Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

CRITICAL INFRASTRUCTURE NETWORK ANALYSIS ENABLED BY SIMULATION METAMODELING

Scott L. Rosen
David Slater
Emmet Beeker
Samar Guharay
Garry Jacyna

The MITRE Corporation
22102 Colshire Drive
McLean, VA, 22012, USA

ABSTRACT

This paper presents an application of simulation metamodeling to improve the analysis capabilities within a decision support tool for Critical Infrastructure network evaluation. Simulation metamodeling enables timeliness of analysis, which was not achievable by the original large-scale network simulation due to long set-up times and slow run times. We show through a case study that the behavior of a large-scale simulation for Critical Infrastructure analysis can be effectively captured by Neural Network metamodels and Stochastic Kriging metamodels. Within the case study, metamodeling is integrated into the second step of a two-step analysis process for vulnerability assessment of the network. This consists first of an algorithmic exploration of a power grid network to locate the most susceptible links leading to cascading failures. These links represent the riskiest links in the network and were used by the metamodels to visualize how their failure probabilities affect global network performance measures.

1 CRITICAL INFRASTRUCTURE MODEL-BASED VUILNERABILITY ASSESSMENT

Risk analysis is a well-established systems engineering discipline that includes risk assessment and risk management as components. Risk assessment is concerned with how extrinsic and intrinsic events lead to vulnerabilities, both at the component and system levels. In the analysis of Critical Infrastructure (CI) networks, vulnerabilities are discovered through design of experiments using simulations of CI networks. All CIs of interest can be represented as a network of connected components (nodes and links). These components can be described mathematically in terms of their spatial (topological) and/or functional properties. For this study, we have examined the Power Grid where the nodes are the generators, substations, and loads. The links are the transmission lines connecting the generators to the substations and loads. Network connectivity is a topological property while power flow is a functional property. Both properties are viewed as equally important.

A Model-Based Vulnerability Assessment (MBVA), (Lewis 2006) for a Critical Infrastructure (CI) involves the application of models and decision support tools to quantify the vulnerabilities, uncertainties, and risks associated with mitigation strategies. Within this framework, CIs are decomposed into a network of connected components, critical node and links. As shown in Fig. 1, vulnerabilities are examined and quantified through the use of simulation models and translated into fault trees, where the fault probabilities are enumerated using event trees. Component faults or system failures are then examined and associated risks are assessed. A set of tradeoffs then determine the best policies and/or resources for risk mitigation based on optimal resource allocation.

978-1-4673-9743-8/15/$31.00 ©2015 IEEE

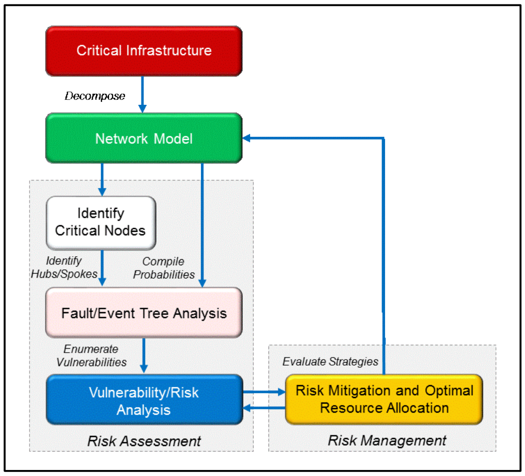


Figure 1: Model-based Vulnerability Assessment (MBVA) Components.

A key component for CI risk analysis and assessment is the identification of the costs and associated system risks as well as the optimal investment strategies required to reduce risk in the most effective manner (Lewis 2006). This step involves both risk assessment and risk management. In risk assessment, there is a determination of what can go wrong, the likelihood of something going wrong, and the consequences of something going wrong (Kaplan 1997). In risk management, various risk-mitigating strategies are considered, tradeoffs are evaluated in terms of risks, costs, impacts, and other factors, and current decisions/policies are assessed with respect to their future impacts

The methodology for CI risk analysis and assessment involves identifying and modeling interdependencies through network analysis methods, i.e., determining shared system states and the synergistic/pernicious couplings between systems. As such, the risk analysis tools are both functionally and operationally coupled to the following three logical frameworks: Model-based Network Generation and Analysis, Dynamic Model Experimental Design and Execution, and Metamodel Formulation and Execution. The Dynamic Model Experimental Design and Execution component is described in the next section of the paper. This component is utilized to develop the Metamodel Formulation and Execution component. A case study is also provided in this paper to illustrate the development and implementation of the Metamodel Formulation and Execution component.

2 DYNAMIC MODEL EXPERIMENTAL DESIGN AND EXECUTION

MBVA consists of a framework for Dynamic Model Experimental Design and Execution that provides plug-and-play support for application-specific dynamic models while providing the capability to define and execute experiments of interest based on a selected list of Design of Experiment (DOE) parameters and Measure-of-Effectiveness (MOE) model outputs. A robust visualization interface provides an intuitive method of understanding network performance within the context of associated risks, vulnerabilities, and resiliencies. Although we incorporated two dynamic power grid models: Cascading Failure Simulation (CFS) and ORNL-PSERC-Alaska (OPA) (Carreras 2004), the study presented in this paper focuses on the CFS model.

The CFS model is a slightly modified version of the cascading failure simulator from Korkali et al. (2014) which is used to model the physics of the electrical grid. An algorithm flow chart for the CFS model is shown in Fig. 2. Due to the uncertainties associated with system parameters and operator's strategy to avoid system instability, real grid dynamics are simplified with DC power-flow approximations in this simulator. A triggering event (e.g. random or targeted attack) leads to an initial failure of some power system components (substations, transmission lines, etc.) that changes the balance of the power flows within the grid. This imbalance leads to a redistribution of power flow in the grid via generator ramping or load shedding that can result in further local overloads initially and potential blackouts of various sizes. The triggering event might then cause an overload in other parts of the network; thus, the protection system trips the overloaded components, and the power flow is rerouted, possibly inducing further overloads. If an islanding (parts of the power grid becoming separated from each other) occurs, cascading failures may persist in each island in which individual generators or loads are shed in order to achieve a balance of power. The cascade of failures continues until no further components are overloaded or the power grid operation has degraded past a pre-defined level. After the cascading subsides, the robustness of the grid against failures is quantified either in terms of the fraction of the served power or the size of the largest portion of the grid that continues to operate with power.

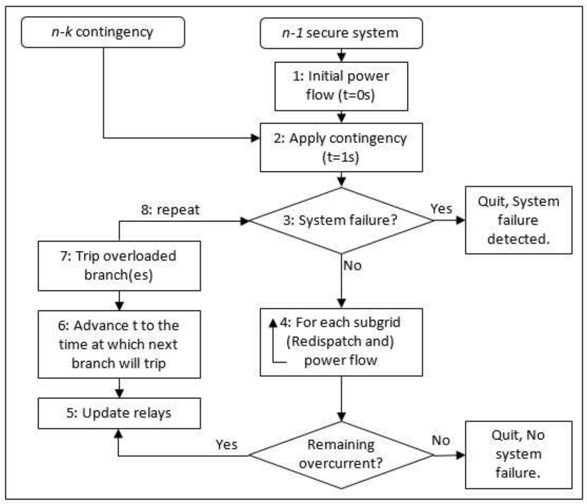


Figure 2: Algorithm Flow Diagram for the CFS Model.

The CFS model is utilized in the Dynamic Model Experimental Design and Execution framework to locate the set of blackout-causing contingencies within a power grid network: for any size $\boldsymbol{k}$ we are interested in the set of $\boldsymbol{n-k}$ contingencies where n is the number of links in the original network and k is the number of failed links. Blackout contingencies are combination sets of network links that lead to a cascading failure in the network. Not all $\boldsymbol{n-k}$ sets of links will result in a blackout. Locating the blackout-causing contingencies is a critical step in understanding overall network vulnerability and assessing and managing risk. Fewer blackout contingencies indicate greater power grid robustness and this is an objective when considering alternatives for system design.

2.1 Graph Exploration for n-k Contingencies

The biggest challenge in using metamodeling to expedite the analysis of CI networks is in identifying the blackout causing contingencies. Given that there are n branches in the network, there are $\binom{n}{k}$ possible contingencies. For example, in the Polish Power Grid, described in Section 4, there are 2896 links, so there are $\binom{n}{2}$ = 4,191,960 different possible combinations to search across for $n-2$ contingencies alone, making the location of susceptible branch combinations of size $1,2 \ldots, k$ a computationally intensive search problem. The main approach suggested in literature to locate blackout-causing contingencies is based on the Random Chemistry algorithm. See Buzas (2013) and Rezaei (2014) for more information. Random Chemistry is essentially a sequential bifurcation type of factor screening procedure that starts each trial of its search with a large set of branches (80 has been suggested in the literature). Given that this large subset produces a cascading blackout, a successive subset of half the size is randomly generated. If the reduced subset is found to result in a cascading blackout, it is reduced in half randomly again to generate another subset. If it does not produce a cascading blackout, then other randomly generated subsets are continued to be constructed until one that generates a cascading blackout is discovered or a predetermined number of subsets have been attempted without a cascading blackout result. In a Random Chemistry application from Eppstein and Hines (2012), when a $n-5$ blackout contingency is found, then all possible 2, 3, and 4 link combinations are tested for cascading blackouts. Eppstein and Hines defined a blackout contingency to occur whenever the giant component (largest connected subgraph) of the network was less than 90% of the original network size.

For the Polish Power Grid, we were able to perform an exhaustive search of all $n-2$ contingencies and found 594 contingencies that resulted in cascading blackouts, based upon Eppstein and Hines definition. There were 340 distinct links in the 594 contingencies. Some links were more critical than others; for example, link 169 occurred in 169 of the blackout contingencies. The eight most frequent links were involved with 466 of the contingencies. Because the number of blackout contingencies is small, they are difficult to find. Eppstein and Himes (2012) used 735,500 successful Random Chemistry trials, resulting in over 33 million simulation runs to find 336 of the blackout contingencies in the Polish Power Grid network. Because a few links are found in most blackout contingencies, they can be discovered with far fewer trials. We executed 300 trials and found 109 blackout contingencies. These contingencies had 91 distinct links. When we paired each of the 91 links with each of the remaining links, we found 593 blackout contingencies using 263,445 simulation runs. Only one blackout contingency had two links that were not involved with some other contingency.

2.2 Analysis of Risk due to Link Failures

Acquiring the set of $n-k$ contingencies expands the scope of analyses that can be performed on the critical infrastructure network and is the first step to risk analysis. Initially, these sets of contingencies can be examined to determine repeating links that are involved in cascading failures. Those links that are repeatedly involved in the set of cascading failures are the focal points for network vulnerability studies and, ultimately, network repair. An analyst would first want to study how the links most frequently involved in initiating cascading failures are coupled to the overall performance of the network as characterized by the number of broken links, the power loss, size of the giant component after a failure, and the number of non-working nodes. Additionally, the analyst would want to know how the failure probabilities associated with these critical links affect global network performance measures. This provides an understanding of how repairing an individual link to improve failure probability improves the overall network performance with respect to these performance measures. This analysis can support acquisition decisions germane to network repair in critical areas to determine where the most significant improvements exist within the network.

Simulation can support these decisions, but the slow set-up times and run times associated with the CFS simulation hinder deep analysis. Simulation metamodeling is introduced here as a critical step in enabling this type of analysis. Simulation metamodeling provides a means for real-time analysis and easy scenario configuration, which leads to achieving the types of analyses discussed above.

3 SIMULATION METAMODELING FOR EXPEDITED ANALYSIS

Simulation metamodeling is recognized as the key to achieving more rapid analysis capabilities inside a decision support tool for critical infrastructure network evaluation. Simulation metamodels are essentially models of simulation models that capture the input / output relationships of a large-scale simulation for the purpose of generating model observations in real-time. A simulation metamodel is constructed from sampling the simulation model at discrete points in the design space and fitting a model through the observations generated at these sample points. It can be configured as a closed-form mathematical expression, but can also be inclusive of rule-based artificial intelligence (AI) approaches.

Metamodels are best understood as mathematical functions with the relationship $y=f(x)=g(x)+\epsilon$, where y and x are scalar output and vector valued inputs to the simulation model, respectively. Moreover, consider $f(x)$ to be an implicit function representing the mapping between input parameters of the simulation model and the output performance measures of the simulation. The simulation metamodel $g(x)$ is an approximation of $f(x)$ with an error term ϵ.

There is good body of literature on the topic of simulation metamodeling; see Barton and Meckesheimer (2006) for a recent review of techniques. A few key techniques are highlighted here, which have all been applied to simulation metamodeling:

(a) Response Surface techniques (Myers 1976; Box and Draper 1987; Myers et al. 2009),
(b) Splines (Eubank 1988; Deboor 1978; Myers et al. 1996),
(c) Radial Basis Functions (Shin et al. 2002; Dyn et al. 1986; Meghabghab 2001; Hussain et al. 2002),
(d) Stochastic Kriging (Sacks et al. 1989; Kleijnen 2000; Staum 2009; Kleijnen 2009; Ankenman et al. 2010),
(e) Neural Networks (Lippman 1987; Fonseca 2003; Al-Hindi 2004), Inductive Learning (Michalski 1983), and
(f) Genetic Programming (Koza 1992).

The best metamodel technique is dependent on the characteristics of the simulation under experimentation, such as, the complexity of the response surface, the number of input parameters, and the prevalence of input parameters, which are discrete. A comprehensive definition of a metamodel technique must entail several factors including the form of the underlying basis functions, how they are integrated together, and the fitting strategy that leads to the smallest errors. But in the metamodeling literature, a metamodeling technique is often defined by one attribute, which can cause confusion. Radial basis functions serve as an example of this as they do not fully define the metamodel technique itself or fully classify it into a unique family as the basis functions can be expanded or mapped together in a variety of ways.

When addressing large scale simulations of Critical Infrastructures, basis functions of complex shapes are needed along with a flexible and complex structure for piecing together these basis functions. The most intuitive method that provides this feature is the metamodel family of Neural Networks. In our past research on applying simulation metamodeling for cases with a high input parameter space, we have observed Neural Networks to provide very high fidelity and a superior goodness-of-fit to simulations relative to other metamodel families (Rosen et al. 2014). In addition Stochastic Kriging appears to be another candidate metamodel technique that can be used to capture the very complex response surface of the Critical Infrastructure simulation. See Rosen et. al (2015) for more information on the selection and application of metamodel techniques. Both of these methods will be applied to this problem in a case study, which is described in the next section.

4 CASE STUDY: POLISH POWER GRID

In this section we demonstrate how simulation metamodeling can be used to analyze network vulnerability resulting from susceptible links within the Polish Power Grid. The motivation for applying metamodeling here is to enable this analysis capability to be contained within a real-time decision support toolkit for key decision makers. Simulation metamodeling is applied through a structured approach (Rosen et al. 2014) that guides the selection of the best metamodeling techniques to apply, an experimental design for calibration, and validation statistics. There are four network metrics of interest for global risk: power loss, number of non-working nodes, size of the giant component, and the number of broken links. The size of the giant component is a measure of power grid degradation. What has been found through previous experimentations was that the network tends to break down into subnets where most nodes are still powered. Even with a small giant component size, most loads are satisfied. Power loss and non-working nodes appear to be more important than the size of the giant component, but that is subjective to the decision maker. Total links failed is also important because it gives an indication of the cost to repair. All four performance measures can be assessed in different ways by different decision makers.

A simulation metamodel is calibrated for each expected value of each of the four network performance measures stated in the paragraph above. After the metamodels are calibrated and validated, we provide illustrations of the expedited analysis that can be performed with these metamodels. The input-output relationship of the simulation captured by metamodel consisted of mapping the failure probabilities of eight critical links in the network to the four network performance measures listed above. There was one additional simulation input, which was the initial event size or the number of failed links at the start of the simulation. The eight links selected as inputs represent links that were the most representative in the set of n-k contingencies discovered through the network search algorithm discussed in Section 2.1. The vulnerability analysis is a two-step process: locating the most malignant links through expiration on the CFS model and then developing metamodels to determine the sensitivity that these eight links have on the global network performance measures.

Based on Rosen et al. (2014) it was concluded that a Neural Network and Stochastic Kriging metamodel were the best techniques to try for this particular problem. To calibrate these metamodels, an experiment consisting of a Latin Hypercube design was used across the design component consisting of the eight failure probabilities. A 100 replications of the simulation were each performed at varying values of the initial event size: 10, 50, 100, 150, 200, 250, and 300 were used. This totaled 80,000 simulation runs. For the Neural Network metamodel, a multiple layer feed-forward Neural Network was applied as shown in Figure 3 where m is the number of hidden nodes and α and β represent the connection weights of the network. The input layer is the column of nodes on the left-hand side of the network with each node pertaining to a single input parameter of the simulation. The hidden layer is contained in the middle column of nodes, which constitute a transformation from some subset of the input nodes in the network through weight terms $\beta_{i,j}$ with the first subscript referring to the hidden layer of the Neural Network and the second subscript referring to the node within the hidden layer. Within each node in the hidden layer is a threshold transfer function $p_k = \theta_k(\beta, x, \phi)$ of sigmoid form (Lippman 1987) by a threshold ϕ. The network then maps each of the function outputs p_k to a single-valued output y, which is targeting the value of the simulation model under a vector of inputs x.

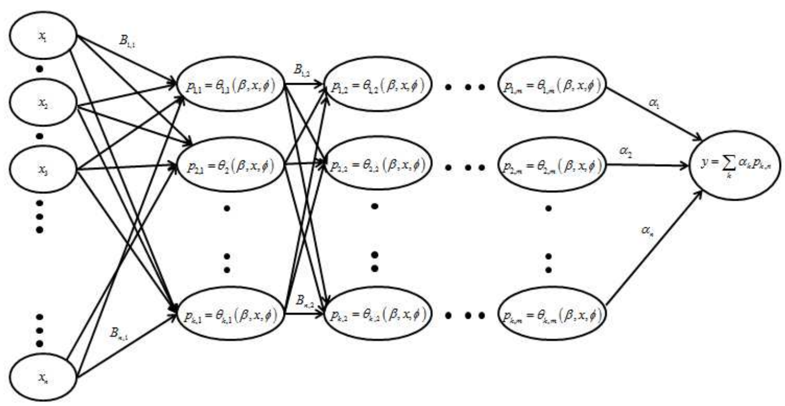


Figure 3: Neural network for simulation metamodeling with multiple layers.

In Stochastic Kriging, the metamodel assumes the simulation response surface $y(x)$ to be a realization of a Gaussian random field where the simulation outputs $Y(x) = b(x)B + M(x)$. Here the b(x)B component of the metamodel is analogous to a trend model and $M(x)$ represents a mean-zero Gaussian random field with a covariance matrix derived by spatial correlation to measure deviation from the trend $b(x)B$. The uncertainty captured by $M(x)$ is the uncertainty extrinsic to the simulation model. In Stochastic Kriging, the intrinsic uncertainty within the simulation model is also captured with the addition of an additional error term to Y(x) that is estimated through performing multiple replications with the simulation model. Standard Kriging was applied in this case study using a linear trend and structuring $M(x)$ through a Gaussian covariance function.

4.1 Comparison of Metamodel Results

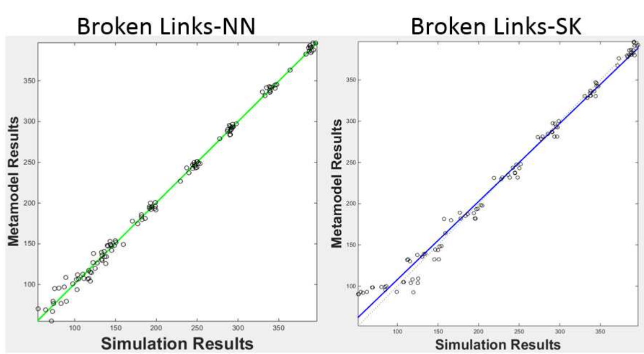
The most important question prior to the implementation of a simulation metamodel involves its accuracy. If the metamodel is not an accurate representation of the original large-scale simulation then it is of no value for analysis. Therefore, for this power grid network problem, testing needed to be performed to determine the goodness of fit of the metamodels with respect to the CFS simulation.

Again, Neural Network and Stochastic Kriging metamodels were calibrated for each of the four network outputs of interest (number of broken links, the number of non-working nodes, the power loss, and the size of the largest giant component) using 80,000 simulation runs. Validation tests using root mean squared error (RMSE) and R-values are promising with the number of broken links and number of non-working nodes responses showing the best fit. The R-values and RMSE values are shown below (Table 1) for the Neural Network metamodels and Stochastic Kriging metamodels for each of the four network outputs of interest. These R-values and RMSE values were computed using testing points that were not involved in the fitting of any of the metamodels. Neural Network metamodels appear to have a slightly better goodness of fit, but Stochastic Kriging performs adequately and even quite well for the broken links and non-working nodes performance measures. One possible reason for Stochastic Kriging's inferior performance was the inability to utilize all 80,000 simulation runs for its calibration. Under nine input variables, our Stochastic Kriging code cannot handle all of those data points. This is due to the significantly large computations from the repeated flops necessary to do the covariance matrix inversions in the calibration procedure. Therefore, for Stochastic Kriging, a smaller subset consisting of 25% of the simulation data was used.

Table 1: R-values and RMSE for the Neural Network and Stochastic Kriging Metamodels.

		Broken Links	Powerloss	Size_Giant_Component	Non-working Nodes
Stochastic Kriging	RMSE	8.123	1220.574	956.841	0.925
	R-value	0.994	0.804	0.745	0.998
Neural Network	RMSE	6.036	635.123	57.833	0.549
	R-value	0.998	0.962	0.951	0.998

In addition to RMSE and R-values for each of the Stochastic Kriging and Neural Network metamodels, goodness of fit plots are also provided below (Fig. 4). These plots are regression plots, plotting the results of the CFS simulation (x axis) to the results generated from the metamodel (y axis). These observations were obtained through design points that were not used to train the simulation metamodel. These regression plots reinforce visually that the Neural Networks had a slightly superior fit for all performance measures and are more reliable for analysis. The y axis of the plots below give insight into the numerical ranges involved for each of the performance measures and a better appreciation for the RMSE values listed above in Table 1.



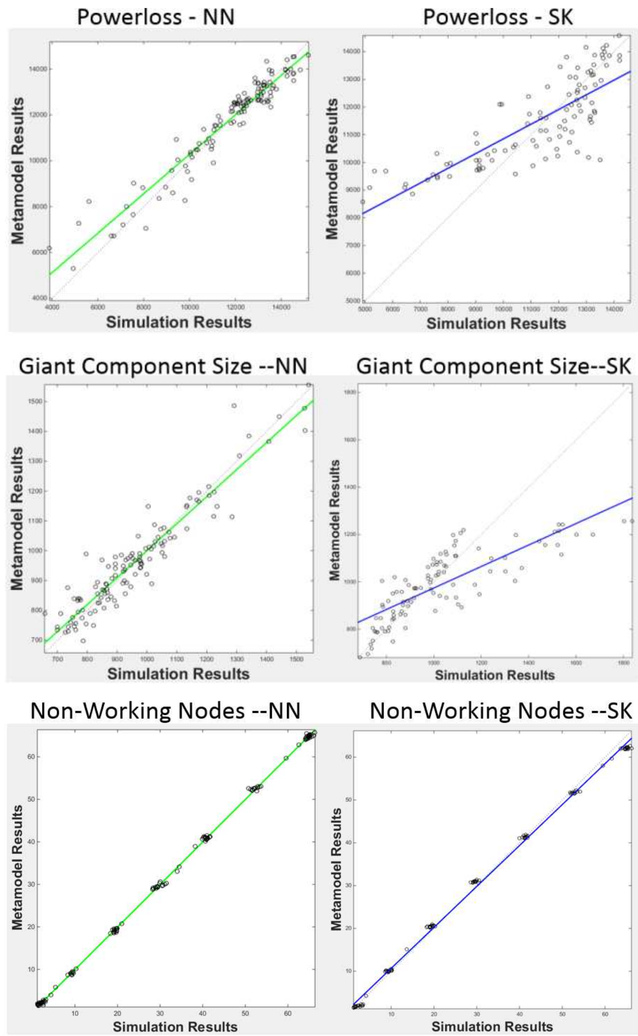


Figure 4: Comparison of Goodness of Fit Results Between Neural Network (NN) Metamodels and Stochastic Kriging (SK) Metamodels for All Four Global Network Performance Measures.

4.2 Visualizations with Metamodeling

In this section we showcase charts for network risk assessment that depict the distribution of the four global network performance measures: number of broken links, size of the giant component, powerloss and number of non-working nodes. The distribution spans across cases where 2-300 links are initially removed from the power grid at the start of the simulation. Each chart is generated through the metamodel and through the simulation model under a specific level of failure probabilities for the eight susceptible links discovered in the Random Chemistry search procedure. The vector of failure probabilities for the eight critical links are [0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8] and are also the inputs to the metamodel. In building these charts, the metamodels were integrated into a Monte Carlo sampling script to generate each of these points. This was accomplished by also fitting metamodels for the variance of each of these four performance measures. Sampling was done through a Gaussian process utilizing the expected value generated through the metamodels shown in 4.1 as well as additional metamodels developed specifically for the variance.

The charts presented for the specific instance of failure probabilities are box-whisker plots where each vertical line represents the distribution over 1000 replications across varying initial event sizes of link removals from 2-300. The dark lines demarcate the first and third quartiles with the black dotted line indicating the median value. Lighter lines represent outliers. Furthermore, near and far outliers are depicted as dark and gray points, respectively. As expected, the four global performance measures of the network degrade on average as more lines are initially removed. However, the variability in network performance as a function of the initial event size emphasizes the fact that not all line combinations are created equal and only a small set of line combinations that result in large blackouts.

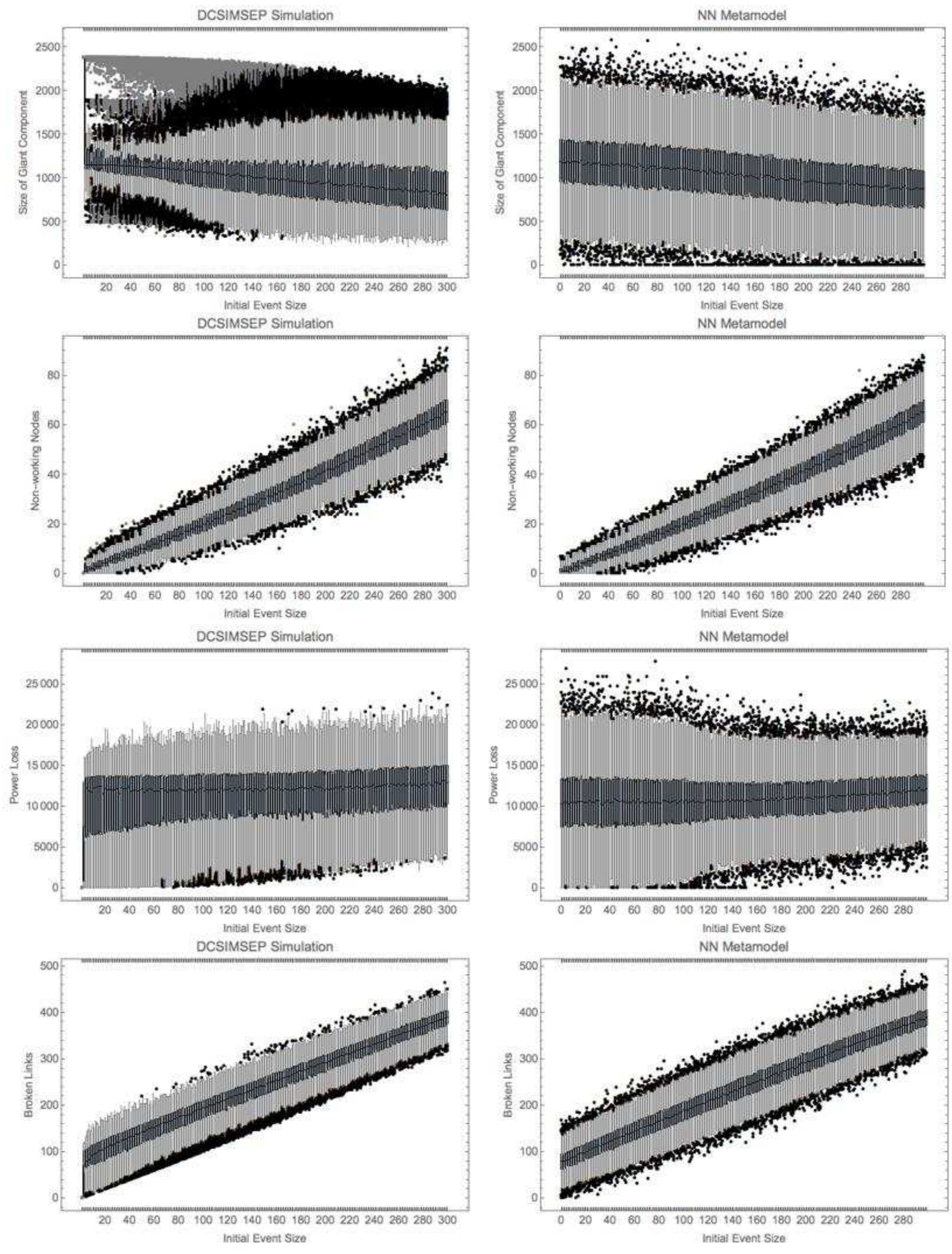


Figure 5: Metamodel Generated and Simulation Generated Box-Whisker Plots for Broken Links and Number of Non-Working Nodes as a Function of the Initial Event Size for the Polish Power Grid.

In Figure 5 box-whisker plots from both the simulation model and metamodel are compared side by side to examine the amount of information that is captured by the metamodel. The number of broken links charts are shown in the top while the non-working nodes box-whisker plots are shown at the bottom. The expected value and trend is captured very nicely by the metamodel for both the number of non-working nodes and the number of broken links. For powerloss and the size of the giant component performance measures, a slight deviation in the expected value can be observed, but the trend is still captured. The variability is captured nicely in the number of non-working nodes and broken links box-whisker plots, but not as nicely with the other two performance measures. For these performance measures the metamodel output is providing a 20% higher variability across each event size, which is due to a poorer fit with these component metamodels for variance. However, the metamodel provides these visualizations in real-time and provides a very similar risk assessment to that of the simulation for all of these performance measures. This is a vast improvement compared to the original simulation model, which requires about 30 seconds to execute a single simulation replication, resulting in about 2500 hours of computing time to generate a single box-whisker plot. When distributing across the 500 nodes in our clustered computing environment, that still translates to 50 clock hours per chart.

5 CONCLUSIONS AND FUTURE DIRECTIONS

The case study presented in this paper demonstrates how simulation metamodeling can be used effectively in a decision support toolkit for the analysis of Critical Infrastructure networks. We showed that both the Neural Network and Stochastic Kriging metamodels developed for the global network performance measures have an adequate goodness of fit with respect to the original simulation. Metamodeling was utilized as part of a two-step analysis process in this study. The first step consisted of an exploration of a power grid network to locate the most malignant links in the network by identifying the set of (n-2) blackout causing contingencies and then determining the most frequently occurring ones across the set. The metamodels are then applied to assess how their failure probabilities affect global network performance measures, such as, size of the giant component, power loss, number of non-working nodes, and the number of broken links. An example of a box-whisker plot was shown to illustrate how the visualizations can be produced through the fitted metamodels in real-time.

This study revealed that metamodel performance for the size of the giant component and power loss performance measure could be improved. Future research involves investigating other metamodeling techniques that could provide a better fit to those performance measures. More research is also being done to integrate more visualizations with the metamodels to aid decision making as well as to develop more visualizations to help with the validation of the metamodels in this decision making environment. In addition, future research involves investigating other system input to output mappings that can be captured effectively by metamodeling to assist risk assessment and risk management practices for critical infrastructure networks.

ACKNOWLEDGMENTS

The authors would like to thank John Kreger, Mary McGinley, and Emily Frye for valuable discussions and their support to this research.

REFERENCES

Al-Hindi, H. 2004. “Approximation Of A Discrete Event Stochastic Simulation Using An Evolutionary Artificial Neural Network.” Engineering Science 15(1): 3-16.

Ankenman, B., B. Nelson, J. Staum. 2010. “Stochastic Kriging For Simulation Metamodeling.” Operations Research 58(2): 371-382.

Barton, R., and M. Meckesheimer. 2006. “Metamodel-Based Simulation Optimization,” Handbook in OR and MS, edited by S.G. Henderson.

Box, G. and N. Draper. 1987. Empirical model-building and response surfaces, John Wiley and Sons, New York, 1987.
Buzas, J. 2013. “Optimized Random Chemistry.” arXiv: 1302.2895.
Carreras, B. 2004. Complex Dynamics of Blackouts in Power Transmission Systems. Chaos 14(3): 643-652.
Deboor, C. 1978. A Practical Guide To Splines, New York, Springer-Verlag.
Dyn, N., D. Levin and S. Rippa. 1986. “Numerical Procedures For Surface Fitting Of Scattered Data By Radial Functions.” SIAM Journal of Scientific and Statistical Computing 7: 639-659.
Eppstein, M. and Himes, P. 2012. “A Random Chemistry Algorithm for Identifying Collections of Multiple Contingencies that Initiate Cascading Failure.” IEEE Transactions on Power Systems: 1698-1705.
Eubank, R. 1988. Spline Smoothing And Nonparametric Regression, Marcel Dekker, New York.
Hussain, M., R. Barton, S. Joshi. 2002. “Metamodeling: Radial Basis Functions, Versus Polynomials.” European Journal of Operational Research 138: 142-154.
Fonseca, D., D. Navaresse, and G. Moynihan. 2003. “Simulation Metamodeling Through Artificial Neural Networks.” Engineering Applications of Artificial Intelligence 16:177-183.
Hect-Nielsen, R. 1989. “Theory Of The Backpropagation Neural Network.” In Proceedings of the International Joint Conference on Neural Networks, 1: 593-605.
Kaplan, S. 1997. The Words of Risk Analysis. Risk Analysis 17(4): 407-417.
Kleijnen, J. 2009. “Kriging Metamodeling In Simulation: A Review.” European Journal of Operational Research 192: 707-716.
Korkali, M. 2014. “Topology, Physics, and Coupling in Independent Power and Communications Networks.” MITRE White Paper.
Koza, J. 1992. Genetic Programming: On The Programming Of Computers By Means Of Natural Selection. MIT Press, Boston.
Lewis, T. 2006. Critical Infrastructure Protection in Homeland Security – Defending a Networked Nation, J. Wiley and Sons, New Jersey, 108-128.
Lippman, R. 1987. “An Introduction To Computing With Neural Nets.” IEEE ASSP Magazine: 4-22.
Meghabghab, M. 2001. “Iterative radial basis functions neural networks as metamodels of stochastic simulations of the quality of search engines in the World Wide Web.” Information Processing and Management 37: 571-591.
Michalski, R. 1983 A Theory And Methodology Of Inductive Learning, Machine Learning: An Artificial Intelligence Approach. TIOGA Publishing Co., Palo Alto.
Montgomery, D. 2008. Design and Analysis of Experiments. 7th Edition, Wiley New York.
Myers, R. 1976. Response Surface Methodology. Allyn and Bacon, Boston.
Myers, R., M. Montgomery, and C. Anderson-Cook. 2009. Response Surface Methodology: Process And Product Optimization Using Designed Experiments, 3rd Edition, Wiley New York.
Myers, R., V. Ugru, X. Luo and R. Grandhi, 1996. “MIDAS for pre- and post-processing of ASTROS unsteady aerodynamic flutter models, 6th AIAA/NASA/ISSMO.” In Symposium of Multi-disciplinary Analysis and Optimization.
Rezaei, P. Hines, P., and Eppstein, M. 2014. “Estimating Cascading Failure Risk with Random Chemistry.” IEEE Transactions on Power Systems, 99: 1-10.
Rosen, S., M. Tierney, C. Saunders, and S. Guharay. 2013. “Configuration of a Standoff Detection System via Rapid, Model Based Systems Engineering.” In Proceedings of the 2013 IEEE Systems of Systems Engineering Conference.
Rosen, S., C. Saunders, and S. Guharay. 2015. “A Structured Approach for Rapidly Mapping Multi-Level System Measures via Simulation Metamodeling.” Systems Engineering 18(1): 87-101.
Sacks, J., W. Welch, T. Mitchell and H. Wynn. 1989. “Design And Analysis Of Computer Experiments.” Statistical Science 4: 409-435.
Shin, M., R. Sargent and A. Goel. 2002. “Gaussian Radial Basis Functions For Simulation Metamodeling.” In Proceedings of the 2002 Winter Simulation Conference, edited by J. L. Snowdon, J. M. Charnes, E.

Yucesan, and C. Chen, 483- 488. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Staum, J. 2009. "Better Simulation Metamodeling: The Why, What, And How Of Stochastic Kriging." In Proceedings of the 2009 Winter Simulation Conference, edited by A. Dunkin, R. Ingalls, E. Yucesan, M. Rossetti, R. Hill, and B. Johansson, 119-133. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

SCOTT L. ROSEN is the Chief Engineer for the MITRE Corporation's Operations Research Department. He has spent the last ten years at MITRE applying Operations Research to some of the United States' most critical and challenging Systems Engineering problems. His research interests include Simulation Optimization, Simulation Metamodeling, Decision Analysis, and Quantitative Systems Engineering. He received his B.S. from Lehigh University in Industrial and Systems Engineering in 1998 and a M.S. and Ph.D. in Industrial Engineering and Operations Research from The Pennsylvania State University in 2000 and 2003, respectively. His email address is srosen@mitre.org.

EMMET BEEKER is a Principal Modeling and Simulation Engineer with The MITRE Corporation. His research interests include large-scale parallel and distributed discrete event simulation and simulation in decision support and command and control systems. He received his B.A. in Mathematics and M.S. in Computer Science from Indiana University. His email address is ebeeker@mitre.org.

DAVID SLATER is a Lead Operations Research Analyst within the MITRE Corporation's Operations Research Department. He has spent the last four years at MITRE applying Applied Mathematics to some of the United States' most critical and challenging Systems Engineering problems. His research interests include complex systems analysis, network science, and complexity science. He received his B.S from Michigan Technological University in 2005 and a M.S. and Ph.D. in Applied Mathematics from Cornell University in 2009 and 2011, respectively. His email address is dslater@mitre.org.

SAMAR K. GUHARAY has been leading research in interdisciplinary problems covering sensing technology, algorithm development, and rapid systems engineering analysis. He has been affiliated with academia in different capacities including Adjunct Professor of the Chemistry Department, Washington State University and Industrial Advisory Board member of the Electrical and Computer Engineering Department, Old Dominion University. He obtained his Ph.D. in Physics in 1980. His email address is sguharay@mitre.org.

GARRY JACYNA is a MITRE Fellow and a recognized expert in signal processing, specializing in analytical-based performance analysis models for several Navy and Department of Homeland Security (DHS) programs, including novel applications of cutting-edge algorithms to real-world problems. These include acoustic reconstruction tools for the analysis of the Russian Kursk disaster, multi-target sonar detection, tracking, and classification algorithms, complexity-based analysis tools for Marine Corps agent-based simulation models, distributed netted sensor localization and tracking methods and system-level decision support tools. He has also developed adaptive image fusion architectures to support a broad range of DHS programs where multiple sensor displays are used. Dr. Jacyna also chairs a number of high-level working groups. His email address is gjacyna@mitre.org.