considering the best MAE value discovered, SA achieves the lowest MAE, followed by SMAC, GA, Optuna, and LHS, respectively. SMAC converges to a good value faster in the early iterations, as the algorithm uses a random forest surrogate model and a Bayesian approach combining a local search and random sampling. This allows SMAC to find optimal solutions with comparatively few function evaluations. However, SMAC requires the most computation time as shown in Table 2.

In the cluster analysis from stage 2, each black-box optimization algorithm provides slightly different cluster formations. The LHS algorithm, which divides each parameter into equally probable intervals, shows more variety in the solutions. Hence, the solutions from LHS are scattered over the parameter range and are not concentrated on near-optimal areas. On the other hand, SMAC and SA, which perform the best in terms of minimizing MAE, provide multiple clusters of solutions, however, the location of each cluster is in a narrower range of solutions. Optuna also allows more exploration, as evidenced by the histogram and cluster results.

While SMAC and SA achieved the smallest GOF relatively quickly, we do not necessarily consider the algorithms as the best calibration method. Our rationale is that the target data has uncertainty so achieving the sole minimization of GOF may be less crucial for robust modeling. Instead, we aim to identify a broad range of good-enough parameter sets, as this is more important in enhancing model robustness. Thus, we find the results from LHS and Optuna more aligned with our goal, as they offer a wider variety of parameter sets and future trajectories.

We attribute the differences in algorithmic outcomes to the strategies employed for exploration and exploitation. SMAC and SA excel at exploiting and finding optimal solutions, whereas LHS and Optuna excel at exploring and uncovering a wide range of good-enough solutions. We recommend that modelers explore more regions once good-enough solutions are found, which can be achieved through strategies such



(a) Vacinated population

(b) Infectious population

(c) Dead population

Figure 5: Health outcomes from the disease-opinion compartmental model, (a) vaccinated population, (b) infectious polulation and (c) dead population, using the medoid parameter sets for all five algorithms, LHS, Optuna, SMAC, SA, and GA respectively (from left to right).

as having enough different starting points instead of a single long-run, or adjusting algorithmic parameter settings to explore other regions once a threshold is met.

Another consideration for calibration is the choice of GOF measures. When the target data consists of multiple types, such as our model's target data type (i.e, vaccination rates across different age groups, infectious and dead population), the GOF measure may be aggregated to a single value. In our study, we assigned equal weights to each target data type. As shown in Figure 5a, the calibration trajectories exhibit a closer match with the red and blue lines, while relatively less alignment with the green line. Hence, we recommend modelers to carefully select appropriate GOF aggregate metrics based on their specific context and priorities. While we aggregated the GOF scores and selected a threshold value (i.e., MAE < 2), when individual target data have different implications, importance, or reliability, it may be advisable to set different threshold values for each target data type. We will pursue this avenue in the future.

One reason for clustering is to reduce the computation time when performing model analysis. This is especially beneficial in cases where the computation time of the model after the calibration period is longer than the computation time during the calibration period. As an example, in (Lee et al. 2021), the model runtime for the calibration period (152 days) was approximately 500 minutes, whereas the runtime for the prediction period (348 days) took over twice that. Additionally, the model was used to analyze 300 policy scenarios over the prediction period. In this type of situation, it is desirable that a few selected parameter sets are representative to reflect uncertainty in the target data and corresponding model outcomes.

Limitations exist in our model. While we constrained each calibration parameter with box constraints (l_i and u_i), this may limit the parameter space. As the choice of these bounds may impact the model performance, careful consideration is needed in choosing the bounds. While we chose the PAM algorithm to identify K-medoid point, other clustering algorithms should be explored.

In summary, we propose that modelers consider parameter calibration from various perspectives. Instead of solely aiming to minimize the goodness-of-fit quickly, they should 1) recognize that data inaccuracies may exist and allow a certain level of error between the data and model outcomes, 2) formulate the calibration problem with GOF and threshold to identify a range of good-enough parameter sets, 3) explore a broad range of parameter sets so that the model captures the variability of possible future trajectories, and 4) find a sweet spot between exploration and exploitation, wherein exploration promotes broad range of parameter sets while exploitation enhances accuracy in aligning with the target data.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation, grants CMMI-1935403 and CMMI-2204872.

REFERENCES

Akiba, T., S. Sano, T. Yanase, T. Ohta, and M. Koyama. 2019. “Optuna: A Next-Generation Hyperparameter Optimization Framework”. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD ’19, 2623–2631. New York, NY, USA: Association for Computing Machinery.

Anastacio, M., and H. Hoos. 2020. “Model-Based Algorithm Configuration with Default-Guided Probabilistic Sampling”. In *Parallel Problem Solving from Nature – PPSN XVI*, edited by T. Bäck, M. Preuss, A. Deutz, H. Wang, C. Doerr, M. Emmerich, and H. Trautmann, 95–110. Cham: Springer International Publishing.

Bergstra, J., R. Bardenet, Y. Bengio, and B. Kégl. 2011. “Algorithms for Hyper-Parameter Optimization”. In *Advances in Neural Information Processing Systems*, edited by J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, Volume 24. Neural Information Processing Systems: Curran Associates, Inc.

Cheng, C.-T., M.-Y. Zhao, K. Chau, and X.-Y. Wu. 2006. “Using Genetic Algorithm and Topsis for Xinanjiang Model Calibration With a Single Procedure”. *Journal of Hydrology* 316(1):129–140.

Dahabreh, I. J., J. A. Chan, A. Earley, D. Moorthy, E. E. Avendano, T. A. Trikalinos, E. M. Balk, and J. B. Wong. 2017. “Modeling and Simulation in the Context of Health Technology Assessment: Review of Existing Guidance, Future Research Needs, and Validity Assessment”. Technical Report 16(17)-EHC020-EF, Tufts Evidence-based Practice Center, Rockville, MD.

Golovin, D., B. Solnik, S. Moitra, G. Kochanski, J. Karro, and D. Sculley. 2017. “Google Vizier: A Service for Black-Box Optimization”. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’17, 1487–1495. New York, NY, USA: Association for Computing Machinery.

Holland, J. H. 1992. “Genetic Algorithms”. *Scientific American* 267(1):66–73.

Hutter, F., H. H. Hoos, and K. Leyton-Brown. 2011. “Sequential Model-Based Optimization for General Algorithm Configuration”. In *Learning and Intelligent Optimization*, edited by C. A. C. Coello, 507–523. Berlin, Heidelberg: Learning and Intelliigent Optimization (LION): Springer Berlin Heidelberg.

Jalal, H., T. A. Trikalinos, and F. Alarid-Escudero. 2021. “BayCANN: Streamlining Bayesian Calibration With Artificial Neural Network Metamodeling”. *Frontiers in Physiology* 12.

Kaufman, L., and P. J. Rousseeuw. 1990. “Partitioning Around Medoids (Program PAM)”. In *Finding Groups in Data*, Chapter 2, 68–125. John Wiley & Sons, Ltd.

Kerr, C. C., D. Mistry, R. M. Stuart, K. Rosenfeld, G. R. Hart, R. C. Núñez, J. A. Cohen, P. Selvaraj, R. G. Abeysuriya, M. Jastrzębski, L. George, B. Hagedorn, J. Panovska-Griffiths, M. Fagalde, J. Duchin, M. Famulare, and D. J. Klein. 2021. “Controlling COVID-19 via test-trace-quarantine”. *Nature Communications* 12(1):2993.

Ketchen, D. J., and C. L. Shook. 1996. “The Application of Cluster Analysis in Strategic Management Research: An Analysis and Critique”. *Strategic Management Journal* 17(6):441–458.

Krauledat, M., M. Schröder, B. Blankertz, and K.-R. Müller. 2006. "Reducing Calibration Time For Brain-Computer Interfaces: A Clustering Approach". In *Advances in Neural Information Processing Systems*, edited by B. Schölkopf, J. Platt, and T. Hoffman, Volume 19. Neural Information Processing Systems: MIT Press.

Lee, S., S. Liu, and Z. B. Zabinsky. 2023. "Optimal Budget Allocation To Vaccine Promotion Campaigns Considering Disease Transmission And Opinion Propagation". Presented at the INFORMS Healthcare Conference, July 2023.

Lee, S., Z. B. Zabinsky, J. N. Wasserheit, S. M. Kofsky, and S. Liu. 2021. "COVID-19 Pandemic Response Simulation in a Large City: Impact of Nonpharmaceutical Interventions on Reopening Society". *Medical Decision Making* 41(4):419–429.

Maurice, C., F. Madrigal, and F. Lerasle. 2017. "Hyper-Optimization Tools Comparison for Parameter Tuning Applications". In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–6. IEEE.

Mckay, M. D., R. J. Beckman, and W. J. Conover. 2000. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code". *Technometrics* 42(1):55–61.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. 2004, 12. "Equation of State Calculations by Fast Computing Machines". *The Journal of Chemical Physics* 21(6):1087–1092.

Rao, I. J., and M. L. Brandeau. 2022. "Sequential Allocation of Vaccine to Control an Infectious Disease". *Mathematical Biosciences* 351:108879.

Vanni, T., J. Karnon, J. Madan, R. G. White, W. J. Edmunds, A. M. Foss, and R. Legood. 2011. "Calibrating Models in Economic Evaluation". *PharmacoEconomics* 29(1):35–49.

Xie, W., P. Zhang, and Q. Zhang. 2017. "A Stochastic Simulation Calibration Framework for Real-Time System Control". In *Proceedings of the 2017 Winter Simulation Conference*, edited by W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. H. Page, 1914–1925. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

SERIN LEE is a Ph.D. candidate in the Department of Industrial and Systems Engineering at the University of Washington. Her research interest lies in agent-based simulation, mathematical modeling, simulation optimization, and applications to healthcare policy. Her email address is serinlee@uw.edu.

PARIYAKORN MANEEKUL is a Ph.D. candidate in the Department of Industrial and Systems Engineering at the University of Washington. She is also a researcher at Chula Social Innovation at Chulalongkorn University. Her research interest includes optimization under uncertainty and the interplay of statistical learning methods and optimization. Her email address is parim@uw.edu.

ZELDA B. ZABINSKY is a Professor in the Department of Industrial and Systems Engineering at the University of Washington. She is an INFORMS Fellow and an IISE Fellow. Her Ph.D. is from the University of Michigan, in Industrial and Operations Engineering. Her research interests are in global optimization under uncertainty for complex systems, and she has worked in many application areas. Her email address is zelda@uw.edu. Her website is https://ise.washington.edu/facultyfinder/zelda-zabinsky.