Proceedings of the 2014 Winter Simulation Conference
A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

INVERSE UNCERTAINTY PROPAGATION FOR DEMAND DRIVEN DATA ACQUISITION

Philipp Baumgärtel
Gregor Endler
Andreas M. Wahl
Richard Lenz

Department of Computer Science (Data Management)
Friedrich-Alexander University Erlangen-Nürnberg (FAU)
Martensstraße 3, D-91058 Erlangen, GERMANY

ABSTRACT

When using simulations for decision making, no matter the domain, the uncertainty of the simulations' output is an important concern. This uncertainty is traditionally estimated by propagating input uncertainties forward through the simulation model. However, this approach requires extensive data collection before the output uncertainty can be estimated. In the worst case scenario, the output may even prove too uncertain to be usable, possibly requiring multiple revisions of the data collection step. To reduce this expensive process, we propose a method for inverse uncertainty propagation using Gaussian processes. For a given bound on the output uncertainty, we estimate the input uncertainties that minimize the cost of data collection and satisfy said bound. That way, uncertainty requirements for the simulation output can be used for demand driven data acquisition. We evaluate the efficiency and accuracy of our approach with several examples.

1 INTRODUCTION

In simulation projects, several factors need to be taken into account in order to receive reliable results. Besides the challenges of simulation modeling and validation, the uncertainty of simulation input data affects the reliability of the results. Real world data is often used to estimate the parameters of probability distributions that model random events in the simulation (Law 2007). This introduces the sampling error of the real world data as an additional source of uncertainty.

As there are different kinds of uncertainty in simulations that need to be considered (O'Hagan 2006), we define the nomenclature used throughout this paper as follows: *Aleatory uncertainty* results from system inherent variability, which is represented by probability distributions. *Epistemic uncertainty* or *input uncertainty* results from a lack of knowledge about the parameters of these distributions stemming from a lack of real world data. Lastly, we need to consider *code uncertainty* arising from using mathematical models to approximate the simulation response. These metamodels are commonly used to analyze the output of black box simulation models (Santner, Williams, and Notz 2003). To assess the uncertainty of the simulation output, we need to propagate the input uncertainty through the simulation model or metamodel while considering aleatory and code uncertainty. While building metamodels requires some effort, simulation output analysis and optimization call for a high number of simulation runs in any event. Thus, we do not consider the effort of running the simulation as existing output data can be leveraged when building metamodels.

Barton (2012) gives an overview of several existing approaches to deal with input uncertainty in simulations. For example, Barton, Nelson, and Xie (2010) use stochastic kriging metamodels developed by Ankenman, Nelson, and Staum (2010) to approximate the simulation response and estimate the output uncertainty. Barton, Nelson, and Xie (2010) use a bootstrapping approach to propagate the input uncertainty

through the metamodel. Girard and Murray-Smith (2005) developed an approach closely related to this. For Gaussian process metamodels, which are a special case of kriging metamodels, Girard and Murray-Smith (2005) found closed-form solutions for approximate and exact uncertainty propagation.

Ankenman and Nelson (2012) developed a method to assess the impact of the input uncertainty on the simulation output. This method and its improved version (Song and Nelson 2013) allow to identify the input parameters with dominant influence on simulation output uncertainty. Therefore, they are able to "identify the input data sources from which additional observations would lead to the greatest reduction in input uncertainty" (Song and Nelson 2013). However, they state as a remaining unsolved problem: "Open questions remain about the design of the follow-up experiment, and in particular the total budget N that should be expended on this experiment to obtain reliable results." (Song and Nelson 2013)

The contribution of this paper is the solution to this problem. After reviewing related work in Section 2, we define demand driven data acquisition in Section 3. Our work is based on the method for uncertainty propagation through Gaussian processes (Girard and Murray-Smith 2005), which we recapitulate in Section 4. We utilize the Cramér-Rao bound (Cramér 1946) to estimate the number of measurements that is required to reduce the uncertainty of each input parameter below a certain threshold. The number of measurements can then be used as a cost function for the data acquisition process. To avoid spending unnecessary effort during data collection by accumulating data with little impact on simulation output, we provide an efficient method to find the data acquisition strategy with minimal costs that results in the desired target uncertainty of the simulation's output. We call this process *inverse uncertainty propagation* (IUP) and provide a detailed description in Section 5. Inverse uncertainty propagation can be used both for demand driven data acquisition and to assess the feasibility of simulation goals.

We use a discrete event simulation of an $M/M/1$ queue as a running example throughout the paper. To evaluate our approach, we assess its efficiency and accuracy using real world discrete event simulations from a configurable communication middleware (Fischer, Wahl, and Lenz 2014) in Section 6. Additionally, we use synthetic data to evaluate the efficiency of our approach for simulations with a large number of input parameters.

2 RELATED WORK

The estimation of input parameters and their uncertainty based on measurements of the output of the real system is called backward or inverse uncertainty quantification. This is distinct from our approach, as our method does not rely on measurements of the output of a real system. Chantrasmi and Iaccarino (2012) use a Bayesian approach to estimate a probability density function for an input parameter based on measured output data. The estimated density of the input parameters corresponds to the uncertainty based on the lack of measured output data. Mares, Mottershead, and Friswell (2006) use a gradient based optimization method to fit model parameters and estimated uncertainties to measured output data. Fonseca et al. (2005) use a maximum likelihood approach to estimate the uncertainty of an input variable based on experimental output data. Arendt, Chen, and Apley (2011) try to optimize the identifiability of uncertainty sources by combining the information of multiple output variables. All of these approaches rely on measured output data of the real system to fit their models.

The estimation of bounds for input uncertainties based on accuracy requirements for the simulation output can be done in a similar way using these optimization approaches. However, as we will show in our evaluation, numerical optimization is not suitable for simulations with a high number of input variables.

3 DEMAND DRIVEN DATA ACQUISITION

To formalize the goal of our approach, this chapter defines the cost function for demand driven data acquisition. We assume that a maximum likelihood estimator $\hat{x}$ is used to estimate the unknown parameter u of a statistical distribution $p(d,u)$ from dataset $(d_1,\ldots,d_n)$. This is known as simulation input modeling, an important step in every stochastic simulation (Biller and Gunes 2010, Law 2007). Several distributions

represent random processes within a simulation and the parameters of these distributions together serve as the input parameters of the simulation model.

Maximum likelihood estimators for distribution parameters are well studied and applicable to a wide range of problems. It is noteworthy that the estimator $\hat{x}$ itself is a random variable. The distribution of a parameter estimated by the maximum likelihood method is asymptotically normal (Cramér 1946). Therefore, the uncertainty about an input parameter u can be represented by the variance of a normal distribution. In the following, we use "uncertainty" and "variance" synonymously.

The variance of an estimator is bounded by the Cramér-Rao bound (Cramér 1946), which is the reciprocal of the Fisher information. The Fisher information $\mathscr{I}$ is a measure of the amount of information gained by one observation (Cramér 1946):

$$\mathscr{I}(u) = \mathrm{E}\left[\left(\frac{\partial \log p(d,u)}{\partial u}\right)^2\right] = -\mathrm{E}\left[\frac{\partial^2 \log p(d,u)}{\partial u^2}\right]$$

As the Fisher information of n observations is n times the Fisher information of one observation, the Cramér-Rao bound for n observations is $\mathrm{var}(\hat{x}) \geq (n\mathscr{I}(u))^{-1}$. An estimator is called "efficient" if the equality is achieved. Maximum likelihood estimators are asymptotically efficient (Cramér 1946). Now, we can use the Cramér-Rao bound to estimate the number of measurements n required to achieve a specific uncertainty $v = \mathrm{var}(\hat{x})$: $n \geq (v\mathscr{I}(u))^{-1}$. With c being the cost of one measurement, the cost to achieve the uncertainty v with an efficient estimator is: $\mathrm{cost}(v) = c/(v\mathscr{I}(u))$.

However, this approach requires knowing u upfront to estimate the achievable variance for $\hat{x}$. We can either assume to know an approximation of u or we can use best, worst and average case estimations for $\mathscr{I}(u)$. We will explain this in more detail in Section 5.4.

For a given uncertainty v, we can now estimate the required number of measurements and the cost to acquire them. This cost function can be used for every distribution parameter that is estimated by a series of measurements using maximum likelihood estimators. Therefore, the cost function covers a wide range of problems. There is also a multivariate version of the Cramér-Rao bound, which defines the cost function for distributions with more than one parameter (i.e. in this case u is a vector).

3.1 Example

We use the $M/M/1$ queue as a running example throughout the paper. New customers arrive at random with the time between arrivals being sampled from an exponential distribution with arrival rate r_a. A server serves one customer at a time with service times sampled from an exponential distribution with rate r_s. The output of the simulation is the number of customers in the system at a certain point in time. We replicate this simulation 1000 times to get the average number of customers in the system.

To determine the parameters r_a and r_s from real world data with a specific uncertainty as input for our simulation, we need the Fisher information of the rate parameter of the exponential distribution: $\mathscr{I}(r) = r^{-2}$. For example, if the arrival rate r_a is 0.5 (i.e. one new customer every 2 time units) and we want to achieve an uncertainty of 0.01 (standard deviation) for this rate, then we need a sample size of $n \geq (v\mathscr{I}(r_a))^{-1} = (0.5/0.01)^2 = 2500$. Because of clarity reasons, we use the standard deviation instead of the variance to represent the uncertainty in our examples.

An obvious problem in this example is that we need to know the value of a parameter in advance to estimate the number of samples required to determine the parameter with a specific uncertainty. In Section 5.4, we will show how to overcome this problem.

4 UNCERTAINTY PROPAGATION

We use Gaussian processes as approximate metamodels of black-box simulations, as Gaussian processes provide means for propagation of uncertainty. In this section, we present the uncertainty propagation

method developed by Girard and Murray-Smith (2005). We will extend this method for inverse uncertainty propagation in the next section.

4.1 Gaussian Processes as Metamodels

Definition 1 "A Gaussian process is a collection of random variables, any finite number of which have (consistent) joint Gaussian distributions." (Rasmussen 2004)

We model a given black-box function $f(\mathbf{x})$ ($\mathbf{x} \in \mathbb{R}^D$) by a Gaussian process $G(\mathbf{x}) \sim GP(m(\mathbf{x}), C(\mathbf{x}_i, \mathbf{x}_j))$. $m(\mathbf{x}) = E[G(\mathbf{x})]$ is the mean function of the Gaussian process and $C(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{Cov}[G(\mathbf{x}_i), G(\mathbf{x}_j)]$ is the covariance function. For any given set of inputs $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, $(G(\mathbf{x}_1), \ldots, G(\mathbf{x}_n))$ is a random vector with an n-dimensional normal distribution $\mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$. $\mathbf{m}$ is the vector of mean values $(m(\mathbf{x}_1), \ldots, m(\mathbf{x}_n))$ and $\mathbf{\Sigma}$ is the covariance matrix with $\Sigma_{ij} = C(\mathbf{x}_i, \mathbf{x}_j)$ (Girard and Murray-Smith 2005).

For modeling $f(\mathbf{x})$, we use the Gaussian squared exponential covariance function $C(\mathbf{x}_i, \mathbf{x}_j)$ $= v \exp\left[-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{W}^{-1} (\mathbf{x}_i - \mathbf{x}_j)\right]$, as it performs well regarding accuracy (Rasmussen 1996). However, the methods presented in this paper are independent of the actual covariance function. Without loss of generality, we use $m(\mathbf{x}) = 0$.

$\mathbf{W}^{-1} = \mathrm{diag}(w_1, \ldots, w_D)$ and v are hyperparameters of the Gaussian process. We have a set of observed simulation results $\mathscr{D} = \{(\mathbf{x}_i, t_i)\}_{i=1}^N$ as supporting points with $\mathbf{x}_i \in \mathbb{R}^D$ and $t_i = f(\mathbf{x}_i) + \varepsilon_i$ with white noise $\varepsilon_i \sim \mathcal{N}(0, v_t)$. This noise represents the aleatory uncertainty of the simulation that calculates $f(\mathbf{x})$. To model noisy observations Girard and Murray-Smith (2005) use $\mathbf{K} = \mathbf{\Sigma} + v_t \mathbf{I}$ as covariance matrix for regression. Therefore, we have an additional hyperparameter v_t. The hyperparameters $\mathbf{W}^{-1}$, v_t, and v can be estimated from a given dataset with a maximum likelihood approach. In the following, we use $\beta = \mathbf{K}^{-1}\mathbf{t}$ with $\mathbf{t} = (t_1, \ldots, t_N)^T$ to simplify notation.

Now the Gaussian process can be used as a surrogate for the simulation model. The input parameter vector $\mathbf{x}$ consists of the parameters of all input distributions of the simulation and can be estimated from real world data. The prediction $\mu(\mathbf{x})$ and code plus aleatory uncertainty $\sigma^2(\mathbf{x})$ at a new input $\mathbf{x}$ is (Girard and Murray-Smith 2005):

$$\mu(\mathbf{x}) = \sum_{i=1}^{N} \beta_i C(\mathbf{x}, \mathbf{x}_i) \qquad \sigma^2(\mathbf{x}) = C(\mathbf{x}, \mathbf{x}) - \sum_{i,j=1}^{N} K_{ij}^{-1} C(\mathbf{x}, \mathbf{x}_i) C(\mathbf{x}, \mathbf{x}_j) + v_t$$

With this covariance function, v_t models a constant aleatory uncertainty. The metamodel developed by Ankenman, Nelson, and Staum (2010) use a more sophisticated representation for the aleatory uncertainty. However, as we will show in the following sections, we are not restricted to a specific covariance function and are able to incorporate other more complex covariance functions modeling a variable aleatory uncertainty.

4.2 Gaussian Processes for Uncertainty Propagation

We assume that the input parameters will be estimated from real world data using a maximum likelihood approach. Under this assumption, we can represent the uncertainty about an input parameter as a normal distribution. For an uncertain input $\mathbf{x} \sim \mathcal{N}(\mathbf{u}, \mathbf{\Sigma}_x)$, the output distribution can be written as (Girard and Murray-Smith 2005):

$$p(y|\mathscr{D}, \mathbf{u}, \mathbf{\Sigma}_x) = \int_{-\infty}^{\infty} p(y|\mathscr{D}, \mathbf{x}) p(\mathbf{x}|\mathbf{u}, \mathbf{\Sigma}_x) \, \mathrm{d}\mathbf{x}$$

Here, $\mathbf{\Sigma}_x$ is the covariance matrix of the uncertain input and incorporates the uncertainty and dependencies between the input parameters. As the distribution $p(y|\mathscr{D}, \mathbf{u}, \mathbf{\Sigma}_x)$ is hard to determine analytically, Girard

and Murray-Smith (2005) approximate its mean and variance:

$$
\begin{aligned}
m(\mathbf{u},\mathbf{\Sigma}_x) &= \mu(\mathbf{u}) + \tfrac{1}{2}\sum_{i=1}^{N}\beta_i \mathrm{Tr}[\mathbf{C}''(\mathbf{u},\mathbf{x_i})\mathbf{\Sigma}_x] \\
v(\mathbf{u},\mathbf{\Sigma}_x) &= \sigma^2(\mathbf{u}) + \tfrac{1}{2}\mathrm{Tr}[\mathbf{C}''(\mathbf{u},\mathbf{u})\mathbf{\Sigma}_x] - \sum_{i,j=1}^{N}(K_{ij}^{-1} - \beta_i\beta_j)\mathrm{Tr}[\mathbf{C}'(\mathbf{u},\mathbf{x}_i)\mathbf{C}'(\mathbf{u},\mathbf{x}_j)^T\mathbf{\Sigma}_x] \\
&\quad - \tfrac{1}{2}\sum_{i,j=1}^{N}K_{ij}^{-1}\Big(C(\mathbf{u},\mathbf{x}_i)\mathrm{Tr}[\mathbf{C}''(\mathbf{u},\mathbf{x}_j)\mathbf{\Sigma}_x] + C(\mathbf{u},\mathbf{x}_j)\mathrm{Tr}[\mathbf{C}''(\mathbf{u},\mathbf{x}_i)\mathbf{\Sigma}_x]\Big) \qquad (1)
\end{aligned}
$$

$\mathbf{C}'$ is the Gradient and $\mathbf{C}''$ is the Hessian of the covariance function. In case of the squared exponential, $\mathbf{C}''(\mathbf{u},\mathbf{u})$ equals 0. As this approach makes no special assumptions about the covariance function, other functions can be plugged into this method as long as they are twice differentiable.

We use this approximation for the output uncertainty $v(\mathbf{u},\mathbf{\Sigma}_x)$ as a basis for our inverse uncertainty propagation. At this point, the uncertainty propagation uses two layers of approximation: First, the simulation is approximated by a Gaussian process and second, the uncertainty propagation is done using an approximate formula.

Using this approach, we can clearly distinguish between aleatory, epistemic and code uncertainty. Aleatory uncertainty is represented by v_t, code uncertainty is represented by $\sigma^2(\mathbf{u}) - v_t$ and epistemic uncertainty is $v(\mathbf{u},\mathbf{\Sigma}_x) - \sigma^2(\mathbf{u})$.

4.3 Example

We use 1000 random supporting points from the $M/M/1$ simulation with the input parameters $\mathbf{x} = (r_a, r_s)$. Assuming $\mathbf{x}$ to be around $(0.5, 0.6)$, we draw the supporting points from the set $[0.475, 0.525] \times [0.575, 0.625]$. We consider the uncertain input $\mathbf{x} \sim \mathcal{N}(\mathbf{u},\mathbf{\Sigma}_x)$ with mean $\mathbf{u} = (0.5, 0.6)$ and standard deviation 0.005 for each input parameter. By using a Monte Carlo method with 10^5 samples for uncertainty propagation through the simulation and Girard's approximate method with the Gaussian process, we can estimate the output uncertainty (standard deviation): Monte Carlo ≈ 0.4145 (2.9 h); Approximation ≈ 0.4182 (uncertainty propagation: 0.5 s, 1000 samples for the metamodel: 18 min). We used 10^5 samples for the Monte Carlo method, as this yielded a suitable 95% confidence interval for the output uncertainty: $(0.4088, 0.4203)$.

5 INVERSE UNCERTAINTY PROPAGATION

In this section, we extend the methods for uncertainty propagation to find input uncertainties that produce a given output uncertainty v_{out}. As there are multiple input variables, there are many possible combinations of input uncertainties that lead to v_{out}. However, these solutions may result in different costs for the data acquisition. Hence, we present a method that finds the input uncertainties that minimize the cost of data collection.

As in Section 3, we assume that measurements are utilized to estimate the uncertain simulation input parameter vector $\mathbf{x} \sim \mathcal{N}(\mathbf{u},\mathbf{\Sigma}_x)$, which serves as the parameters of several input distributions. Additionally, we assume the variances of the input parameters to be independent from each other: $\mathbf{\Sigma}_x = \mathrm{diag}(\mathbf{v}) = \mathrm{diag}(v_1,\ldots,v_n)$. This assumption holds as for most input distributions an orthogonal parameterization can be found that leads to independent maximum likelihood estimates for each input parameter (Jeffreys 1961). The cost to achieve the input variance vector $\mathbf{v} = (v_1,\ldots,v_n)$ is based on the cost function for one parameter from Section 3:

$$\mathrm{cost}(\mathbf{v}) = \sum_{h=1}^{n}\frac{c_h}{v_h\mathscr{I}_h(u_h)} = \sum_{h=1}^{n}\frac{\kappa_h}{v_h}$$

To simplify notation, we introduce $\kappa_h = c_h/\mathscr{I}_h(u_h)$. Again, we need the true value $\mathbf{u} = (u_1,\ldots,u_n)$ of the simulation input parameters. This constraint will be relaxed in Section 5.4.

We want to find the input variances that minimize the cost function and satisfy the bound v_{out} on the output uncertainty. We write $v(\mathbf{u}, \mathbf{\Sigma}_x)$ from (1) as $v(\mathbf{v})$ to simplify notation. It is noteworthy that $v(\mathbf{v})$ is an affine transformation. Now, we can write the bound on the output uncertainty as an equality constraint: $v(\mathbf{v}) = v_{\text{out}}$.

Actually, this is originally an inequality constraint as the output uncertainty is allowed to be less than the desired one. However, we know that the optimum lies on the boundary because of the structure of the optimization problem. Now, the method of Lagrange multipliers allows us to find the minimum of cost($\mathbf{v}$) that satisfies this equality constraint. To this end, we need to find the solution $\mathring{\mathbf{v}}$ to the following system of equations:

$$v(\mathring{\mathbf{v}}) = v_{\text{out}}; \qquad \nabla_{\mathbf{v}} \text{cost}(\mathring{\mathbf{v}}) = -\mu \nabla_{\mathbf{v}} v(\mathring{\mathbf{v}})$$

This set of equations can be solved for v_h, which has to be positive as it is a variance:

$$\mathring{v}_h = \left(\frac{\mu}{\kappa_h} \frac{\partial v(\mathring{\mathbf{v}})}{\partial v_h} \right)^{-1/2}, \qquad \forall h = 1 \dots n \tag{2}$$

As $v(\mathbf{v})$ is an affine transformation, $\partial v(\mathring{\mathbf{v}})/\partial v_h$ is a constant:

$$\frac{\partial v(\mathring{\mathbf{v}})}{\partial v_h} = \sum_{i,j=1}^{N} \left[(\beta_i \beta_j - K_{ij}^{-1}) [\mathbf{C}'(\mathbf{u}, \mathbf{x}_i) \mathbf{C}'(\mathbf{u}, \mathbf{x}_j)^T]_{hh} - \tfrac{1}{2} K_{ij}^{-1} \left(C(\mathbf{u}, \mathbf{x}_i) [\mathbf{C}''(\mathbf{u}, \mathbf{x}_j)]_{hh} + C(\mathbf{u}, \mathbf{x}_j) [\mathbf{C}''(\mathbf{u}, \mathbf{x}_i)]_{hh} \right) \right]$$

Here $[\mathbf{A}]_{ij}$ denotes the element of the matrix $\mathbf{A}$ in the ith row and the jth column. Now, we can substitute the $\mathring{v}_h$ from (2) as $\mathbf{v}$ into (1) and solve it for μ.

Additionally, we incorporate input parameters with a fixed uncertainty. We assume o input parameters with fixed variance $(\tilde{v}_1, \dots, \tilde{v}_o)$ in addition to the first n parameters for which the desired variance $(\mathring{v}_1, \dots, \mathring{v}_n)$ is unknown. We can split up the matrices in (1) into block matrices with $\lambda = \mu^{-1/2}$:

$$\mathbf{\Sigma}_x = \begin{pmatrix} \lambda \mathbf{\Sigma}_{x1} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_{x2} \end{pmatrix}, \qquad \lambda \mathbf{\Sigma}_{x1} = \text{diag}(\mathring{v}_1, \dots, \mathring{v}_n), \qquad \mathbf{\Sigma}_{x2} = \text{diag}(\tilde{v}_1, \dots, \tilde{v}_o)$$

$$\mathbf{C}'(\mathbf{u}, \mathbf{x}_i) = \begin{pmatrix} \mathbf{C}'_{i1} \\ \mathbf{C}'_{i2} \end{pmatrix}, \qquad \mathbf{C}''(\mathbf{u}, \mathbf{x}_i) = \begin{pmatrix} \mathbf{C}''_{i11} & \mathbf{C}''_{i12} \\ \mathbf{C}''_{i21} & \mathbf{C}''_{i22} \end{pmatrix}$$

By using the rules for matrix multiplication and (1) we get:

$$\lambda = \left[\sum_{i,j=1}^{N} (K_{ij}^{-1} - \beta_i \beta_j) \text{Tr}[\mathbf{C}'_{i1} \mathbf{C}'^T_{j1} \mathbf{\Sigma}_{x1}] + \tfrac{1}{2} \sum_{i,j=1}^{N} K_{ij}^{-1} \left(C(\mathbf{u}, \mathbf{x}_i) \text{Tr}[\mathbf{C}''_{j11} \mathbf{\Sigma}_{x1}] + C(\mathbf{u}, \mathbf{x}_j) \text{Tr}[\mathbf{C}''_{i11} \mathbf{\Sigma}_{x1}] \right) \right]^{-1}$$
$$\cdot \left[\sigma^2(\mathbf{u}) - v_{\text{out}} - \sum_{i,j=1}^{N} \left[(K_{ij}^{-1} - \beta_i \beta_j) \text{Tr}[\mathbf{C}'_{i2} \mathbf{C}'^T_{j2} \mathbf{\Sigma}_{x2}] + \tfrac{1}{2} K_{ij}^{-1} \left(C(\mathbf{u}, \mathbf{x}_i) \text{Tr}[\mathbf{C}''_{j22} \mathbf{\Sigma}_{x2}] + C(\mathbf{u}, \mathbf{x}_j) \text{Tr}[\mathbf{C}''_{i22} \mathbf{\Sigma}_{x2}] \right) \right] \right]$$

Resubstituting $\mu = \lambda^{-2}$ into (2) yields the $\mathring{v}_h$ and therefore the optimal solution $\mathring{\mathbf{v}}$ that minimizes the cost function.

In addition to the approximate uncertainty propagation, which we use for our inverse propagation, Girard and Murray-Smith (2005) also provide exact formulas for the mean and variance of the output distribution. As utilizing Girard's exact method for inverse uncertainty propagation leads to nonlinear equations, we use numerical optimization to find the solution. Additionally, Girard's exact formulas are restricted to a specific kind of covariance function, which restricts the flexibility of this approach. In our evaluation, we compare the accuracy of the exact and the approximate uncertainty propagation and the performance of the different approaches.

In the next section, we will use our running example to illustrate our inverse uncertainty propagation. In Section 5.2, 5.3, and 5.4, we will address several other problems that arise when this method is applied to real simulations.

5.1 Example

We use the Gaussian process for the $M/M/1$ simulation from our previous example in Section 4.3. The $M/M/1$ queue uses two exponential distributions to represent random processes. Each exponential distribution is parameterized by a rate parameter. In the previous example, we considered the uncertain input $\mathbf{x}$ with mean $\mathbf{u} = (r_a, r_s) = (0.5, 0.6)$ and standard deviation $\sigma_{r_a} = \sigma_{r_s} = 0.005$ for each input parameter. The output standard deviation was around 0.4.

In this example, we want to know the minimal number of measurements for each input parameter to achieve an output standard deviation of at most 0.2. In Section 3.1, we described the Fisher information for exponential distributions, which now serves as the cost function. We define the cost for one measurement to be 1. Table 1 shows the resulting optimal solutions and the total number of measurements required to achieve this result. The approximate solution is the result of our IUP method from this section. Additionally, we used the Cobyla optimization algorithm (Powell 1994) in combination with Girard's exact propagation to find the exact optimal solution. It is noteworthy that the exact solution still uses the Gaussian process approximation instead of the real simulation. We use the exact uncertainty propagation for both solutions to evaluate whether we achieve the desired output standard deviation. The approximate IUP is much faster than using the exact IUP and finds a solution that is near optimal. It underestimates the total number of measurements slightly and the output standard deviation is only $4 \cdot 10^{-5}$ above the desired bound.

Table 1: Inverse uncertainty propagation for the $M/M/1$ queue.

	Runtime (seconds)	σ_{r_a}	σ_{r_s}	Exact output std. dev.	Total number of measurements
Approximate IUP	0.18	$9.2821 \cdot 10^{-4}$	$1.11389 \cdot 10^{-3}$	0.20004	580314
Exact IUP	5.51	$9.3216 \cdot 10^{-4}$	$1.10627 \cdot 10^{-3}$	0.20000	581872

5.2 Coestimated Parameters

Some input distributions may contain several parameters, which are estimated from the same sample using the maximum likelihood method. Most distributions can be parameterized in a way that renders the estimates for its parameters independent. However, the cost function has to incorporate the fact that all parameter estimates for this distribution stem from the same sample. Hence, parameters i and j are estimated with the same sample size $n_i = n_j$ and therefore using the Cramér-Rao bound: $v_i \mathscr{I}_i(u_i) = v_j \mathscr{I}_j(u_j)$ (Section 3). This can be incorporated in our IUP method.

5.3 Multiple Constraints

In most cases simulations generate several results and therefore there could be a constraint on each of the output uncertainties. We have the cost function $\text{cost}(\mathbf{v})$ and several functions $v_1(\mathbf{v}), \ldots, v_q(\mathbf{v})$ for estimating the output uncertainties. Now, we have several inequality constraints, as the optimum no longer lies on all boundaries:

$$v_k(\mathbf{v}) \leq v_{k,\text{out}}, \forall k = 1, \ldots, q$$

The method of Karush-Kuhn-Tucker (KKT) conditions, which is a generalization of the method of Lagrange multipliers, allows us to find the optimum for a convex cost function and convex functions in the inequality constraints. As $\text{cost}(\mathbf{v})$ and $v_k(\mathbf{v})$ are convex, we can apply this method to our problem. The $v_k(\mathbf{v})$ are affine functions, therefore all additional regularity conditions of the KKT conditions are automatically met.

5.4 Unknown Input Parameters

In Section 3, we assumed to know the true value $\mathbf{u}$ of uncertain input parameters in advance and in Section 5, we needed the true value of the input parameters for inverse uncertainty propagation. In order to be able to use our method for real simulations, we need to relax this constraint. Using our method from Section 5, we obtain the optimal data collection strategy (number of samples) by estimating the input variances leading to the lowest data acquisition cost. Our framework is basically a deterministic function $\mathrm{Opt}(\mathbf{u})$, which determines the optimal costs and data collection strategy for a given $\mathbf{u}$. Now, we can ask experts to give us an approximation for $\mathbf{u}$ or we can collect a small sample of data to get a rough preliminary estimation for $\mathbf{u}$. We can represent this uncertain knowledge as any arbitrary distribution and use well known uncertainty propagation methods for deterministic functions (Lee and Chen 2007). That way, we are able to get best, worst and average case estimates for the data collection costs and the required number of samples.

5.4.1 Example

For our example, we use Genz-Keister numerical integration rules (Genz and Keister 1996) for numerical uncertainty propagation as they are not affected by the curse of dimensionality. The number of function evaluations depends only polynomially on the number of input parameters (Novak and Ritter 1999). Again, we use our $M/M/1$ queue example and assume to know $\mathbf{u} \approx (0.5, 0.6)$ with standard deviation 0.005 for each parameter which leads to an output standard deviation of about 0.4. Now, we want to know the minimal number of measurements for each input parameter to achieve an output standard deviation of at most 0.2.

We propagate the input uncertainty through $\mathrm{Opt}(\mathbf{u})$ numerically using 45 function evaluations and use a Pearson distribution system (Cramér 1946) to model the cost distribution. We compare this distribution to a normal approximation of the cost distribution. Additionally, we use a Monte Carlo approach with 10^5 function evaluations to get a histogram representing the output distribution. The results are depicted in Figure 1(a). The Pearson system is well suited to represent the cost distribution. Especially for estimating the 95% confidence interval of the data collection cost, the Pearson system approximates the tails of the distribution better than a normal distribution. The estimated 95% confidence intervals for the data collection cost are: Normal : $(236437, 1014173)$; Pearson : $(385853, 1131650)$; Monte Carlo : $(380105, 1134960)$.

6 EVALUATION

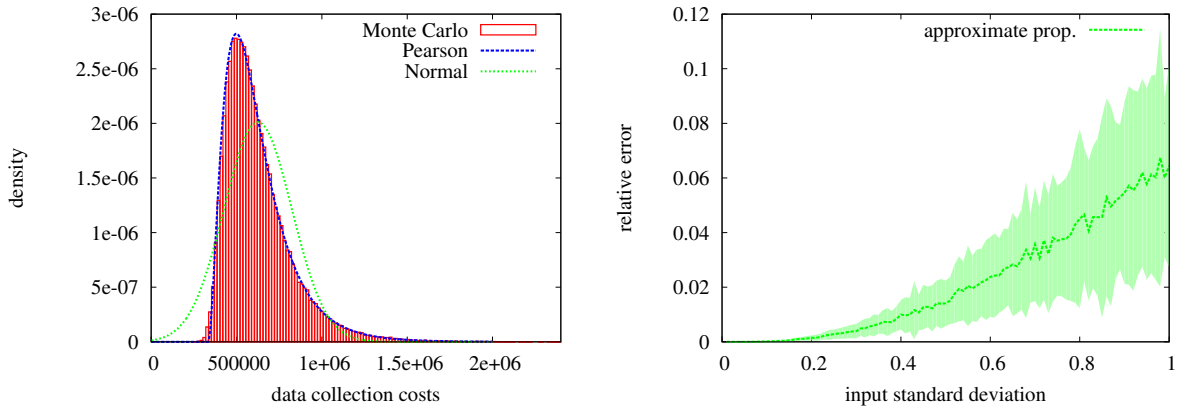
In this section, we use a real world example to evaluate the precision of our methods. Additionally, we evaluate the performance of our prototype and compare it to traditional optimization methods. Our prototypic implementation uses Python, Scipy and Numpy to perform the calculations.

6.1 Real World Example

We use M^{2}etis (Massive Multiuser EvenT InfraStructure), a configurable communication middleware (Fischer, Wahl, and Lenz 2014), as a real world evaluation scenario. M^{2}etis derives an optimal configuration of the middleware by using discrete event network simulations and supervised learning techniques.

We used results from the M^{2}etis simulator to analyze the average delivery latency in a content delivery scenario: A group of 100 nodes received messages from a single sender over an IPv6-based overlay network with a symmetric bandwidth of 1 Gbit/s. Messages were sent at 10 Hz and carried a payload of 1024 Byte. All network nodes were communicating directly without intermediate overlay hops. To ensure delivery, acknowledgements for each message were sent by the recipients.

As input distributions, we use a normal distribution with its mean m_e and standard deviation σ_e to model the network delay. Additionally, we use probability p_e to model packet loss. From previous measurements we know the approximate mean of these parameters: $\mathbf{u}_e = (m_e, \sigma_e, p_e) = (15.05, 5, 0.025)$



(a) Distribution of the data collection cost.

(b) Approximate propagation.

Figure 1: Cost distribution example (a) and evaluation of the relative error (b).

with input uncertainty $\mathbf{\Sigma}_e = \text{diag}(2^2, 1^2, 0.005^2)$. The simulation output (average delivery latency) is about 0.3. Considering the input uncertainty, the output uncertainty is about 0.041 (standard deviation).

We want to know the number of measurements to estimate $\mathbf{u}$ well enough to get an output uncertainty below 0.01. To this end, we build a Gaussian process metamodel with 1000 supporting points in the area of interest from the simulation and use our techniques from Section 5. It is noteworthy that m_e and σ_e are estimated from the same series of measurements, as they belong to the same distribution. If we assume $\mathbf{u} = \mathbf{u}_e$ to be exact, our inverse uncertainty propagation estimates a sample size of 453 measurements for m_e and σ_e and 523 measurements for p_e. To evaluate, whether our estimated number of measurements is sufficient to get an output uncertainty below 0.01 from the real simulation, we use Monte Carlo uncertainty propagation. This yields a real output uncertainty of 0.0112, which is only slightly above the desired one.

Now, we relax the assumption to know $\mathbf{u}$ exactly and use only our knowledge from previous experiments $\mathbf{u} \sim \mathcal{N}(\mathbf{u}_e, \mathbf{\Sigma}_e)$. As in Section 5.4.1, we propagate this uncertainty through $\text{Opt}(\mathbf{u})$ and compare Monte Carlo propagation with 10^5 function evaluations to Genz-Keister numerical integration with 165 function evaluations in combination with the Pearson system. The 95% confidence intervals for the total number of measurements required to achieve the output uncertainty of 0.01 are:
Pearson : $(922, 1683)$, Monte Carlo : $(895, 1677)$.

6.2 Performance Evaluation

For performance evaluation, we use synthetic Gaussian processes with an arbitrary number N of supporting points and an arbitrary number of dimensions D. We create a dataset $\{(\mathbf{x}_i, t_i)\}_{i=1}^N$ with N random points $\mathbf{x} \in [0, 10]^D$, $t_i = y_i + \varepsilon_i$ and white noise $\varepsilon_i \sim \mathcal{N}(0, v_t)$ with $v_t = 0.01$. The y_i are a random realization of a Gaussian process with the hyperparameters $w_i = 0.04$ and $v = 2$. This dataset is then used as supporting points for another Gaussian process.

We use the synthetic Gaussian processes to evaluate the error of Girard's approximate propagation compared to Girard's exact propagation (Girard and Murray-Smith 2005). Here, we only evaluate the forward propagation, as our inverse method finds the exact optimum for the approximate propagation and does not introduce an additional error. To this end, we use $N = 100$ and $D = 5$ and evaluate the error of the approximate propagation relative to the exact method. For all input parameters, we use the same input uncertainty. In Figure 1(b), we depict the mean and standard deviation of the relative error of 100 measurements per experiment. Here, the relative error of the approximate solution is always below 10%. Additionally, our examples showed that the accuracy of the approximate method is suitable for real world problems.

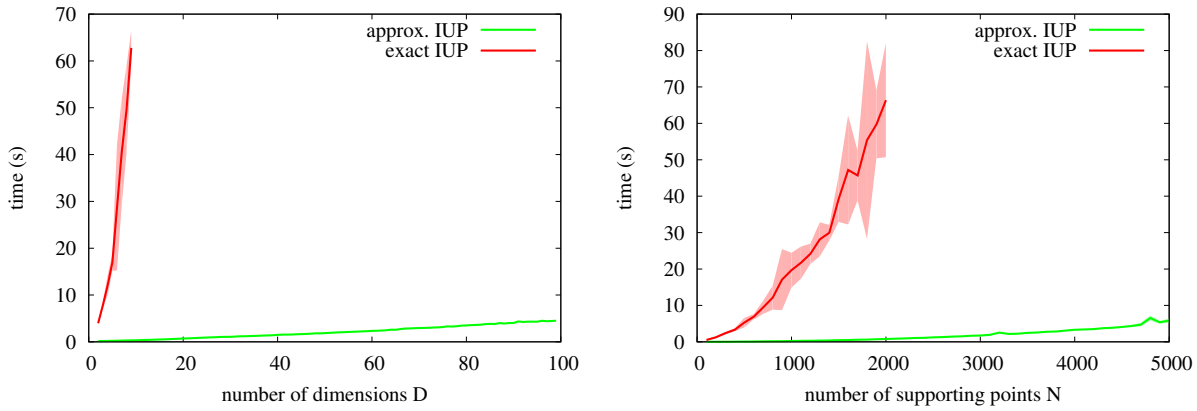


Figure 2: Performance of the IUP methods depending on (a) D and (b) N.

We compare our inverse method from Section 5 to the Cobyla optimization algorithm (Powell 1994) in combination with Girard's exact propagation. For this problem, Cobyla is the fastest and most reliable optimization algorithm from the Scipy library. For each experiment, we depict the mean and standard deviation of 25 measurements. Figure 2(a) depicts the performance of the optimization solutions for $N = 1000$ depending on D. Figure 2(b) shows the performance for $D = 5$ depending on N.

We do not evaluate methods for coping with unknown input parameters as described in Section 5.4 because Lee and Chen (2007) already evaluated methods for propagating uncertainties through deterministic black-box functions. However, our evaluation shows that only our approximate method is fast enough to be used for further numerical analyses.

7 CONCLUSIONS AND FUTURE WORK

In this paper, we presented a methodology to use bounds on the output uncertainties of simulations for demand driven data acquisition. We use Gaussian processes and Girard's forward propagation method (Girard and Murray-Smith 2005) to estimate input uncertainties for a given bound. We defined a cost function based on simulation input modeling and the Cramér-Rao bound to estimate the number of measurements required to achieve the desired bound. The contribution of this paper is a novel approach to find the optimal input uncertainties that satisfy the bound on the simulation output uncertainty. With our approach, we are able to find an approximation to the optimal solution analytically. As many simulations rely on simulation input modeling and measurements, this method is applicable to a wide range of problems. Simulation experts can use prototypical models and rough estimates for simulation parameters to find the optimal data collection strategy. Our inverse uncertainty propagation solves the open question from Song and Nelson (2013) about the number of measurements required to get reliable results from simulations.

We evaluated our approach using a real world example from communication middleware simulations (Fischer, Wahl, and Lenz 2014). Additionally, we evaluated the accuracy and performance of our approach using synthetic Gaussian processes. Our evaluation showed that our approach is precise enough for real world applications and very fast compared to numerical optimization methods. Therefore, we are able to manage simulations with more than 100 input parameters and Gaussian processes with more than 5000 data points. However, we need to be able to cope with much more dimensions and data points to make our inverse propagation framework applicable for large scale real life scenarios. To this end, we need to investigate how to use Sparse Gaussian Processes (Quinonero-Candela, Rasmussen, and Williams 2007, Snelson and Ghahramani 2006) and dimensionality reduction (Snelson 2006) for inverse uncertainty propagation. As one step in this direction, Groot, Lucas, and van den Bosch (2011) already extended Girard's propagation

methods for Sparse Gaussian Processes. In future work, we want to integrate these solutions into our simulation input management framework, which we described in (Baumgärtel and Lenz 2012).

ACKNOWLEDGMENTS

On behalf of the ProHTA Research Group. This project is supported by the German Federal Ministry of Education and Research (BMBF), project grant No. 13EX1013B.

REFERENCES

Ankenman, B., and B. Nelson. 2012. "A Quick Assessment of Input Uncertainty". In *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. Uhrmacher, 241–250. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Ankenman, B., B. L. Nelson, and J. Staum. 2010, March. "Stochastic Kriging for Simulation Metamodeling". *Oper. Res.* 58 (2): 371–382.

Arendt, P. D., W. Chen, and D. W. Apley. 2011. "Improving Identifiability in Model Calibration Using Multiple Responses". *ASME Conference Proceedings* 2011 (54822): 1213–1222.

Barton, R. 2012. "Tutorial: Input Uncertainty in Output Analysis". In *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. Uhrmacher, 67–78. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Barton, R., B. Nelson, and W. Xie. 2010. "A Framework for Input Uncertainty Analysis". In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, 1189–1198. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Baumgärtel, P., and R. Lenz. 2012. "Towards Data and Data Quality Management for Large Scale Healthcare Simulations". In *Proceedings of the International Conference on Health Informatics*, 275–280: SciTePress - Science and Technology Publications.

Biller, B., and C. Gunes. 2010. "Introduction to Simulation Input Modeling". In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, 49–58. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Chantrasmi, T., and G. Iaccarino. 2012. "Forward and Backward Uncertainty Propagation for Discontinuous System Response Using the Padé-Legendre Method". *International Journal for Uncertainty Quantification* 2 (2): 125–143.

Cramér, H. 1946. *Mathematical Methods of Statistics*. Princeton University Press.

Fischer, T., A. M. Wahl, and R. Lenz. 2014. "Automated QoS-Aware Configuration of Publish-Subscribe Systems at Design-Time". In *Proceedings of the 8th ACM International Conference on Distributed Event-Based Systems*.

Fonseca, J. R., M. I. Friswell, J. E. Mottershead, and A. W. Lees. 2005. "Uncertainty Identification by the Maximum Likelihood Method". *Journal of Sound and Vibration* 288 (3): 587 – 599.

Genz, A., and B. Keister. 1996. "Fully Symmetric Interpolatory Rules for Multiple Integrals Over Infinite Regions with Gaussian Weight". *Journal of Computational and Applied Mathematics* 71 (2): 299 – 309.

Girard, A., and R. Murray-Smith. 2005. "Gaussian Processes: Prediction at a Noisy Input and Application to Iterative Multiple-Step Ahead Forecasting of Time-Series". In *Switching and Learning in Feedback Systems*, edited by R. Murray-Smith and R. Shorten, Volume 3355 of *Lecture Notes in Computer Science*, 158–184. Springer Berlin Heidelberg.

Groot, P., P. Lucas, and P. van den Bosch. 2011. "Multiple-Step Time Series Forecasting with Sparse Gaussian Processes". In *Proceedings of the 23rd Benelux Conference on Artificial Intelligence (BNAIC 2011), Ghent*, 105–112.

Jeffreys, S. H. 1961. *Theory of Probability*. 3rd ed. Oxford University Press.

Law, A. M. 2007. *Simulation Modeling and Analysis*. 4th ed. McGraw Hill.

Lee, S. H., and W. Chen. 2007. "A Comparative Study of Uncertainty Propagation Methods for Black-Box Type Functions". *ASME Conference Proceedings* 2007 (48078): 1275–1284.

Mares, C., J. Mottershead, and M. Friswell. 2006. "Stochastic Model Updating: Part 1 - Theory and Simulated Example". *Mechanical Systems and Signal Processing* 20 (7): 1674 – 1695.

Novak, E., and K. Ritter. 1999. "Simple Cubature Formulas with High Polynomial Exactness". *Constructive Approximation* 15 (4): 499–522.

O'Hagan, A. 2006. "Bayesian Analysis of Computer Code Outputs: A Tutorial". *Reliability Engineering & System Safety* 91 (10-11): 1290 – 1300. The Fourth International Conference on Sensitivity Analysis of Model Output (SAMO 2004) - SAMO 2004.

Powell, M. 1994. "A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation". In *Advances in Optimization and Numerical Analysis*, edited by S. Gomez and J.-P. Hennart, Volume 275 of *Mathematics and Its Applications*, 51–67. Springer Netherlands.

Quinonero-Candela, J., C. E. Rasmussen, and C. K. Williams. 2007. *Large-Scale Kernel Machines*, Chapter Approximation Methods for Gaussian Process Regression, 203–224. Cambridge, MA: MIT Press.

Rasmussen, C. 2004. *Gaussian Processes in Machine Learning*, Volume 3176 of *Lecture Notes in Computer Science*, 63–71. Heidelberg: Springer-Verlag. Copyright by Springer.

Rasmussen, C. E. 1996. *Evaluation of Gaussian Processes and Other Methods for Non-Linear Regression*. Ph. D. thesis, University of Toronto.

Santner, T., B. Williams, and W. Notz. 2003. *The Design and Analysis of Computer Experiments*. Springer Series in Statistics. Springer.

Snelson, E. 2006. "Variable Noise and Dimensionality Reduction for Sparse Gaussian Processes". In *Proceedings of the 22nd Annual Conference on Uncertainty in AI*: AUAI Press.

Snelson, E., and Z. Ghahramani. 2006. "Sparse Gaussian Processes Using Pseudo-Inputs". *Advances in Neural Information Processing Systems* 18.

Song, E., and B. L. Nelson. 2013. "A Quicker Assessment of Input Uncertainty". In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 474–485. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

PHILIPP BAUMGÄRTEL is a member of the research staff at the Department of Computer Science (Data Management) at the Friedrich-Alexander University Erlangen-Nürnberg (FAU). He received a Diplom in computer science in 2010. His research deals with simulation input data management and uncertainty management. His email address is philipp.baumgaertel@fau.de.

GREGOR ENDLER is a member of the research staff at the Department of Computer Science (Data Management) at the FAU. He received a Diplom in computer science in 2009. His research deals with data quality and evolutionary information systems. His email address is gregor.endler@fau.de.

ANDREAS M. WAHL is a member of the research staff at the Department of Computer Science (Data Management) at the FAU. Since 2013 he holds a Master's degree in computer science. He is involved in research on design-time configuration of publish-subscribe middleware. His email address is andreas.wahl@fau.de.

RICHARD LENZ is Professor for Data Management at the FAU. He holds a Diplom in Computer Science from the University of Kaiserslautern, a Dr.-Ing. in Computer Science from the University of Erlangen and a Habilitation in Computer Science from the University of Marburg. His research interests include evolutionary information systems and data quality management. His email address is richard.lenz@fau.de.