

EXPLORATION OF MULTIFIDELITY APPROACHES FOR UNCERTAINTY QUANTIFICATION IN NETWORK APPLICATIONS

Gianluca Geraci¹, Laura P. Swiler¹, Jonathan Crussell² and Bert J. Debuschere³

¹Sandia National Laboratories
Optimization and Uncertainty Quantification, Albuquerque, NM
e-mail: {ggeraci,lpswile}@sandia.gov

²Sandia National Laboratories
Data Science & Cyber Analytics, Livermore, CA
e-mail: jcrusse@sandia.gov

³ Sandia National Laboratories
Reacting Flow Research, Livermore, CA
e-mail: bjdebus@sandia.gov

Abstract. *Communication networks have evolved to a level of sophistication that requires computer models and numerical simulations to understand and predict their behavior. A network simulator is a software that enables the network designer to model several components of a computer network such as nodes, routers, switches and links and events such as data transmissions and packet errors in order to obtain device and network level metrics. Network simulations, as many other numerical approximations that model complex systems, are subject to the specification of parameters and operative conditions of the system. Very often the full characterization of the system and their input is not possible, therefore Uncertainty Quantification (UQ) strategies need to be deployed to evaluate the statistics of its response and behavior. UQ techniques, despite the advancements in the last two decades, still suffer in the presence of a large number of uncertain variables and when the regularity of the systems response cannot be guaranteed. In this context, multifidelity approaches have gained popularity in the UQ community recently due to their flexibility and robustness with respect to these challenges. The main idea behind these techniques is to extract information from a limited number of high-fidelity model realizations and complement them with a much larger number of a set of lower fidelity evaluations. The final result is an estimator with a much lower variance, i.e. a more accurate and reliable estimator can be obtained. In this contribution we investigate the possibility to deploy multifidelity UQ strategies to computer network analysis. Two numerical configurations are studied based on a simplified network with one client and one server. Preliminary results for these tests suggest that multifidelity sampling techniques might be used as effective tools for UQ tools in network applications.*

Keywords: Cyber Analytics, Emulytics, Uncertainty Quantification, Monte Carlo, Multifidelity

1 INTRODUCTION

Uncertainty quantification (UQ) is a field of study drawing from statistics, mathematics, and computational science [37] [29] [6]. Simulation models for engineering and physics applications are often developed to help assess a design or performance requirement. The past few decades have seen an unprecedented increase in the complexity and sophistication of computational simulation models due to improvements in computer architectures/processors as well as in advanced software frameworks. Typically, one does not run a simulation model just once but multiple times, to explore the effects of different parameters and scenarios. The capability to quantify the effects of uncertainty when using a model to inform a scientific or regulatory decision is critical. There have been a number of large-scale regulatory assessments performed using uncertainty quantification on computational models. Notable examples include the performance of geologic repositories for the disposal of nuclear waste [20], computational fluid dynamics for aircraft design, and climate model predictions [38].

The basic framework for uncertainty quantification is identifying and characterizing uncertain input parameters, representing the input uncertainty (typically in the form of probability distributions), propagating uncertainties in the inputs through the model (typically by drawing samples of the uncertain parameters from their respective distributions and running the model at those sample values to create an ensemble of model runs), and analyzing the output to determine statistics on the output quantities of interest. A number of activities related to UQ that can inform the UQ process include sensitivity analysis [36], verification and validation [32], and dimension reduction [4]. There are many related issues and research directions in UQ which include sample design (e.g. how does one choose the input samples at which to run the model), inclusion of other uncertainties (e.g. numerical uncertainties, uncertainties in observational data used to calibrate models, model form uncertainty), and types of uncertainties (e.g. aleatory, epistemic, interval uncertainties). The scientific computing community has endeavored to develop methods which are as efficient as possible to perform UQ on computationally expensive simulation models. In this paper, we present one particular class of UQ methods called multifidelity methods that we feel is well suited for the analysis of network and cyber modeling. Multifidelity UQ techniques have gained popularity in the last decade or so when the need for UQ of high-fidelity numerical simulations led to the design of techniques capable of containing the overall computational burden. In this contribution, the focus is on multifidelity sampling strategies given the features of the network applications. In a broad sense, it is possible to include the so-called multi-level and multi-index approaches [15, 16, 19], multifidelity MC [33, 34], multilevel-multifidelity techniques [11, 7, 14] and approximate control variates [17] in this class of approaches. Multifidelity UQ strategies have been successfully used in a variety of context ranging from turbulent-laden flows in a radiative environment [22], aerospace applications [14] and cardiovascular flows [9]. Our goal in this work is to explore these methods in the context of UQ on computer network applications.

Network models can aid network operators and designers when making decisions. For instance, network operators can use models to understand the potential impacts of changes to their network before affecting the operation system. Network designers can use models to understand design trade-offs before network creation. These models can drastically reduce the cost and risk of deployment. The terminology of network modeling generally designates two distinct choices: simulations and emulations. Simulations are similar to their physics-modeling counterparts: they use a deep understanding of the underlying processes to simulate network components and interactions in software. Emulations, on the other hand, run the real software

on virtualized hardware which allows them to capture unknown or poorly understood behaviors. This realism comes at the cost of increased computational cost.

The reasons for performing uncertainty quantification on network models is similar to that of engineering models: to understand how uncertainty in inputs (such as device and network configuration, threats, and network topology) propagates to network outputs (such as network availability, traffic loads, etc.) In this exploratory study, the focus is on multifidelity sampling UQ strategies which has the potential to naturally treat system responses with noise, bifurcations, or discontinuities in the presence of a large number of uncertain parameters. This scenario is expected to be particularly relevant for network simulations and emulations.

The remainder of the manuscript is organized as it follows. In Section 2 the network modeling approach is described and, in particular, two network softwares are described, namely a simulator `ns-3` and an emulator `minimega`. Section 3 introduces some generalities on the multifidelity sampling approaches. Numerical examples are presented in Section 4. Conclusions close the paper in Section 5.

2 NETWORK MODELS

As stated earlier, there are generally two types of network modeling: network simulation and network emulation. Network simulators rely on careful implementations of how “real systems” respond to inputs and the processes that drive them which makes them useful to study well-understood behaviors of systems but not necessarily emergent behaviors. Depending on what the model is being used for, this could require a very in-depth understanding of the system that we wish to model. Simulations can even aid designers that wish to understand the trade-offs in the underlying processes when the real software has not been created yet. On the other hand, network emulation runs the real software on virtualized hardware which decreases the semantic gap between the model and the operational system.

Comparing simulations and emulations, we find that they have different strengths. Simulations can be fast to develop and capture the core behavior of well-understood system. Since they control the clock, simulations can run faster than real time. Additionally, multiple network simulations can run in parallel because they are not timing dependent or reliant on virtualized hardware which may be limited. This means that we can run many instances of our network simulation for every emulation. Emulations, which run the real software, should more closely match the real systems. In our multifidelity UQ, we aim to leverage the strengths of both forms of modeling. We can use the inexpensive network simulation as our low-fidelity model and the emulation as the high-fidelity model.

In addition to network modeling, network operators and designers may also use physical testbeds in order to understand their systems. Physical testbeds are costly to build and maintain and may not be suitable for all types of tests. Related work has compared network emulation to physical testbeds to discover where and how they differ [5]. In future work, we could expand upon our levels of multifidelity to include results from a physical testbed (or even an operational network) as the highest fidelity.

2.1 The `ns-3` network simulator

`ns-3` [21] is a discrete network simulator for Internet Protocol (IP) and non-IP networks. It has been widely used by the academic community to understand existing and emerging network designs and protocols [8, 31, 35, 39]. `ns-3` allows users to construct simulations from reusable components to configure nodes, topologies, and applications.

Interestingly, ns-3 supports leveraging code from real applications or kernels in the simulator. For example, there are tests to incorporate the entire Linux kernel networking stack. This hybridization of ns-3 likely increases its fidelity which benefits our multifidelity UQ approach since the more correlated our low- and high-fidelity models are, the faster the convergence.

2.2 The minimega network emulator

minimega [28] is a toolset developed by Sandia National Laboratories to launch and manage virtual machines (VMs) to emulate networks. It wraps QEMU [3] and KVM [23] to launch the VMs and Open vSwitch [10] to connect the VMs to virtual networks with user-defined topologies. minimega includes a scriptable interface that includes many APIs to support the experimentation lifecycle such as capturing data and running services.

3 MULTIFIDELITY UNCERTAINTY QUANTIFICATION

In this section a multifidelity sampling approach is described. For this particular application, it is reasonable to assume that the high-fidelity (HF) model is unbiased and that lower accuracy network representations are generated and added to a limited number of HF evaluations in order to decrease the variance of the sampling estimator, *i.e.* increasing its reliability from an user perspective. This is a slightly different scenario than, for instance, a classical multilevel MC application where usually it is possible to control the accuracy (bias) of the high-fidelity model in order to balance the full mean square error of the estimator [16]. For a generic quantity of interest (QoI) of the system, $Q : \mathbb{R}^d \ni \Xi \rightarrow \mathbb{R}$, *e.g.* the number of requests per second processed by a server, the goal is to compute some statistics. In this work, the expected value $\mathbb{E}[Q]$ of the QoI is considered, but an extension to higher-order moments it is also possible. The Monte Carlo (MC) estimator for $\mathbb{E}[Q]$ can be written as

$$\mathbb{E}[Q] = \int_{\Xi} Q(\boldsymbol{\xi})p(\boldsymbol{\xi}) d\boldsymbol{\xi} \approx \hat{Q}_N^{\text{MC}} = \frac{1}{N} \sum_{i=1}^N Q(\boldsymbol{\xi}^{(i)}) = \frac{1}{N} \sum_{i=1}^N Q^{(i)}, \quad (1)$$

where N realizations of the vector of random input $\boldsymbol{\xi} \in \Xi$ are drawn according to the joint probability distribution $p(\boldsymbol{\xi})$. For each realization of the vector of random input $\boldsymbol{\xi}$, the value of the QoI $Q^{(i)} = Q(\boldsymbol{\xi}^{(i)})$ is evaluated by performing a network simulation and extracting the desired quantity. \hat{Q}_N^{MC} represents a random variable itself and, if Q has finite variance $\text{Var}[Q] < \infty$, it is possible to show that the estimator is unbiased, *i.e.* $\mathbb{E}[\hat{Q}_N^{\text{MC}}] = \mathbb{E}[Q]$ and

$$\text{Var}[\hat{Q}_N^{\text{MC}}] = \frac{\text{Var}[Q]}{N}. \quad (2)$$

A classical result, that follows from the central limit theorem, states that for $N \rightarrow \infty$ the error $\hat{Q}_N^{\text{MC}} - \mathbb{E}[Q]$ is distributed as a normal distribution with zero mean and variance equal to $\text{Var}[\hat{Q}_N^{\text{MC}}]$. It follows that the root mean square error (RMSE) is equal to $\text{Var}^{\frac{1}{2}}(Q)/\sqrt{N}$, from which it follows the well known rate of convergence of $\mathcal{O}(N^{-1/2})$ for the MC estimator. The inspection of the RMSE reveals important features of the MC estimator that make it particularly suited for the network applications considered in this work. Albeit the slow rate of convergence corresponds to a limit in obtaining accurate statistics with a limited number of realizations of the QoI (*i.e.* network simulations), it is also possible to note that neither the dimensionality of the system or the smoothness of Q appear in the RMSE. This situation is different from other quadrature rules in which the rate of convergence is ultimately related to the

order of continuous derivatives of the integrand and the number of dimensions. The MC estimator is therefore a convenient, and very often the only practical choice, when one deals with both noisy responses and possibly bifurcations/discontinuities of the system response. Both cases are common in network simulations. Moreover, it is reasonable to imagine that for a realistic network topology the number of uncertain parameters, d , might easily reach order hundreds of parameters, thus preventing the efficient use of other UQ techniques like spectral methods, *i.e.* Polynomial Chaos expansions (PC) [27]. Given the prohibitive computational cost required for each network simulation, which limits the maximum affordable number N , in order to decrease the RMSE of the estimator the only viable solution is to change the problem in a way that reduces $\text{Var}^{\frac{1}{2}}(Q)$ while keeping the value of $\mathbb{E}[Q]$ unaltered. It is important to note that, whenever a computationally cheaper evaluation of Q might be obtained without sacrificing the overall numerical accuracy, this possibility should be considered. In this work, every model introduced to alleviate the computational burden is assumed to introduce a non-negligible bias with respect to the target network system (which in this work is considered the truth system).

The pivotal idea of the multifidelity sampling strategies is the following. A small set of evaluations of the high-fidelity system is used to guarantee the convergence of the estimator to its statistics; in addition to this set, a larger number of evaluations from inaccurate but more computationally efficient systems (*e.g.* ns-3 network simulations as opposed to high-fidelity minimega emulations) is aggregated with the high-fidelity set in order to obtain an estimator with the lowest variance given a prescribed computational budget. The so-called optimal control variate (OCV) method can be used for this scope [24, 26, 25]. In the OCV estimator, a MC estimator based on N high-fidelity evaluation, $\hat{Q}_N^{\text{HF,MC}}$, is extended to include weighted sums of contributions based on M lower-fidelity models for which we consider their expected value to be known *a priori*

$$\hat{Q}^{\text{OCV}} = \hat{Q}_N^{\text{HF,MC}} + \sum_{i=1}^M \alpha_i \left(\hat{Q}_i - \mu_i \right), \quad (3)$$

where \hat{Q}_i and μ_i represent a MC estimator and the exact mean of the i th low-fidelity model, respectively and the weights $\alpha = [\alpha_1, \dots, \alpha_M]^T \in \mathbb{R}^M$ are introduced as optimization parameters. For simplicity and without loss of generality, the number of the N_i evaluations of the i th low-fidelity model is assumed proportional to the number of high-fidelity simulation N through a coefficient r_i , *i.e.* $N_i = \lceil r_i N \rceil$. By means of simple manipulations it is possible to show that such estimator is unbiased, *i.e.* $\mathbb{E}[\hat{Q}^{\text{OCV}}] = \mathbb{E}[\hat{Q}_N^{\text{HF,MC}}] = \mathbb{E}[Q]$ for any choice of the vector α . Under this framework, once the covariance matrix $C \in \mathbb{R}^{M \times M}$ amongst Q_i and the vector of covariances c between Q and each Q_i are defined, the optimal weights α^* are obtained as

$$\alpha^* = \underset{\alpha}{\text{argmin}} \text{Var} \left[\hat{Q}^{\text{OCV}} \right] = -C^{-1}c, \quad (4)$$

and the corresponding variance is

$$\text{Var} \left[\hat{Q}^{\text{OCV}} \right] = \frac{\text{Var} [Q]}{N} \left(1 - \frac{c^T C^{-1} c}{\text{Var} [Q]} \right) = \frac{\text{Var} [Q]}{N} (1 - R_{\text{OCV}}^2). \quad (5)$$

It is evident that $R_{\text{OCV}}^2 = \frac{c^T C^{-1} c}{\text{Var} [Q]}$ represents a positive quantity and $0 \leq R_{\text{OCV}}^2 \leq 1$, therefore the variance of the OCV estimator is always lesser or equal than the corresponding MC variance (based on high-fidelity realizations only). It is also important to note that if the OCV estimator is obtained as an extension of a MC estimator based on N high-fidelity simulations by

adding N_i low-fidelity simulations for $i = 1, \dots, M$, its overall cost would naturally be higher than MC. An optimal sample allocation between models is in general needed in order to obtain an efficient OCV estimator given a prescribed computational cost. Although the OCV method provides an elegant mathematical solution to decrease the RMSE of a plain MC estimator, in a practical computational settings it is necessary to estimate the values of μ_i which are unknown at the beginning of the computations, *e.g.* in this work it is not even known a priori the expected value of the ns-3 QoI. In order to address this limitation, it is possible to partition the set of low-fidelity evaluation in two (possibly overlapping) subsets and using each of them to compute the term \hat{Q}_i and an approximation of μ_i , $\hat{\mu}_i$, respectively. Interesting properties and analogy between this approach and other multifidelity approaches discussed in literature can be drawn for this framework, called Approximate Control Variate [17], however this is beyond the scope of the present work. We only note here that for particular choices of the low-fidelity simulations partitioning, it is possible to show that these estimators might exhibit an higher variance reduction than an OCV estimator with only one low-fidelity model, OCV-1 (although the final variance of the estimator would ultimately depend on the possibility to approach the theoretical variance reduction without incurring in a overwhelming low-fidelity cost). On the contrary, this possibility is prevented in more classical recursive schemes for which it is possible to demonstrate that the variance reduction is lesser than the one corresponding to OCV-1 [17].

In the present work, the goal is to demonstrate that is indeed possible to use the multifidelity sampling idea in the context of network simulations, therefore the extension to the most efficient partitioning scheme of the low-fidelity evaluations N_i is left for a future work. Given this narrower focus, here the case of a single low-fidelity model is explicitly addressed. For the case of a single low-fidelity model two possible choices of partitioning for the low-fidelity simulations are available. The set can be split in both overlapping or independent sets of simulations (by construction we assume that the cardinality of the set adopted to evaluate $\hat{\mu}_i$ is larger than the one corresponding to the set used for \hat{Q}_i). In both cases, the performances of the estimator (in term of its variance) are equivalent (the difference is limited to a dissimilar value for the optimal coefficient α_1), therefore in this work the case of fully overlapping partitioning is considered. Under these assumptions the ACV-1 estimator is equivalent to the multifidelity Monte Carlo (MFMC) estimator adopted in [33, 30, 34], *i.e.* the term \hat{Q}_i is computed by means of N evaluations (shared with the HF model) whereas the approximation $\hat{\mu}_i$ is evaluated by adding another set of $N_1 - N = (r_1 - 1)N$ independent evaluations. The final form of the estimator is

$$\hat{Q}^{\text{ACV-1}} = \frac{1}{N} \sum_{i=1}^N Q^{(i)} + \alpha_1 \left(\frac{1}{N} \sum_{i=1}^N Q_1^{(i)} - \frac{1}{N_1} \left(\sum_{i=1}^N Q_1^{(i)} + \sum_{j=1}^{N_1-N} Q_1^{(j)} \right) \right). \quad (6)$$

Simple manipulations lead to an optimal coefficient selection where

$$\begin{aligned} \alpha_1^* &= -C^{-1}c = -(\text{Var}[Q_1])^{-1} \left(\rho_1 \text{Var}^{\frac{1}{2}}(Q_1) \text{Var}^{\frac{1}{2}}(Q) \right) \\ &= -\rho_1 \frac{\text{Var}^{\frac{1}{2}}(Q)}{\text{Var}^{\frac{1}{2}}(Q_1)}, \end{aligned} \quad (7)$$

where ρ_1 denotes Pearson's correlation coefficient. This coefficient choice corresponds to a minimum variance for the multifidelity estimator (ACV-1) equal to

$$\text{Var} \left[\hat{Q}^{\text{ACV-1}} \right] = \frac{\text{Var}[Q]}{N} \left(1 - \frac{r_1 - 1}{r_1} \rho_1^2 \right). \quad (8)$$

It is important to note that, for the case of one single low-fidelity model, the OCV estimator reduces to OCV-1 and its variance reduction term is $R_{\text{OCV-1}}^2 = \rho_1^2$, thus the factor $\frac{r_1-1}{r_1} < 1$ stems from the need for estimating $\hat{\mu}_1$ in the ACV setting. The optimal sample allocation for the generic ACV estimator can be obtained in closed form only in the case of a single low-fidelity model, and again corresponds to the solution previously discussed in literature [33, 30, 34]: the optimal number of low-fidelity simulations to obtain a prescribed variance for the estimator, *i.e.* $\text{Var} [\hat{Q}^{\text{ACV-1}}] = \varepsilon^2$, corresponds to a value of r_1 equal to

$$r_1^* = \sqrt{\frac{\mathcal{C}}{\mathcal{C}_1} \frac{\rho_1^2}{1 - \rho_1^2}}, \quad (9)$$

where \mathcal{C} and \mathcal{C}_1 corresponds to a measure of the computational cost (for instance the runtime of a simulation) for the high-fidelity and low-fidelity model, respectively. For particular choices of the low-fidelity partitioning that are based on imposing a recursive sampling scheme as noted in [17], a solution in closed form can be obtained also for $M > 1$ and this case is the MFMC introduced in [34], however in this latter case the variance reduction would always be $R_{\text{MFMC}}^2 < \rho_1^2$. The corresponding number of required high-fidelity simulations to obtain a variance equal to ε^2 is obtained as

$$N^* = \frac{\text{Var} [Q]}{\varepsilon^2} \left(1 - \frac{r_1^* - 1}{r_1^*} \rho_1^2 \right) = \frac{\text{Var} [Q]}{\varepsilon^2} \Lambda(r^*, \rho_1^2), \quad (10)$$

where the function $\Lambda = \Lambda(r, \rho^2) = \left(1 - \frac{r-1}{r} \rho^2 \right)$ is introduced for compactness. The previous equation is also useful to quantify the computational cost reduction that might be obtained through the ACV-1 estimator. A MC estimator based on N_{MC} would have a variance equal to $\text{Var} [Q] / N_{\text{MC}}$, therefore for obtaining an ACV-1 estimator with equivalent variance the total number of high-fidelity simulations would be equal to $N_{\text{MC}} \Lambda(r^*, \rho_1^2)$ and its total cost

$$\mathcal{C}_{\text{tot}} = N^* \left(1 + \frac{\mathcal{C}_1}{\mathcal{C}} r^* \right) = N_{\text{MC}} \Lambda(r^*, \rho_1^2) \left(1 + \frac{\mathcal{C}_1}{\mathcal{C}} r^* \right). \quad (11)$$

The ACV-1 estimator would be more efficient as the product $\Lambda(r^*, \rho_1^2) \left(1 + \frac{\mathcal{C}_1}{\mathcal{C}} r^* \right)$ decreases. It should be noted that this term depends only on the efficiency of the low-fidelity model, *i.e.* $\frac{\mathcal{C}_1}{\mathcal{C}}$, and its correlation with the high-fidelity model, *i.e.* ρ_1^2 . To summarize, in a practical setting a multifidelity estimator as ACV-1 might be used to obtain slightly different objectives. Given a target accuracy for the estimator, the computational burden can be optimally distributed between the high- and low-fidelity model to guarantee that only the minimum possible computational cost is required. Alternatively, given a MC estimator based on high-fidelity simulations, an additional set of low-fidelity evaluations can be added to obtain the most efficient estimator given the computational effort invested in the high-fidelity realizations and the characteristics of the low-fidelity model, *i.e.* computational efficiency and correlation.

4 NUMERICAL EXAMPLES

In this section two numerical tests are conducted. A simple network topology consisting of a server and a client is studied under different operative conditions (see below for details). Two multifidelity test cases are considered in the following. In the first test, both high- and low-fidelity models are defined in the ns-3 network simulator. This test case has also served

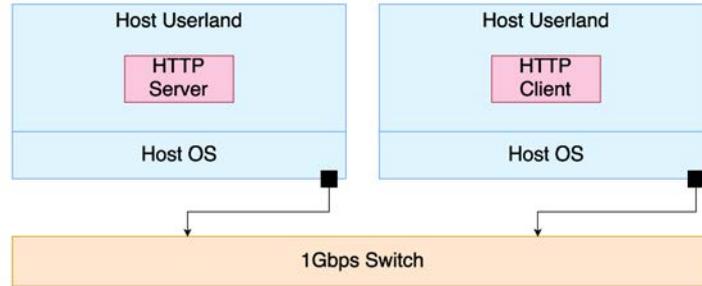


Figure 1: Simple network configuration used in this work for testing the multifidelity UQ approaches.

to test the coupling between `ns-3` and the Sandia National Laboratories' UQ software Dakota [1, 2]. In contrast, for the second demonstration problem, the high-fidelity model is defined in `minimega` whereas the low-fidelity model is based on `ns-3`. For both test cases, the performance of two possible low-fidelity models are considered to provide a preliminary indication about the achievable trade-off between correlation and computational efficiency. For the `minimega/ns-3` case, the use of both low-fidelity models at the same time is also considered as an exploratory investigation of the efficiency of the OCV strategy (which uses more than one low-fidelity model) compared to OCV-1 (which is based on a single low-fidelity model).

4.1 Experiment workload

For our numerical examples, we study a simple network topology consisting of a server and a client as depicted in Figure 1. The topology consists of two endpoints, one of which runs an HTTP server and the other of which runs an HTTP client. We will attempt to model the interactions between the client and server in this topology using both simulation and emulation.

The primary QoI in our scenario is the number of requests the client completes per second. We use multifidelity UQ to study the affects of several uncertain parameters such as the size of the HTTP response (*ResponseSize*), the delay introduced by the switch (*Delay*), and the speed of the switch (*DataRate*).

For the `minimega` emulation, we leveraged the models that we constructed during previous work [5]. Since this is an exploratory study, we do not attempt to vary *design* parameters such as the virtual network interface type as done in the previous work. In this work, we use `e1000` network drivers and 1 virtual CPU.

For the `ns-3` simulations, we created a topology to match Figure 1. We modified the built-in HTTP server and client implementations to better match the behaviors of the HTTP server and client used in the emulation (`protonuke`, a traffic generation tool in the `minimega` [28] toolset, and `ApacheBench`, a server benchmarking tool, respectively). Specifically, we modified the built-in simulated client to close and re-establish TCP connections after each request. We made this modification because the keep-alive behavior has been shown to have significant effects on HTTP performance [18]. In future work, we could explore how the correlation between the high- and low-fidelity models changes based on this modification. `ApacheBench` also supports keep-alives, creating yet another possible experiment. To match the emulation, we also parameterized the number of requests to perform and the response size in `ns-3`.

4.2 A simplified network topology in ns-3

The first demonstration is based on ns-3 simulations for both the high- and low-fidelity models. The goal of the UQ analysis is to quantify the expected value for the number of requests per second in a scenario in which a total of 100 requests are exchanged between client and server with a payload of 16MB. The case of two uncertain network parameters is considered: the *DataRate* is considered uniformly distributed between 5 and 500 Megabits per second (Mbps), whereas the *Delay* is uniformly distributed between 1 and 3 milliseconds. The uncertain parameters and their distributions are reported in Table 1.

Uncertain variable	Distribution
<i>DataRate</i>	$\mathcal{U}(5, 500)$ Mbps
<i>Delay</i>	$\mathcal{U}(1, 3)$ ms

Table 1: Uncertain parameters and their distribution for the first demonstration case.

Two low-fidelity models are considered for this test case. The first low-fidelity model is obtained by reducing the payload from 16MB to 1MB, this model is dubbed simply LF. The second low-fidelity model is generated by both reducing the payload from 16MB to only 500B and the number of requests from 100 to 10 in an attempt to obtain a very fast simulation; this latter model is named LF*. The computational runtime for the three models and their computational cost normalized with respect to the high-fidelity model (HF) are reported in Table 2.

Model	runtime [s]	Normalized Cost
HF	1200	1
LF	50	0.0417
LF*	0.15	0.000125

Table 2: Runtime and computational cost for the models used in the first demonstration.

The responses of the three models are shown in Figure 2 for reference.

As a first result, a total of 700 high-fidelity simulations has been obtained for the high-fidelity model. Afterwards, a subset of the high-fidelity simulations has been extracted and paired with an equivalent number of low-fidelity simulations in order to estimate the correlation. Once the correlation between the high- and low-fidelity model has been evaluated, the optimal number of low-fidelity realizations has been computed by resorting to Eq. (9) and the $(r_1 - 1)N$ additional number of independent low-fidelity evaluations has been obtained. The total cost of the estimator, expressed in term of equivalent HF network simulations, is evaluated by resorting to Eq. (11) and the corresponding variance is computed with Eq. (8). In Figure 3a the value of the standard deviation of the ACV-1 estimator is reported with respect to the equivalent computational cost. Note that the convergence of all the estimators is roughly order $N^{-1/2}$, whereas their constant reflects the reduced variance achieved by introducing the low-fidelity evaluations as control variate.

From a practical standpoint, a reduced variance/standard deviation corresponds to a tighter confidence interval for the estimation of the expected value of the QoI. In order to demonstrate

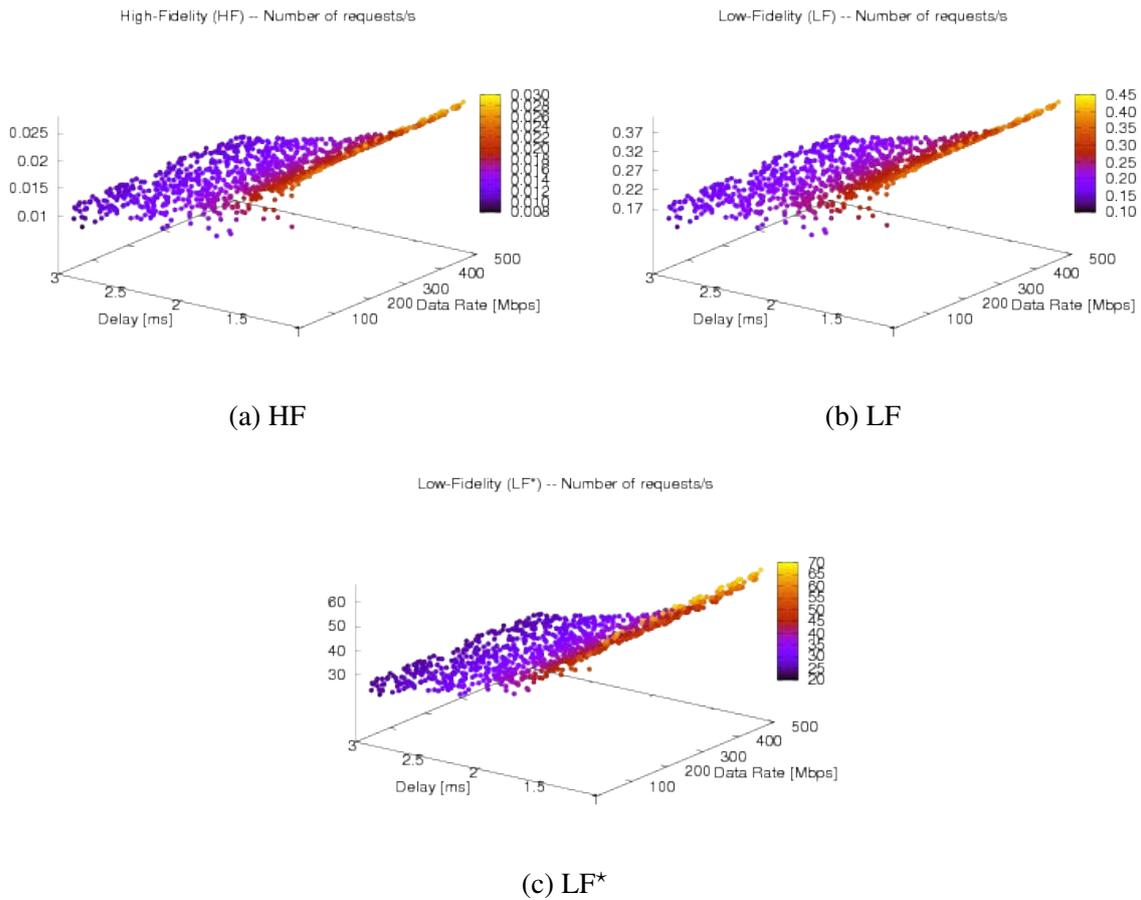
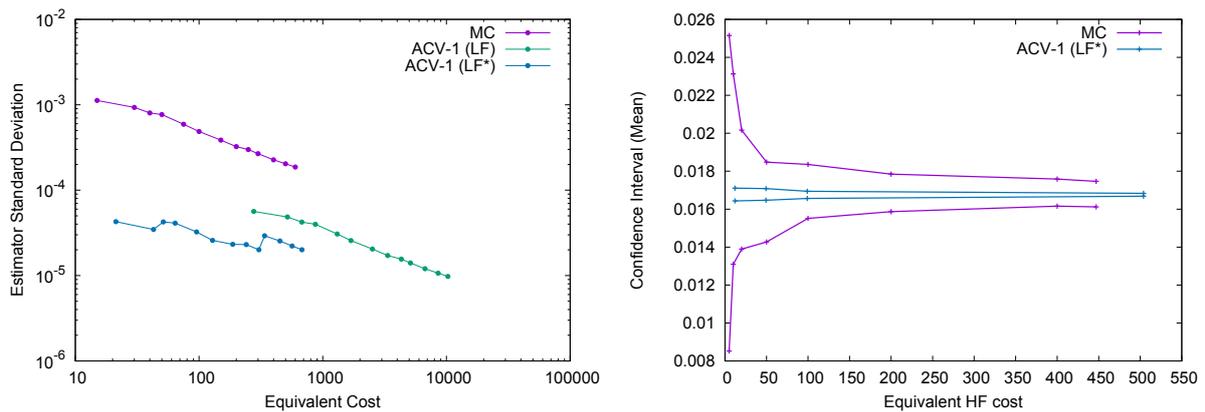


Figure 2: Responses for the three models of the first demonstration case. The qualitative behavior is similar for the three cases, however the values of requests/s predicted by the two low-fidelity model is much higher than the HF model.



(a) Estimator Standard Deviation for the first test case. (b) 99.7% confidence interval for the estimator value.

Figure 3: Estimator standard deviation for the simple MC and two variants of the ACV-1 estimator (a). The 99.7% confidence interval for the MC and ACV-1 (LF*) estimator values (b). In both figures, the QoI is the number of requests/s for which the expected value is desired.

this, Dakota has been coupled with `ns-3` and several estimator evaluations have been obtained by targeting several estimator variances. The values of the 99.7% confidence intervals for MC and ACV-1 based on `LF*` are reported in Figure 3b. It is possible to observe that the ACV-1 estimator produces much more reliable estimations for the expected value of the QoI for a very limited computational cost: a very tight confidence interval can be obtained with a computational cost that corresponds to approximately only 100 HF network simulations. A comparable confidence interval cannot be obtained with even 450 HF simulations when using a plain single fidelity MC estimator.

4.3 Extension to `minimega/ns-3` multifidelity analysis

The second experiment has a more realistic flavor and consists in the analysis of the same network configuration presented in Figure 1 by means of emulations based on `minimega`. Therefore, in this scenario `minimega` represents the unbiased high-fidelity model. The computational cost and resources needed to pursue a UQ study based of a network emulation model is generally prohibitive, thus a `ns-3` simulation model is introduced as low-fidelity model. The goal of the UQ analysis is to compute the expected value of the number of requests per seconds for an operative conditions in which 100 requests are exchanged between server and client. For this test case, two uncertain parameters are considered. Consistently with the previous example the *DataRate* has been considered uniformly distributed between 5 and 500 Mbps. The second uncertain parameter has been chosen to be the payload, *i.e.* *ResponseSize* which is assumed to be log-uniformly distributed between 500B and 16MB. The uncertain parameters are reported in Table 3.

One of the main difference with respect to the previous test case is that the simulations in `minimega` are intrinsically stochastic, *i.e.* distinct repetitions of the same network configuration are expected to produce slightly different results. This is a product of emulation being subject to real-world timing in the virtual (and underlying physical) hardware and not running off a simulated clock. For this simple configuration it has been observed that a limited number of repetitions, of the order of 10 repetitions, was sufficient to characterize in average the response of a system for a fixed set of uncertain parameters. Complex network configurations might require the adoption of more sophisticated techniques to control the overall error induced on the statistics by the variability in `minimega`, however this is beyond the scope of the exploratory study conducted here and it is left for subsequent studies.

Uncertain variable	Disribution
<i>DataRate</i>	$\mathcal{U}(5, 500)\text{Mbps}$
<i>ResponseSize</i>	$\ln\mathcal{U}(500, 16 \times 10^6)\text{B}$

Table 3: Uncertain parameters and their distribution for the second demonstration.

Two low-fidelity models are defined by using `ns-3`. The first low-fidelity model has been obtained by reducing the number of requests from 100 to 10. Additionally, the parameter *Delay*, which does not have a counterpart in `minimega`, has been chosen as 50ms by observing its impact on the response. In the future, in the presence of more complex network configurations and a large set of parameters, a formal calibration process might be performed. Hereinafter, this low-fidelity model is referred as LF. The second low-fidelity model has been obtained by reducing the number of requests to the extreme case of a single request and the parameter *Delay*

has been fixed at the value of 5ms. The runtime and the normalized computational cost for the three models used in this numerical experiment are reported in Table 4.

Model	runtime [s]	Normalized Cost
HF	2680	1
LF	42.88	0.016
LF*	5.36	0.002

Table 4: Runtime and computational cost for the models used in the second demonstration.

It is important to note that in the following numerical experiments the computational cost is measured in terms of equivalent runtime for a serial execution. This is not necessarily the case when the low-fidelity simulations (as `ns-3` in this case) might be potentially evaluated in parallel. In this latter case, the LF computational cost normalized by the HF cost would have been smaller (*i.e.* more efficient LF model) than the normalized cost reported in Table 4. Nonetheless, since the serial execution is expected to provide the worst case scenario, this is the chosen metric for the performance comparison in the following. Another advantage stemming from this choice is that the results might be seen as hardware independent, in contrast to the parallel execution scenario in which the results would be only relative to the particular configuration adopted, *i.e.* the number of parallel threads available.

The responses of the three models for the second test case are also reported in Figure 4 for reference. A total of 500 network emulations has been obtained for `minimega` and several realizations of a MC estimator have been obtained for an increasing number of simulations. From the set of 500 HF runs, a sequence of subsets with increasing number of runs, has been extracted to serve as a basis for the ACV-1 estimators. These subsets are first used to compute corresponding LF simulations and their correlation with the HF. Afterward, the oversampling ratio r_1 is estimated from Eq. (9) and the corresponding set of (additional) LF runs is evaluated in `ns-3`. Finally, the ACV-1 estimator is evaluated by resorting to Eq. (6). In Figure 5a the performance of the different estimators are reported in term of their standard deviation. The expected rate of convergence for all the sampling estimators, $\mathcal{O}(N^{-1/2})$ is also observed.

The 99.7% confidence interval on the expected value for the number of requests per second is also reported in Figure 5b to demonstrate the increased reliability of the MF estimators. The ACV-1 estimator based on the LF* model exhibits the highest performance. However, the ACV-1 estimators based on both LF show similar performance and they can be clearly seen as more efficient than the plain MC estimator based on high-fidelity evaluations only.

4.4 Exploring the potential of including multiple low-fidelity models

In order to explore the possibility to obtain an additional variance reduction by introducing more than one low-fidelity model, the performance of the OCV estimator (which assumes the low-fidelity statistics known) based on the simultaneous use of LF and LF* is compared to the OCV-1 estimator (where one single low-fidelity model is used). These results are meant to serve only as an indication of the potentiality of an ACV estimator (in which multiple LF models are used simultaneously but their expected values are unknown) as described in [17] because the final performance of the algorithm would need to include the cost of the low-fidelity models.

First, the correlation matrix for this test case is reported in Table 5. Both low-fidelity models are very well correlated with the high-fidelity model, which is an indication that the multifidelity

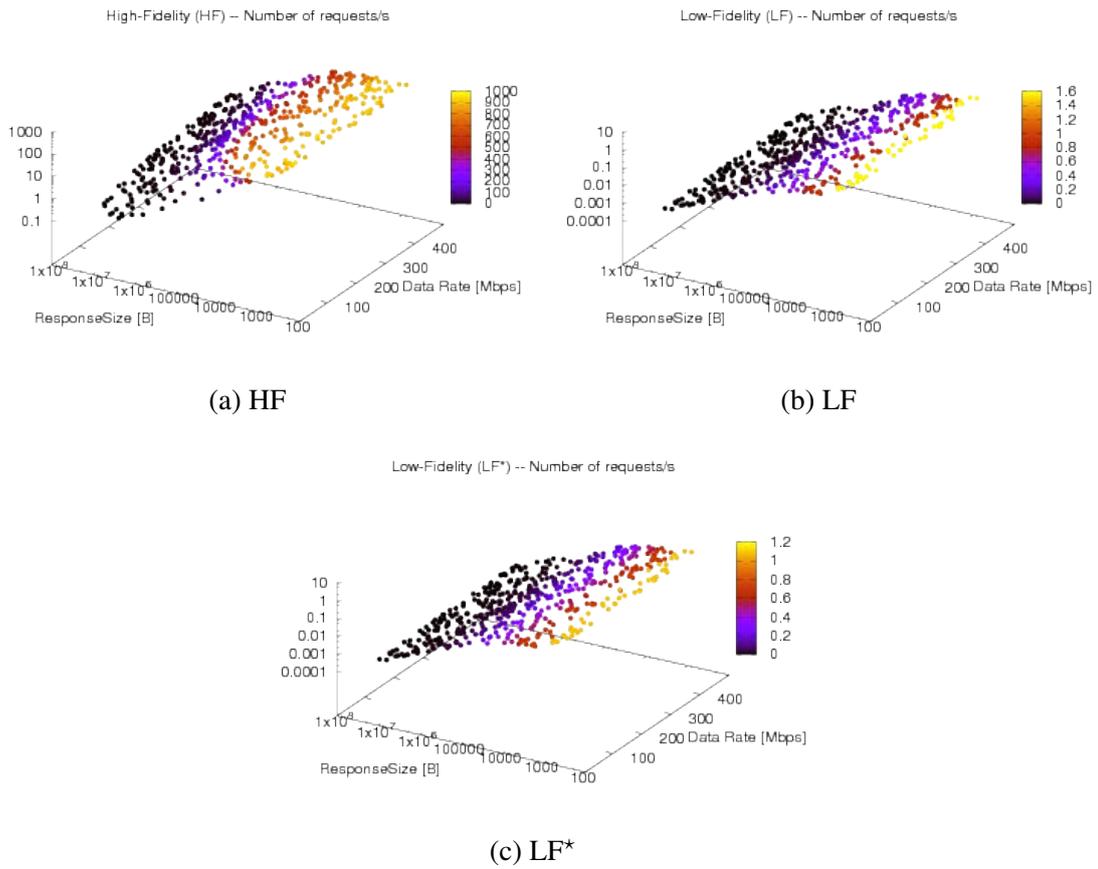
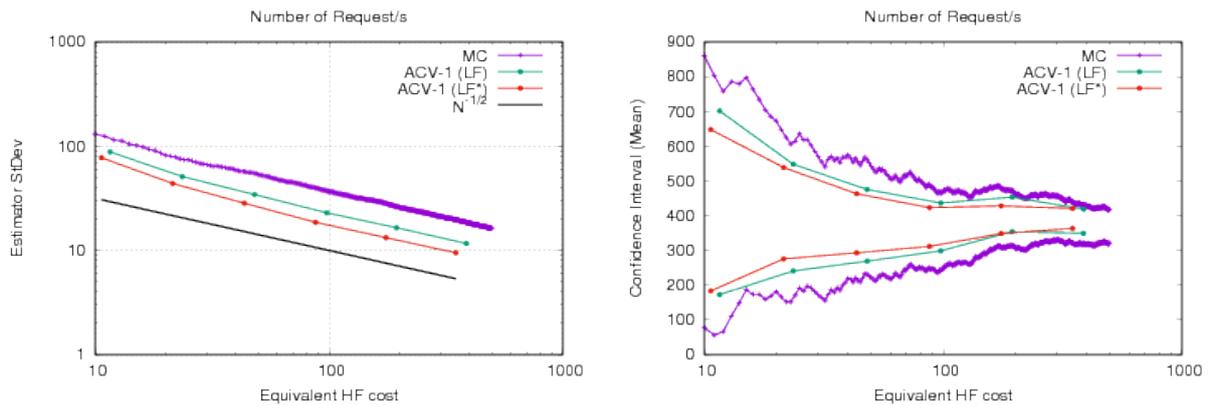


Figure 4: Responses for the three models of the second demonstration case. For this case the two low-fidelity models ($ns-3$) are very similar between them, whereas the HF model minimega exhibits a much higher number of request/s.



(a) Estimator Standard Deviation for the first test case. (b) 99.7% confidence interval for the estimator value.

Figure 5: Estimator standard deviation for the simple MC and two variants of the ACV-1 estimator (a). The 99.7% confidence interval for MC and ACV-1 based on both the low-fidelity estimators are reported (b). In both figures, the QoI is the number of requests/s for which the expected value is desired.

	HF	LF	LF*
HF	1	0.86	0.90
LF	0.86	1	0.99
LF*	0.90	0.99	1

Table 5: Correlation matrix for the models used in the second test cases.

Estimator (low-fidelity models)	$\text{Var} \left[\hat{Q}_N^{\text{OCV}} \right] / \text{Var} \left[\hat{Q}_N^{\text{MC}} \right]$	$\text{Var} \left[\hat{Q}_N^{\text{ACV}} \right] / \text{Var} \left[\hat{Q}_N^{\text{MC}} \right]$
Multifidelity (HF-LF)	0.26	0.39
Multifidelity (HF-LF*)	0.19	0.23
Multifidelity (HF-LF-LF*)	0.08	N/A

Table 6: Variance Reduction obtained by several estimators based on the three models HF, LF and LF* for the second test case.

estimator might be very effective. Moreover, the two low-fidelity models are almost perfectly correlated between them.

In Table 6, the three multifidelity estimators are reported in term of their normalized variance, *i.e.* the ratio between their variance and the one for a plain MC estimator with the same number of HF simulations. In the first column the normalized variance for the case of known low-fidelity statistics (OCV) is reported. The use of both LF models simultaneously achieves the greatest variance reduction exhibiting only 8% of the variance of the corresponding MC estimator. The estimation of the LF statistics, as explained in Section 3, reduces the effectiveness of the OCV estimators as can be observed in the second column of Table 6 where for the ACV-1 estimator the normalized variance is reported. In general, the ACV estimator based on multiple LF models requires the specification of the LF partitioning scheme (see [17]) and a numerical optimization to obtain the sample allocation in closed form. The accurate quantification of the performance of this estimator are left for a future study, however it is promising to observe a variance reduction gap between OCV-1 and OCV which might translate to a similar gap between ACV-1 and ACV.

5 CONCLUDING REMARKS

In this work, multifidelity uncertainty quantification has been performed for network applications. Two approaches have been considered for the network computations: a simulation approach based on the network simulator `ns-3` and the network emulator `minimega`. The UQ tool of choice has been a multifidelity sampling approach based on a control variate which is capable of maximizing the variance reduction whenever multiple low-fidelity models are available. A simple network configuration consisting of a server and a client has been configured and two possible test cases have been addressed. The first case is a simulation only case where both the high- and low-fidelity model are evaluated in `ns-3`. The second case is more realistic and based on `minimega` as high-fidelity model and `ns-3` as low-fidelity one. For both test cases, the multifidelity sampling approach has been demonstrated to be more efficient than a plain MC estimator. Albeit the results obtained in this work are promising they would need to be verified for more complex network configurations where the topology exhibits a higher degree of complexity and the number of uncertain parameters is much larger. Additional care would also need to be devoted to the representation of discrete variables which are very natural when dealing with networks and strategies to automatically create lower fidelity models given a

particular (possibly large) network topology. Future work will also focus on understanding and mitigating the degradation of the correlation amongst network models in the presence of dissimilar input parametrizations following what has been done for computational science models in [13, 12].

6 ACKNOWLEDGEMENTS

This work has been supported by the SECURE project within the Grand Challenge LDRD Program at the Sandia National Laboratories. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energys National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

REFERENCES

- [1] B. M. Adams, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, J. R. Frye, G. Geraci, R. W. Hooper, P. D. Hough, K. T. Hu, J. D. Jakeman, M. Khalil, K. A. Maupin, J. A. Monschke, E. M. Ridgway, A. Rushdi, J. A. Stephens, L. P. Swiler, J. G. Winokur, D. M. Vigil, and T. M. Wildey. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.7 theory manual. Technical Report SAND2014-4253, Sandia National Laboratories, Albuquerque, NM, Updated November 2018. Available online from <http://dakota.sandia.gov/documentation.html>.
- [2] B. M. Adams, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, J. R. Frye, G. Geraci, R. W. Hooper, P. D. Hough, K. T. Hu, J. D. Jakeman, M. Khalil, K. A. Maupin, J. A. Monschke, E. M. Ridgway, A. Rushdi, J. A. Stephens, L. P. Swiler, J. G. Winokur, D. M. Vigil, and T. M. Wildey. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.7 users manual. Technical Report SAND2014-4633, Sandia National Laboratories, Albuquerque, NM, Updated November 2018. Available online from <http://dakota.sandia.gov/documentation.html>.
- [3] F. Bellard. Qemu, a fast and portable dynamic translator. In *USENIX Annual Technical Conference, FREENIX Track*, volume 41, page 46, 2005.
- [4] P. G. Constantine. *Active subspaces: Emerging ideas for dimension reduction in parameter studies*, volume 2. SIAM, 2015.
- [5] J. Crussell, T. M. Kroeger, A. Brown, and C. Phillips. Virtually the same: Comparing physical and virtual testbeds. In *2019 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2019.
- [6] A. Dienstfrey and R. e. Boisvert. *Uncertainty Quantification in Scientific Computing*. 10th IFIP WG 2.5 Working Conference, WoCoUQ2011. Springer, 2012.

- [7] H. Fairbanks, A. Doostan, C. Ketelsen, and G. Iaccarino. A low-rank control variate for multilevel monte carlo simulation of high-dimensional uncertain systems. *Journal of Computational Physics*, 341:121–139, 2017.
- [8] J. Farooq and T. Turetti. An ieee 802.16 wimax module for the ns-3 simulator. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, page 8. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.
- [9] C. M. Fleeter, G. Geraci, D. E. Schiavazzi, A. M. Kahn, M. S. Eldred, and A. L. Marsden. Multilevel multifidelity approaches for cardiovascular flow under uncertainty. In *Sandia Center for Computing Research Summer Proceedings 2017, A.D. Baczewski and M.L. Parks, eds.*, volume Technical Report SAND2018-27800, pages 27–50. Sandia National Laboratories, 2018.
- [10] L. Foundation. Open vswitch, 2019.
- [11] G. Geraci, G. Iaccarino, and M. S. Eldred. A multi fidelity control variate approach for the multilevel monte carlo technique. *CTR Annual Research Briefs 2015*, pages 169–181, 2015.
- [12] G. Geraci and M. S. Eldred. Leveraging intrinsic principal directions for multifidelity uncertainty quantification. In *Technical Report SAND2018-10817*. Sandia National Laboratories, 2018.
- [13] G. Geraci, M. S. Eldred, A. A. Gorodetsky, and J. D. Jakeman. Leveraging active directions for efficient multifidelity uq. In *Proceedings of the 7th European Conference on Computational Fluid Dynamics (ECFD 7)*, pages 2735–2746, 2018.
- [14] G. Geraci, M. S. Eldred, and G. Iaccarino. A multifidelity multilevel monte carlo method for uncertainty propagation in aerospace applications. In *19th AIAA Non-Deterministic Approaches Conference*, page 1951, 2017.
- [15] M. B. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3):607–617, 2008.
- [16] M. B. Giles. Multilevel Monte Carlo methods. *Acta Numerica*, 24:259–328, 2015.
- [17] A. A. Gorodetsky, G. Geraci, M. Eldred, and J. D. Jakeman. A generalized framework for approximate control variates. *arXiv preprint arXiv:1811.04988v2*, 2018.
- [18] D. Gourley, B. Totty, M. Sayer, A. Aggarwal, and S. Reddy. *HTTP: the definitive guide.* ” O’Reilly Media, Inc.”, 2002.
- [19] A.-L. Haji-Ali, F. Nobile, and R. Tempone. Multi-index Monte Carlo: when sparsity meets sampling. *Numerische Mathematik*, 132(4):767–806, Apr 2016.
- [20] J. Helton, C. Hansen, and P. e. Swift. Performance assessment for the proposed high-level radioactive waste repository at yucca mountain, nevada. *Reliability Engineering and System Safety, Special Issue*, pages 1–456, 2014.

- [21] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell, and J. Kopena. Network simulations with the ns-3 simulator. *SIGCOMM demonstration*, 14(14):527, 2008.
- [22] L. Jofre, G. Geraci, H. Fairbanks, A. Doostan, and G. Iaccarino. Multi-fidelity uncertainty quantification of irradiated particle-laden turbulence. In *Center for Turbulence Research Annual Research Briefs*, pages 21–34. Center for Turbulence Research, Stanford University, 2017.
- [23] A. Kivity, Y. Kamay, D. Laor, U. Lublin, and A. Liguori. kvm: the linux virtual machine monitor. In *Proceedings of the Linux symposium*, volume 1, pages 225–230. Dttawa, Dntorio, Canada, 2007.
- [24] S. Lavenberg, T. Moeller, and P. Welch. *Statistical results on multiple control variables with application to variance reduction in queueing network simulation*. IBM Thomas J. Watson Research Division, 1978.
- [25] S. S. Lavenberg, T. L. Moeller, and P. D. Welch. Statistical results on control variables with application to queueing network simulation. *Operations Research*, 30(1):182–202, 1982.
- [26] S. S. Lavenberg and P. D. Welch. A perspective on the use of control variables to increase the efficiency of monte carlo simulations. *Management Science*, 27(3):322–335, 1981.
- [27] O. Le Maitre and O. M. Knio. *Spectral Methods for Uncertainty Quantification with Applications to Computational Fluid Dynamics*. Springer Netherlands, 2010.
- [28] minimega developers. minimega: a distributed vm management tool, 2019.
- [29] M. G. Morgan and M. Henrion. *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*. Cambridge University Press, 1990.
- [30] L. Ng and K. Willcox. Multifidelity approaches for optimization under uncertainty. *Int. J. Numer. Meth. Engng.*, 100(10):746–772, 2014.
- [31] B. Nguyen, A. Banerjee, V. Gopalakrishnan, S. Kasera, S. Lee, A. Shaikh, and J. Van der Merwe. Towards understanding tcp performance on lte/epc mobile networks. In *Proceedings of the 4th workshop on All things cellular: operations, applications, & challenges*, pages 41–46. ACM, 2014.
- [32] W. L. Oberkampf and C. J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, 2010.
- [33] R. Pasupathy, B. W. Schmeiser, M. R. Taaffe, and J. Wang. Control-variate estimation using estimated control means. *IIE Transactions*, 44(5):381–385, 2012.
- [34] B. Peherstorfer, K. Willcox, and M. Gunzburger. Optimal model management for multifidelity Monte Carlo estimation. *SIAM Journal on Scientific Computing*, 38(5):A3163–A3194, 2016.
- [35] K. Rabieh, M. M. Mahmoud, K. Akkaya, and S. Tonyali. Scalable certificate revocation schemes for smart grid ami networks using bloom filters. *IEEE Transactions on Dependable and Secure Computing*, 14(4):420–432, 2017.

- [36] A. Saltelli, K. Chan, and E. Scott. *Sensitivity Analysis*. John Wiley & Sons, 2000.
- [37] R. Smith. *Uncertainty Quantification: Theory, Implementation, and Applications*. Computational Science and Engineering. SIAM, 2013.
- [38] T. Stocker, D. Qin, G.-K. Plattner, M. Tignor, S. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex, and P. M. (eds.). IPCC, 2013: Climate change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change. Technical report, 2013.
- [39] F. Van den Abeele, J. Haxhibeqiri, I. Moerman, and J. Hoebeke. Scalability analysis of large-scale lorawan networks in ns-3. *IEEE Internet of Things Journal*, 4(6):2186–2198, 2017.