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

SURROGATE ASSISTED MODEL REDUCTION FOR STOCHASTIC BIOCHEMICAL REACTION NETWORKS

Prashant Singh
Andreas Hellander

Division of Scientific Computing
Department of Information Technology
Uppsala University
Box 337, SE-751 05
Uppsala, SWEDEN

ABSTRACT

Cellular regulatory mechanisms are typically governed by biochemical reaction networks. Discrete stochastic models are widely used in computational systems biology to analyze such networks. Often, the models involve a large number of highly uncertain parameters and many interacting chemical species. However, one is often interested in observing the output of one, or a few of the species rather than the entire network. Simulating the complete reaction network is inefficient in such cases. This paper explores the use of surrogate models to learn partial stochastic biochemical reaction networks and enable fast near-instant evaluation. The efficacy of the proposed method is demonstrated on a model from the systems biology literature.

1 INTRODUCTION

Biochemical reaction networks involving macromolecules such as DNA, RNA and proteins operate in living cells to regulate cellular function. Due to the small reaction volume, key molecules are only present in very small numbers and discrete stochastic descriptions are preferable over simpler deterministic ordinary differential equations (Vilar et al. 2002, Elowitz et al. 2002, Swain et al. 2002). Such dynamic mechanistic models of biochemical reaction networks model intricate interactions between biochemical species, and are a powerful tool for analysis of cellular behavior.

The most common modeling framework is the continuous-time discrete space Markov processes (CTMC), and statistically correct sample paths, or trajectories can be generated with the stochastic simulation algorithm (Gillespie 1976). By generating large ensembles of trajectories, the statistics of the underlying stochastic model can be analyzed. However, for computational work flows such as parameter exploration, sensitivity analysis (Gunawan et al. 2005, Srivastava et al. 2013) or parameter inference (Golightly and Wilkinson 2011), the computational cost is substantial. This is especially true for large networks with many species and reaction channels, and for stiff systems (Cao et al. 2005).

Although a simulation model of a large biochemical reaction network offers rich detail, there are many scenarios where it might be sufficient to observe only partial outputs of the network. For example, one might be interested in following the evolution of copy numbers of two particular species with respect to changing values of reaction rate constants, that are part of a large reaction network containing tens of species. In such cases, it is wasteful to simulate the entire network repeatedly in order to conduct parameter sweeps. Or if only one species in a large network has been observed experimentally, it would be sufficient to obtain only that species as simulation output in order to conduct parameter inference with e.g. approximate Bayesian computation (Sunnåker et al. 2013). Indeed, for instance there exist reaction networks consisting of 18000 proteins and 44000 interactions (Bader and Madduri 2008). If only a few of those proteins and interactions

are to be studied in detail, simulating the entire network will be very time consuming and inefficient. There are a large number of approximate stochastic simulation methods that offer faster model simulation at the cost of a modeling error, such as tau-leaping (Cao et al. 2007), the Linear Noise Approximation (LNA) (Grima 2015) and moment equations (Engblom 2006). For an overview and comparison of these methods, the reader is referred to (Ferm et al. 2008, Gillespie et al. 2013).

However, none of these methods address the dimension reduction problem of simplifying the chemical network by reducing the number of species and reactions. In fact, to the best of our knowledge, apart from reductions based on time scale separation such as the quasi-steady state assumption (Rao and Arkin 2003), no generally applicable and fully automatic model reduction approach is available to modelers of biochemical networks. This paper proposes a practical and automatic approach to perform model reduction by training a surrogate model that accurately replicates the partial reaction network in question, and offers fast near-instant evaluation. The surrogate can then be used for, e.g., parameter exploration or parameter inference if only partial output of the original model is needed.

Surrogate modeling (Gorissen et al. 2010, Qian et al. 2006) and the related class of surrogate-based optimization methods (Forrester and Keane 2009, Queipo et al. 2005) have emerged as practical alternatives in cases where fast-to-evaluate approximations to simulations are required. A cheap-to-evaluate model is trained by carefully selecting a set of points where simulations are performed. After training a model of sufficient accuracy, the model is used as a *surrogate* in place of the expensive simulator. Since the surrogate model offers near-instant evaluation, it is an ideal choice for use in parameter exploration where repeated evaluations are needed to find the optima.

Surrogate modeling involves training a globally accurate replacement of the complete simulator (or a part of the simulator in case the complexity of the simulator is to be reduced), while surrogate-based optimization involves training a locally accurate model that assists a *sampling algorithm* (Crombecq et al. 2009, Singh et al. 2013) in rapidly sampling towards the optima. Surrogate models are also known as *metamodels* (Queipo et al. 2005), and have been successfully used in a wide variety of applications in the recent years (Singh et al. 2016, Koziel and Leifsson 2016, Huang et al. 2017, Jun-Hee and Kwang-Yong 2016). This work explores model reduction using surrogate models, and does not concern surrogate-based optimization.

The remainder of the paper is organized as follows: Section 2 describes surrogate modeling in detail. Section 3 introduces a novel algorithm for training accurate surrogate models of stochastic simulators. Section 4 evaluates the proposed approach on a well-known biochemical reaction network simulator and the resulting surrogate model is validated by comparing it to the simulator with respect to prediction accuracy and model error. Finally, Section 5 concludes the paper.

2 SURROGATE MODELING

Surrogate models or *metamodels* (Wang and Shan 2007) approximate simulation codes and enable possibilities to either obtain a fast replacement of the simulator, or model a subset of the relationships that the simulator evaluates. This work is focused on the latter use case and aims to obtain a surrogate model that learns a partial biochemical reaction network.

The related category of Model Order Reduction (MOR) (Schilders et al. 2008) methods aims to reduce the complexity of simulation models while preserving the input-output behavior. MOR deals with simplification of dynamical models consisting of a large number of equations and/or variables ($10^5 - 10^9$) (Schilders 2008). Surrogates have also been used to perform MOR across applications (Frangos et al. 2010, Glaz et al. 2010). The problem addressed in this work is a reduction problem, in that a part of the simulator model is to be approximated using a surrogate.

A surrogate model of sufficient accuracy can be built by expending a small number of carefully chosen set of simulations. Thereafter, this surrogate model can enable very fast prediction on test points, and can be used where repeated calls to it are required.

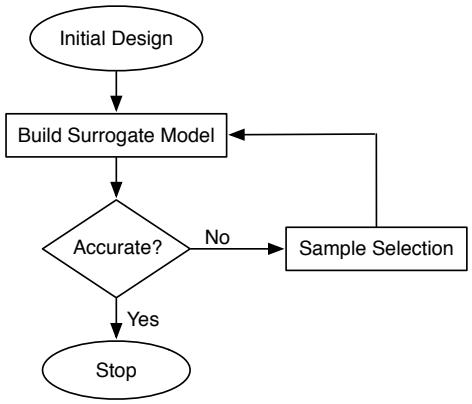

Figure 1: Surrogate modeling flowchart.

Figure 1 depicts the typical surrogate modeling process (Singh 2016). Since the assumption is that nothing is known about the global behavior of the underlying simulator under parameter variations upfront, we begin with a small set of well-chosen locations or points to perform simulations at. This set of points comprising of distinct parameter combinations is known as an initial design X of n points, $X = \{\mathbf{x}_i\}_{i=1}^{n}$. Popular initial designs in literature are factorial designs (Vicente et al. 1998), Latin hypercube sampling (Stein 1987), Box-Behnken (Ferreira et al. 2007), Plackett-Burman (Reddy et al. 2008) and Taguchi designs (Zhang et al. 2007).

The initial design X is evaluated using the simulator f to obtain target values $\mathbf{y}$. The training set $\mathscr{T} = (X, \mathbf{y})$ is used to train a surrogate model $\hat{y}$. Popular surrogate model types in literature include Kriging (Sacks et al. 1989) and the related class of Gaussian process (GP) models (Rasmussen and Williams 2006), Support Vector Machines (SVM) (Cortes and Vapnik 1995), Radial Basis Function (RBF) models (Buhmann 2000), Artificial Neural Networks (ANN) (Haykin 2009), splines (De Boor 1978), rational models (Delbourgo and Gregory 1985), etc.

The surrogate model is then evaluated with respect to specified stopping criteria such as model accuracy, computational budget, etc. If any of the stopping criteria is met, the process halts. If not, a sample selection algorithm selects additional points X' to be evaluated by the simulator resulting in target values $\mathbf{y}'$, and consequently be augmented to the training set,

$$X = X \cup X',$$
$$\mathbf{y} = \mathbf{y} \cup \mathbf{y}'.$$

The surrogate model is re-trained using the updated training set and the stopping criteria are evaluated again. This process continues until one of the stopping criteria is met and a surrogate model of sufficient accuracy is obtained. This work does not involve sampling algorithms, and relies solely on the initial design to arrive at an accurate surrogate model. This is due to the large dimensionality of the parameter space in the target applications. Iterative sampling in high-dimensional ($> 10D$) spaces is computationally expensive to perform (Couckuyt et al. 2014), and the gains to surrogate accuracy will be slow over iterations. For a review of sampling algorithms, the reader is referred to (Forrester and Keane 2009, Forrester et al. 2008).

3 SURROGATE ASSISTED MODEL REDUCTION

Assume N biochemical species $\mathscr{S} = S_1, ..., S_N$ undergoing M reactions $\mathscr{R} = R_1, ..., R_M$ in a given biochemical reaction network with D parameters in the real-valued space $\mathscr{D}$. Let $C_i(t)$ be the number of molecules corresponding to the specie S_i at time t. Let $\mathscr{S}' \subset \mathscr{S}$ be the set of species of interest, involved in the set of reactions $\mathscr{R}' \subset \mathscr{R}$ forming a partial reaction network.

Let $\hat{y}: \mathscr{D} \rightarrow \mathbb{Z}$ be a surrogate model having the mapping from the parameter space $\mathscr{D}$ to the integer-valued space of specie copy numbers $C_i(t), i \in \mathscr{S}'$. The task of the surrogate model is to accurately learn the mapping of the underlying partial biochemical reaction network. The following subsections describe the design of the proposed surrogate-based framework to enable model reduction for stochastic biochemical reaction networks.

3.1 Surrogate Model Type

As listed earlier in Section 1, various surrogate model types such as GP models, SVM, RBF, ANN, etc. are popular in literature (Singh et al. 2016, Forrester and Keane 2009). This work uses GP models due to the following reasons.

1. Bayesian models such as Gaussian processes (GPs) and the related class of Kriging models have proven to perform well across a variety of applications (Forrester and Keane 2009, Kleijnen 2009, Shahriari et al. 2016).
2. GP models are extremely flexible. A correlation function (also known as a kernel) controls the expressive power of the model with regard to the type of target functions that the GP model can fit. There exists a wide variety of such correlation functions, potentially leading to very expressive GP models (Wilson and Adams 2013).
3. GPs provide the uncertainty of prediction that can be utilized by the sampling algorithm (Shahriari et al. 2016, Kleijnen et al. 2012, Couckuyt et al. 2014). Consequently, various sampling algorithms (also known as acquisition functions in the field of Bayesian optimization) that work in tandem with Bayesian models have been proposed and studied. These include improvement-based sampling criteria (Probability of Improvement (PoI) and Expected Improvement (EI)), optimistic criteria (Upper Confidence Bound (UCB) and Lower Confidence Bound (LCB)) and information-based strategies (Thomson Sampling (TS) and Entropy Search (ES)) (Shahriari et al. 2016, Snoek et al. 2012). Although sampling algorithms are not used in this work, the uncertainty of prediction is a very welcome feature to have in a model, and can be used by the practitioner during analysis of the biochemical reaction network. For example, the practitioner may choose to perform additional simulations in regions where the GP surrogate is highly uncertain.

Gaussian process models are very popular as they have a corresponding robust principled probabilistic framework. This enables intuitive understanding and interpretation. The following text describes GP models in detail.

3.2 Gaussian Process Models

Let $y = f(\mathbf{x})$ be a noisy simulator to be approximated using a GP model. A Gaussian process is a generalization of the Gaussian probability distribution (Rasmussen and Williams 2006). In spirit, a GP is an extension of the multivariate Gaussian distribution to an infinite-dimensional stochastic process where any finite combination of dimensions will be jointly-Gaussian (Brochu et al. 2010). A Gaussian process is completely specified by its mean function m, and its covariance function k,

$$f(\mathbf{x}) \sim \mathscr{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \tag{1}$$

For a given test point $\mathbf{x}$, the GP provides the mean and variance of a normal distribution over possible values of f at $\mathbf{x}$.

The covariance function k controls the variation of model prediction from the specified mean function m, and in turn controls the expressive power of the GP. The mean function m encodes prior beliefs about the simulator. In the interest of simplifying calculations, and without loss of generality, setting the mean function $m(\mathbf{x}) = 0$ is a popular choice (Brochu et al. 2010), and convenient.

The choice of the covariance function, or the kernel k is based on the characteristics of the target function. For example, periodic kernels are best suited for applications that entail modeling time series. For non time-varying processes, stationary kernels are more applicable. Different types of covariance functions have been proposed in literature (Wilson and Adams 2013) including the popular squared exponential (SE) function (Snoek et al. 2012) and the Matérn family of functions (Rasmussen and Williams 2006). The Matérn family of kernels consists of different functions with varying levels of sensitivity to dynamic behavior, and are popular in Bayesian and Surrogate-Based optimization. The SE kernel is well-suited for global modeling of smooth functions, and is used in this work. The SE kernel is defined as,

$$k(\mathbf{x}_i, \mathbf{x}_j) = exp\Big(-\frac{1}{2\theta^2}\|\mathbf{x}_i - \mathbf{x}_j\|^2\Big), \tag{2}$$

where θ is a hyperparameter that controls the width of the kernel.

Assume a dataset $\mathscr{T} = (X, \mathbf{y})$ of n points (obtained by evaluating an initial design using the simulator, for example). The kernel matrix $\mathbf{K}$ encoding the covariances between points in X is defined as,

$$\mathbf{K} = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \dots & k(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}_1) & \dots & k(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix}. \tag{3}$$

Let $\mathbf{x}_{n+1}$ be a test point to be predicted as $y_{n+1} = \hat{y}(\mathbf{x}_{n+1})$ using the GP model $\hat{y}$. By definition of GPs, $\mathbf{y}$ and y_{n+1} are jointly-Gaussian,

$$\begin{bmatrix} \mathbf{y} \\ y_{n+1} \end{bmatrix} \sim \mathscr{N}\Big(0, \begin{bmatrix} \mathbf{K} & \mathbf{k} \\ \mathbf{k}^\intercal & k(\mathbf{x}_{n+1}, \mathbf{x}_{n+1}) \end{bmatrix}\Big), \tag{4}$$

where $\mathbf{k} = [k(\mathbf{x}_{n+1}, \mathbf{x}_1,), \dots k(\mathbf{x}_{n+1}, \mathbf{x}_n,)]$. The predictive distribution calculated using the Sherman-Morrison-Woodbury formula (Rasmussen and Williams 2006, Brochu et al. 2010) is then,

$$P(y_{n+1}|\mathscr{T}, \mathbf{x}_{n+1}) = \mathscr{N}(\mu_n(\mathbf{x}_{n+1}), \sigma_n^2(\mathbf{x}_{n+1})), \tag{5}$$

where,

$$\mu_n(\mathbf{x}_{n+1}) = \mathbf{k}^\intercal \mathbf{K}^{-1} \mathbf{y}, \tag{6}$$

$$\sigma_n^2(\mathbf{x}_{n+1}) = k(\mathbf{x}_{n+1}, \mathbf{x}_{n+1}) - \mathbf{k}^\intercal \mathbf{K}^{-1} \mathbf{k}. \tag{7}$$

The calculation of $\mathbf{K}^{-1}$ entails computing the Cholesky decomposition of $\mathbf{K}$ requiring $\mathscr{O}(n^3)$ computational complexity and $\mathscr{O}(n^2)$ storage complexity. The practically confines the applicability of the GP formulation described above to smaller training sets comprising of few thousands of points.

Fortunately, there exist methods that exploit inherent structure in data to speed up computations. Assuming the training set $\mathscr{T}$ consists of n points along a rectilinear grid, it is possible to arrive at the covariance matrix $\mathbf{K}$ with a Kronecker structure. This structure can be exploited to enable GP inference with complexity substantially lesser than $\mathscr{O}(n^3)$ (Wilson et al. 2014).

3.2.1 Fast Scalable Gaussian Process Inference on Rectilinear Grids

Let $\mathscr{T} = (X, \mathbf{y})$ be a training set in D dimensions consisting of n points along a N_p^D Cartesian grid. Let $\mathscr{X}_1, \mathscr{X}_2, ..., \mathscr{X}_D$ be input variables with $\mathbf{x} \in \mathscr{X}_1 \times \mathscr{X}_2 \times ... \times \mathscr{X}_D$. The covariance function or kernel can now be presented as a Kronecker product of D matrices. Consider the product correlation function,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \prod_{p=1}^{D} k(\mathbf{x}_i^{(p)}, \mathbf{x}_j^{(p)}), \tag{8}$$

the correlation matrix $\mathbf{K}$ can be represented by a Kronecker product $\mathbf{K} = \mathbf{K}_1 \bigotimes \mathbf{K}_2 \bigotimes ... \bigotimes \mathbf{K}_D$. Therefore, the calculation of the eigendecomposition of $\mathbf{K}$ as $QVQ^\intercal$ can be performed by separately calculating the eigendecomposition of $\mathbf{K}_1, \mathbf{K}_2, ..., \mathbf{K}_D$. This enables exact inference in $\mathscr{O}(Dn^{\frac{D+1}{D}})$ computational complexity requiring $\mathscr{O}(Dn^{\frac{2}{D}})$ storage (for $D > 1$). A detailed discussion can be found in (Wilson et al. 2014, Wilson and Nickisch 2015).

3.3 Initial Design

The choice of a GP surrogate exploiting grid-structure in training data makes the selection of an initial design straightforward. A full-factorial design forming a $6D$ rectilinear grid with 5 points per dimension is selected. This results in a training set of $n = 5^6 = 15625$ points. It is possible to have a denser grid with more points per dimension, but it will lead to very lengthy simulation times in order to evaluate the training set since each simulation takes 2 seconds for the test problem considered in this work.

4 EXPERIMENTS: A GENETIC OSCILLATOR

As an example we use a biochemical reaction network model of a genetic oscillator from the systems biology literature (Vilar et al. 2002). The model consists of 18 reactions involving 9 species parameterized by 15 reaction constants or variables. Let $\mathscr{S} = \{D_A, D_A^*, M_A, D_R, D_R^*, M_R, C, A, R\}$ be the set of species with initial copy numbers $\{1,0,0,1,0,0,10,10,10\}$ respectively. The model involves the following reactions,

$$sD_A : D_A^* \xrightarrow{\theta_A} D_A, \tag{9}$$

$$sD_A^* : D_A, A \xrightarrow{\gamma_A} D_A^*, \tag{10}$$

$$sD_R : D_R^* \xrightarrow{\theta_R} D_R, \tag{11}$$

$$sD_R^* : D_R, A \xrightarrow{\gamma_R} D_R^*, \tag{12}$$

$$sMA_1 : D_A^* \xrightarrow{\alpha_A^*} D_A^*, M_A, \tag{13}$$

$$sMA_2 : D_A \xrightarrow{\alpha_A} D_A, M_A, \tag{14}$$

$$aM_A : M_A \xrightarrow{\delta_{MA}} \phi, \tag{15}$$

$$sA_1 : M_A \xrightarrow{\beta_A} A, M_A, \tag{16}$$

$$sA_2 : D_A^* \xrightarrow{\theta_A} D_A^*, A, \tag{17}$$

$$sA_3 : D_R^* \xrightarrow{\theta_A} D_R^*, A, \tag{18}$$

$$aA : A \xrightarrow{\delta_A} \phi, \tag{19}$$

$$sC : A, R \xrightarrow{\gamma_C} C, \tag{20}$$

$$sMR_1 : D_R^* \xrightarrow{\alpha_{R^*}} D_R^*, M_R, \tag{21}$$

$$sMR_2 : D_R \xrightarrow{\alpha_R} D_R, M_R, \tag{22}$$

$$aMR : M_R \xrightarrow{\delta_{MR}} \phi, \tag{23}$$

$$sR_1 : M_R \xrightarrow{\beta_R} M_R, R, \tag{24}$$

$$aR : R \xrightarrow{\delta_R} \phi, \tag{25}$$

$$sR_2 : C \xrightarrow{\delta_A} R. \tag{26}$$

For the purpose of experiments, surrogate modeling is used to track the expected value of copy numbers of species C and A over 1000 time steps. This involves training two GP surrogates, one each for species C and A. Since the parameter space of 15 reaction constants is very large and time consuming to analyze, the first 9 parameters are fixed as $\{\alpha_A, \alpha_A^*, \alpha_R, \alpha_R^*, \beta_A, \beta_R, \delta_{MA}, \delta_{MR}, \delta_A\} = \{50, 500, 0.01, 50, 50, 5, 10, 0.5, 1\}$, and signify a region of interest. The remaining 6 parameters are varied in the following range to study the effect of parameter variations on the copy numbers of species in the region of interest,

$$\delta_R \in [0, 0.5], \tag{27}$$

$$\gamma_A \in [0.5, 1.5], \tag{28}$$

$$\gamma_R \in [0.5, 1.5], \tag{29}$$

$$\gamma_C \in [1, 3], \tag{30}$$

$$\theta_a \in [30, 70], \tag{31}$$

$$\theta_r \in [80, 120]. \tag{32}$$

The model is implemented in StochSS (Drawert et al. 2016) and evaluated in Python using GillesPy (Abel et al. 2016). In order to train the grid-based GP surrogates, simulations of the model are performed over a rectilinear grid consisting of 5 points per dimension or parameter, resulting in a training set $\mathscr{T}$ of $n = 5^6 = 15625$ points.

GP models are trained using the GPatt framework (Wilson et al. 2014) implemented as part of the GPML toolbox (Rasmussen and Nickisch 2010) for MATLAB (MATLAB 2015). The GP models are validated on a uniformly generated test set $\mathscr{T}^*$ consisting of 500 points that are distinct from the training set $\mathscr{T}$. The test metrics chosen to evaluate the accuracy of the model are Bayesian Estimation Error Quotient (BEEQ) (Li and Zhao) and Root Relative Squared Error (RRSE), and are briefly described below.

1. BEEQ measures the improvement of error of a Bayesian model $\hat{y}$, over the prior mean $\bar{y}$. BEEQ is computed for n test points as,

$$BEEQ(\hat{y}) = \Big(\prod_{i=1}^{n} \beta_i\Big)^{1/n} \tag{33}$$

   where,

$$\beta_i = \frac{\|y_i - \hat{y}(\mathbf{x}_i)\|}{\|y_i - \bar{y}(\mathbf{x}_i)\|}. \tag{34}$$

   The ideal value of BEEQ is 0. BEEQ is resilient to the effect of very large or small magnitudes of values on the error estimate.
2. RRSE is calculated as,

$$RRSE(\mathbf{y}, \hat{\mathbf{y}}) = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{35}$$

   where y, $\hat{\mathbf{y}}$ and $\bar{y}$ are the true, predicted and mean true response values respectively. RRSE measures the goodness of the approximation relative to the mean approximation. A lower value of RRSE indicates a better approximation.

Table 1 lists the error metrics computed for the trained GP models corresponding to both species. The BEEQ and RRSE estimates are close to 0, which points towards stable and accurate models that have been able to capture the general trend of the simulated values. Considering the rather large $6D$ parameter space, and comparatively few points (5) for training per dimension, this is an encouraging result. In general, it is difficult to define an generally acceptable value of error estimates since the values depend heavily on

Table 1: GP model error estimates computed on a test dataset of 500 points distinct from the training set.

Specie	BEEQ	RRSE
A	0.990045	0.959021
C	0.982493	0.958051

the nature of the underlying simulator. Certain reaction networks are easier to model than others, and may result in more accurate models. Complex reaction networks such as the one described above, are more difficult for the model to learn. This distinction will reflect in the error estimates as well. Therefore, in absence of a generally acceptable stopping criterion, or desired model accuracy, it is recommended to train the surrogate using as many points as practically possible.

Table 2: GP model training and evaluation times in comparison to the simulator. The training set consists of 15625 points. The simulator and GP surrogate are evaluated on 500 test points.

time (s)	GP surrogate	simulator
training	24.208079	-
evaluation	**0.384075**	1325.812937

Table 2 shows a comparison of time taken to evaluate 500 test points by the GP model and the simulator. It also lists the training time involved for the GP model. It can be observed that the GP surrogate is more than 3 orders of magnitude faster than the simulation model, and offers almost-instant evaluation while achieving reasonable accuracy. The advantage of evaluation time is illustrated by 8 hours of computational time taken to simulate 15625 points in the training set. In contrast, the GP surrogate is able to evaluate 15625 points in mere 12 seconds. This enables sophisticated analysis on the behavior and variation of parameters using the surrogate. The training time for the GP surrogate is also reasonable at 24 seconds considering the complexity of the problem. Future work involves scaling up the model to above $100,000$ points, and taking more dimensions into account while increasing the accuracy of the GP surrogate. The approach will also be validated on different biochemical reaction networks.

5 CONCLUSION

A novel surrogate-assisted approach for model reduction of complex large-scale stochastic biochemical reaction network simulators is presented for applications that can benefit from analysis of partial reaction networks. An accurate surrogate model is trained over the domain of the partial reaction network. Thereafter, the surrogate model enables fast efficient inference and analysis of the partial reaction network and alleviates the computational overhead of simulating the entire reaction network. The proposed approach is validated on a well-known stochastic biochemical reaction network simulator, and resulted in an accurate surrogate model that is multiple orders of magnitude faster to evaluate than the simulator.

ACKNOWLEDGMENTS

This work was funded by the Göran Gustafsson foundation and the eSSENCE strategic collaboration on eScience.

REFERENCES

Abel, J. H., B. Drawert, A. Hellander, and L. R. Petzold. 2016. "GillesPy: A Python Package for Stochastic Model Building and Simulation". *IEEE Life Sciences Letters* 2 (3): 35–38.

Bader, D. A., and K. Madduri. 2008. "A Graph-Theoretic Analysis of the Human Protein-Interaction Network Using Multicore Parallel Algorithms". *Parallel Computing* 34 (11): 627–639.

Brochu, E., V. M. Cora, and N. De Freitas. 2010. "A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning". *arXiv preprint arXiv:1012.2599*.

Buhmann, M. D. 2000. "Radial Basis Functions". *Acta Numerica 2000* 9:1–38.

Cao, Y., D. Gillespie, and L. Petzold. 2005. "The Slow-Scale Stochastic Simulation Algorithm". *The Journal of Chemical Physics* 122 (1): 014116.

Cao, Y., D. Gillespie, and L. Petzold. 2007. "Adaptive Explicit-Implicit Tau-Leaping Method with Automatic Tau Selection". *The Journal of Chemical Physics* 126 (22): 224101.

Cortes, C., and V. Vapnik. 1995. "Support-Vector Networks". *Machine learning* 20 (3): 273–297.

Couckuyt, I., D. Deschrijver, and T. Dhaene. 2014. "Fast Calculation of Multiobjective Probability of Improvement and Expected Improvement Criteria for Pareto Optimization". *Journal of Global Optimization* 60 (3): 575–594.

Crombecq, K., L. D. Tommasi, D. Gorissen, and T. Dhaene. 2009. "A Novel Sequential Design Strategy for Global Surrogate Modeling". In *Proceedings of the 2009 Winter Simulation Conference*, edited by M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, 731–742. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

De Boor, C. 1978. *A Practical Guide to Splines*, Volume 27. Springer-Verlag New York.

Delbourgo, R., and J. A. Gregory. 1985. "Shape Preserving Piecewise Rational Interpolation". *SIAM Journal on Scientific and Statistical Computing* 6 (4): 967–976.

Drawert, B., A. Hellander, B. Bales, D. Banerjee, G. Bellesia, B. J. Daigle, Jr., G. Douglas, M. Gu, A. Gupta, S. Hellander, C. Horuk, D. Nath, A. Takkar, S. Wu, P. Ltstedt, C. Krintz, and L. R. Petzold. 2016, 12. "Stochastic Simulation Service: Bridging the Gap between the Computational Expert and the Biologist". *PLOS Computational Biology* 12 (12): 1–15.

Elowitz, M., A. Levine, E. Siggia, and P. Swain. 2002. "Stochastic Gene Expression in a Single Cell". *Nature* 297 (5584): 1183–1186.

Engblom, S. 2006. "Computing the Moments of High Dimensional Solutions of the Master Equation". *Applied Mathematics and Computation* 180 (2): 498–515.

Ferm, L., P. Lötstedt, and A. Hellander. 2008. "A Hierarchy of Approximations of the Master Equation Scaled by a Size Parameter". *Journal of Scientific Computing* 34 (2): 127–151.

Ferreira, S. C., R. Bruns, H. Ferreira, G. Matos, J. David, G. Brandao, E. P. da Silva, L. Portugal, P. Dos Reis, A. Souza et al. 2007. "Box-Behnken Design: An Alternative for the Optimization of Analytical Methods". *Analytica Chimica Acta* 597 (2): 179–186.

Forrester, A., A. Sobester, and A. Keane. 2008. *Engineering Design Via Surrogate Modelling: A Practical Guide*. John Wiley & Sons.

Forrester, A. I., and A. J. Keane. 2009. "Recent Advances in Surrogate-Based Optimization". *Progress in Aerospace Sciences* 45 (1): 50–79.

Frangos, M., Y. Marzouk, K. Willcox, and B. van Bloemen Waanders. 2010. "Surrogate and Reduced-Order Modeling: A Comparison of Approaches for Large-Scale Statistical Inverse Problems [Chapter 7]".

Gillespie, D. T. 1976. "A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reacting Systems". *Journal of Computational Physics* 22 (4): 403–434.

Gillespie, D. T., A. Hellander, and L. R. Petzold. 2013. "Perspective: Stochastic Algorithms for Chemical Kinetics". *The Journal of Chemical Physics* 138 (17): 170901.

Glaz, B., L. Liu, and P. P. Friedmann. 2010. "Reduced-Order Nonlinear Unsteady Aerodynamic Modeling Using a Surrogate-Based Recurrence Framework". *AIAA journal* 48 (10): 2418–2429.

Golightly, A., and D. J. Wilkinson. 2011. "Bayesian Parameter Inference for Stochastic Biochemical Network Models Using Particle Markov Chain Monte Carlo". *Interface Focus* 1 (6): 807–820.

Gorissen, D., I. Couckuyt, P. Demeester, T. Dhaene, and K. Crombecq. 2010. "A Surrogate Modeling and Adaptive Sampling Toolbox for Computer Based Design". *Journal of Machine Learning Research* 11:2051–2055.

Grima, R. 2015. "Linear-Noise Approximation and the Chemical Master Equation Agree up to Second-Order Moments for a Class of Chemical Systems". *Phys. Rev. E* 92:042124.

Gunawan, R., Y. Cao, L. Petzold, and F. J. Doyle. 2005. "Sensitivity Analysis of Discrete Stochastic Systems". *Biophysical Journal* 88.

Haykin, S. S. 2009. *Neural Networks and Learning Machines*, Volume 3. Pearson Upper Saddle River, NJ, USA:.

Huang, B., T. Chen, G. Wang, D. Gao, J. Zhao et al. 2017. "Numerical Simulations and Surrogate-Based Optimization of Cavitation Performance for an Aviation Fuel Pump". *Journal of Mechanical Science and Technology* 31 (2): 705–716.

Jun-Hee, K., and K. Kwang-Yong. 2016. "Surrogate-Based Optimization of a Cratered Cylindrical Hole to Enhance Film-Cooling Effectiveness". *Journal of Thermal Science and Technology* 11 (2): JTST0025–JTST0025.

Kleijnen, J. P. 2009. "Kriging Metamodeling in Simulation: A Review". *European Journal of Operational Research* 192 (3): 707–716.

Kleijnen, J. P., W. Beers, and I. Nieuwenhuyse. 2012. "Expected Improvement in Efficient Global Optimization Through Bootstrapped Kriging". *Journal of Global Optimization* 54 (1): 59–73.

Koziel, S., and L. Leifsson. 2016. *Surrogate-Based Optimization Using Parametric Response Correction*, 75–98. Springer.

Li, X. R., and Z. Zhao. "Evaluation of Estimation Algorithms Part I: Incomprehensive Measures of Performance". *IEEE Transactions on Aerospace and Electronic Systems* 42 (4): 1340–1358.

MATLAB 2015. *Version 8.6.0 (R2015b)*. Natick, Massachusetts: The MathWorks Inc.

Qian, Z., C. C. Seepersad, V. R. Joseph, J. K. Allen, and C. J. Wu. 2006. "Building Surrogate Models Based on Detailed and Approximate Simulations". *Journal of Mechanical Design* 128 (4): 668–677.

Queipo, N. V., R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. K. Tucker. 2005. "Surrogate-Based Analysis and Optimization". *Progress in Aerospace Sciences* 41 (1): 1–28.

Rao, C. V., and A. P. Arkin. 2003. "Stochastic Chemical Kinetics and the Quasi-Steady-State Assumption: Application to the Gillespie Algorithm". *The Journal of Chemical Physics* 118 (11): 4999–5010.

Rasmussen, C. E., and H. Nickisch. 2010. "Gaussian Processes for Machine Learning (GPML) Toolbox". *Journal of Machine Learning Research* 11 (Nov): 3011–3015.

Rasmussen, C. E., and C. K. Williams. 2006. "Gaussian Processes for Machine Learning".

Reddy, L., Y.-J. Wee, J.-S. Yun, and H.-W. Ryu. 2008. "Optimization of Alkaline Protease Production by Batch Culture of Bacillus Sp. RKY3 Through Plackett–Burman and Response Surface Methodological Approaches". *Bioresource Technology* 99 (7): 2242–2249.

Sacks, J., W. J. Welch, T. J. Mitchell, and H. P. Wynn. 1989. "Design and Analysis of Computer Experiments". *Statistical Science* 4 (4): 409–423.

Schilders, W. 2008. "Introduction to Model Order Reduction". In *Model Order Reduction: Theory, Research Aspects and Applications*, 3–32. Springer.

Schilders, W. H., H. A. Van der Vorst, and J. Rommes. 2008. *Model Order Reduction: Theory, Research Aspects and Applications*, Volume 13. Springer.

Shahriari, B., K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. 2016. "Taking the Human out of the Loop: A Review of Bayesian Optimization". *Proceedings of the IEEE* 104 (1): 148–175.

Singh, P. 2016. *Design of Experiments for Model-Based Optimization*. Ph. D. thesis, Ghent University.

Singh, P., I. Couckuyt, K. Elsayed, D. Deschrijver, and T. Dhaene. 2016. "Shape Optimization of a Cyclone Separator Using Multi-Objective Surrogate-Based Optimization". *Applied Mathematical Modelling* 40 (5): 4248–4259.

Singh, P., D. Deschrijver, and T. Dhaene. 2013. "A Balanced Sequential Design Strategy for Global Surrogate Modeling". In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 2172–2179. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Snoek, J., H. Larochelle, and R. P. Adams. 2012. "Practical Bayesian Optimization of Machine Learning Algorithms". In *Advances in Neural Information Processing Systems*, 2951–2959.

Srivastava, R., D. F. Anderson, and J. B. Rawlings. 2013. "Comparison of Finite Difference Based Methods to Obtain Sensitivities of Stochastic Chemical Kinetic Models". *The Journal of Chemical Physics* 138 (7): 074110.

Stein, M. 1987. "Large Sample Properties of Simulations Using Latin Hypercube Sampling". *Technometrics* 29 (2): 143–151.

Sunnåker, M., A. G. Busetto, E. Numminen, J. Corander, M. Foll, and C. Dessimoz. 2013. "Approximate Bayesian Computation". *PLOS Computational Biology* 9 (1): 1–10.

Swain, P., M. Elowitz, and E. Siggia. 2002. "Intrinsic and Extrinsic Contributions to Stochasticity in Gene Expression". *Science* 99 (20): 12795–12800.

Vicente, G., A. Coteron, M. Martinez, and J. Aracil. 1998. "Application of the Factorial Design of Experiments and Response Surface Methodology to Optimize Biodiesel Production". *Industrial Crops and Products* 8 (1): 29–35.

Vilar, J. M., H. Y. Kueh, N. Barkai, and S. Leibler. 2002. "Mechanisms of Noise-Resistance in Genetic Oscillators". *Proceedings of the National Academy of Sciences* 99 (9): 5988–5992.

Wang, G. G., and S. Shan. 2007. "Review of Metamodeling Techniques in Support of Engineering Design Optimization". *Journal of Mechanical design* 129 (4): 370–380.

Wilson, A., E. Gilboa, J. P. Cunningham, and A. Nehorai. 2014. "Fast Kernel Learning for Multidimensional Pattern Extrapolation". In *Advances in Neural Information Processing Systems*, 3626–3634.

Wilson, A., and H. Nickisch. 2015. "Kernel Interpolation for Scalable Structured Gaussian Processes (KISS-GP)". In *International Conference on Machine Learning*, 1775–1784.

Wilson, A. G., and R. P. Adams. 2013. "Gaussian Process Kernels for Pattern Discovery and Extrapolation.". In *International Conference on Machine Learning*, 1067–1075.

Zhang, J. Z., J. C. Chen, and E. D. Kirby. 2007. "Surface Roughness Optimization in an End-Milling Operation Using the Taguchi Design Method". *Journal of Materials Processing Technology* 184 (1): 233–239.

AUTHOR BIOGRAPHIES

PRASHANT SINGH is a post-doctoral researcher in the Division of Scientific Computing, Department of Information Technology at Uppsala University. His research interests include model-based optimization, surrogate modeling, applied machine learning and scientific computing. His e-mail address is prashant.singh@it.uu.se.

ANDREAS HELLANDER is an Associate Professor of Scientific Computing. His research interests are in the fields of systems biology, distributed computing and stochastic simulations. His e-mail address is andreas.hellander@it.uu.se.