Evolutionary Design and Machine Learning for Modeling and Simulation

Dave Ryer, PhD
Zach Little, PhD
26 Aug 2019

The goal of the STAT COE is to assist in developing rigorous, defensible test strategies to more effectively quantify and characterize system performance and provide information that reduces risk. This and other COE products are available at www.afit.edu/STAT.

STAT Center of Excellence
2950 Hobson Way – Wright-Patterson AFB, OH 45433

Table of Contents

Executive Summary

This paper addresses research in evolutionary design and machine learning for modeling and simulation. In many instances, high fidelity computer simulations cannot be exhaustively evaluated due to the computational complexity or problem scope. Initial space-filling designs followed by adaptive sampling, also known as active learning, permit efficient meta-modeling, examination, and testing of simulation responses. This repeatable process was successfully implemented for the Threat Modeling and Analysis Program at the National Air and Space Intelligence Center.

Keywords: evolutionary design, adaptive sampling, active learning, meta-modeling

Introduction

Many modeling and simulation (M&S) studies can suffer from the curse of dimensionality, making exhaustive evaluation in the form of full-factorial designs infeasible given available computation resources. A repeatable, iterative, and efficient process is desired for examination and testing of models within the intelligence community. The methodology was developed in successive phases: (I) development of the overall process, (II) implementation of evolutionary design approaches, and (III) rapid experimentation and adaptive learning.

Background

The Threat Modeling and Analysis Program (TMAP) is developed and used across the intelligence community, employing a mature reoccurring process that begins with threat assessments and intelligence collections. Through a combined process of engineering design, testing and analysis a digital representation (or model) of a threat system is produced. The modeling portfolio represents many systems with expertise and developmental efforts spanning a variety of organizations and centers. TMAP serves a large customer base that includes our allies, all military services, major commands and research and acquisition centers. The models are represented at various levels of fidelity (parametric, analytic, and emulative) and are utilized across the traditional DoD M&S pyramid (engineering, engagement, mission and campaign). The high resolution and computationally intensive models can be efficiently explored using experimental design of the inputs, intelligent sampling methods and collecting relevant outputs to create accurate and efficient meta-models, or mathematical *models of models*, that act as effective surrogates for the system model (Figure 1).

Figure 1: Experimental Design to Meta-modeling

Project Progress

Phase I: Process Development

The initial focus of this research was to provide efficient designs to explore high resolution simulations. These models contain numerous inputs (factors) and outputs (responses) with complex, non-linear behavior. Earlier use of these models employed fine grid designs and sensitivity analysis conducted with a variety of stochastic parameters. After a brief familiarization with a single model, initial project goals were outlined as follows:

- Implement efficient experimental designs and investigative methods
- Effectively manage numerous stochastic inputs by incorporating into design space
- Leverage model failure indicators to identify input combinations that yield feasible solutions
- Define and characterize the system model's operating envelope

Using a single default scenario as a starting design point, relevant input bounds were defined to create the initial space-filling design. As the experimental bounds were expanded, previously evaluated design points were integrated and perturbed to generate solutions across the larger operating range. The efficient designs were able to integrate controllable system parameters previously addressed as stochastic inputs. A simple two-layer feed-forward neural network was developed to recognize patterns present in the simulation's success and failure indicators to predict future success. This metamodel was used to evaluate candidate inputs and select those used for future modeling runs. After applying classification and expansion methods, the focus shifted to automating and accelerating the process through machine learning and distributed computing.

Phase II: Evolutionary Design

During this phase, the methods and techniques were built in the native MATLAB/Simulink environment for model development and testing, where the tasks were automated in a repeatable process (Figure 2):

- Design (initial inputs and parameter space)
- Simulate and Harvest (generate, collect and leverage data)

- Learn (dynamically create and retrain meta-models)
- Explore (utilize results to intelligently guide investigation; either through expansion of input bounds or focusing on specific regions of the system)

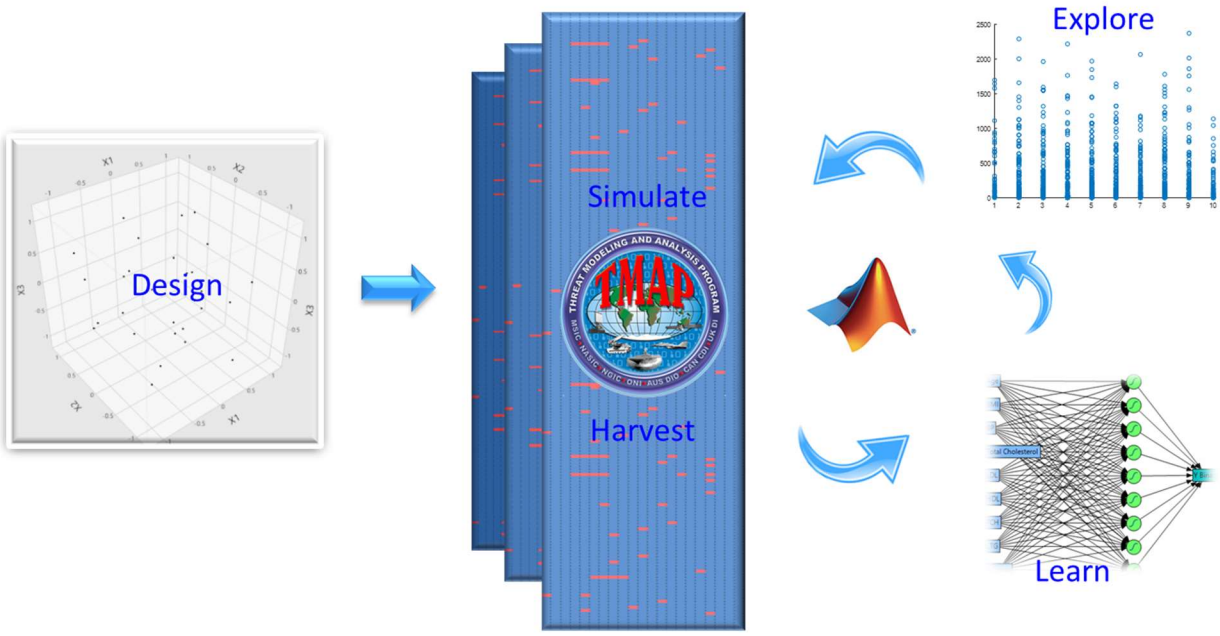

Figure 2: Evolutionary Design

Efficient space-filling experimental designs such as the Latin hypercube (McKay et al.; Iman and Conover; Florian; Owen; Tang) created an initial sampling of the design space and incorporated a variety of scenario details. These initial designs were a foundation for covering the multi-dimensional space but did not consider system responses and variable relationships. The systems were simulated over wide range of autogenerated inputs and data was harvested from all evaluations. This data was then used to construct efficient meta-models (in particular, neural networks (Rosenblatt; McCulloch and Pitts; Hebb)) to more efficiently guide which design points would be selected for further evaluation/simulation by representing input-output relationships over the entire span of the evaluated design space.

The combined steps for design, simulation and harvesting subsequently enabled the rapid evaluation of input combinations prior to selecting design points for simulation runs. The initial generation of candidate inputs are guided by predefined probability distributions, likelihood of success predicted by fitted meta-models, and uniqueness as defined by multi-dimensional measurements such as Mahalanobis distance. The selection of design points for the simulation are determined by neural networks that are updated every learning cycle. The generation of search agents are normally scheduled at the onset of the investigation and are subsequentially managed by multi-objective expiration conditions. Supervised learning and classification techniques are used to adjust factor limits, step sizes

and retrain metamodels by leveraging data and collective knowledge from previous learning cycles and shared across the multiple agents exploring the model design space.

Phase III: Experimentation, Rapid Learning and Adaptation

After automating the design and exploration process, the next phase focused on rapid experimentation following many of the objectives stated in the DoD AI Strategy (DoD, 2018). Experimental practices were established for continuous learning through iteration, adaption and frequent updates. The successful prototypes were scaled across multiple system models in an agile development process. Collaboration was encouraged through the creation and distribution of reusable tools and frameworks. The pioneering paradigms and methods were applied across the organization's modeling portfolio and shared across their analytical enterprise.

A subsequent transition to an open development environment created a rapid prototyping and testing process that resulted in an assortment of enhancements for design, adaptive sampling, meta-modeling, and data visualization. The creation of an experimental framework provided a modular code structure to efficiently examine and select the best combination of methods. With these tools and methods analysts can be supported through a streamlined and nearly autonomous design and test process. The experimental framework provides a structure for capturing and tracking many aspects of the modeling and investigation process. Some of the components used in the current framework are listed below.

- Design space
    - Factors/inputs
    - Initial input bounds
    - Final input bounds (i.e., hard system constraints)
- Evaluative system/model
- Output data
    - Success/failure flags
    - System measures of performance/interest
    - Computation time
- Meta-model(s)
- Evolutionary agents

Extensions to this framework include the implementation of initial designs beyond the Latin hypercube to permit more uniform sampling (Fang et al.), account for input constraints by using clustering-based designs such as with the Fast Flexible Filling designs in JMP (Lekivetz and Jones, "Fast Flexible Space-Filling Designs for Nonrectangular Regions"; Lekivetz and Jones, "Fast Flexible Space-Filling Designs with Nominal Factors for Nonrectangular Regions"), and allow for mixed factor spaces (i.e., designs with categorical, continuous, and discrete factors having different numbers of levels for each) by using Nearly Orthogonal-and-balanced (NOAB) designs (Vieira Jr. et al.; Little et al.). Sequential sampling includes the exploratory Monte Carlo Voronoi approach (van der Herten et al.) to iteratively improve space-filling in the updated design.

Adaptive sampling approaches leverage model responses to further examine design space regions associated with high non-linearity or complexity, while meta-models are used to exploit regions having large error, variance, or model disagreement. Since no single meta-model type is expected to provide the best fit to every system of interest, meta-modeling approaches can be extended beyond neural networks to include both parametric and non-parametric methods such as linear regression (Kutner et al.), classification and regression trees (Breiman et al.), Gaussian processes or Kriging (Matheron), multivariate adaptive regression splines (Friedman), random forests (Breiman), and support vector machines (Drucker et al.). JMP software currently allows for construction of neural networks, classification and regression trees, Kriging, and polynomial regression. MATLAB software has a neural network toolbox as well as functions for classification and regression trees, polynomial regression, radial basis functions, and support vector machines in addition to open-source options for Kriging and multivariate adaptive regression splines. R software and Python both have open-source library packages (*caret* and *scikit-learn*, respectively) and scripts available for each of these modeling techniques. Interactive data visualizations can be rapidly implemented using open tools such as Shiny in R and Bokeh in Python.

Results

Meta-models that can sufficiently act as surrogates to model responses are a useful product for engineers, subject matter experts, and their customer base to understand both their models and associated real-world threat systems. Interactive experimental controls and multi-dimensional data visualizations aid in model understanding. Through the three phases of this project, the initial experimental process has moved from representing and understanding a single model in months using a traditional offline approach, to multiple models in weeks using evolutionary design and machine learning, with the approaching goal of evaluating a model in hours and spanning an entire portfolio in a distributed computing environment. Associated validation and verification efforts for these models are vastly improved by the efficiencies in understanding of such large, stochastic input spaces. These proficiencies allow for rapid understanding of threat models but also allow resources to be assigned to multiple models simultaneously using the evolutionary design approach. Figure 3 provides an overview of current timelines and productivity gains achieved for the threat modeling and analysis program using this process.

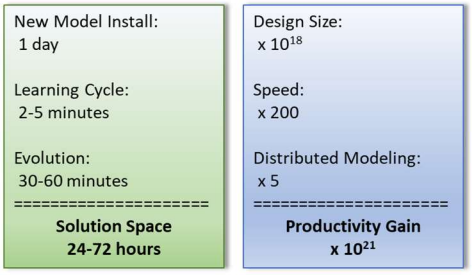

Figure 3: Timelines and Efficiencies Gained thru Evolutionary Design Processes

Conclusion

The evolutionary design process was successfully applied for threat modeling and analysis to evaluate model input spaces and continuously explore, learn, and refine model response surfaces. Sufficiently accurate meta-models of model responses allow for faster exploration and evaluation across the input space than if using the threat models alone, resulting in significant productivity gains for engineers and subject matter experts. The process has now been extended to an open development environment to facilitate efficient incorporation of new experimental design, adaptive sampling, and meta-modeling approaches as well as ensure reusability and scalability across multiple mission areas.

References

Breiman, L., et al. *Classification and Regression Trees*. Wadsworth & Brooks, 1984.

Breiman, Leo. "Random Forests." *Machine Learning*, vol. 45, no. 1, 2001, pp. 5–32, doi:10.1023/A:1010933404324.

Department of Defense (DoD), Summary of the 2018 Department of Defense Artificial Intelligence Strategy - Harnessing AI to Advance Our Security and Prosperity, Washington DC, 2018.

Drucker, Harris, et al. "Support Vector Regression Machines." *Advances in Neural Information Processing Systems*, vol. 9, 1997, pp. 155–61.

Fang, Kai-Tai, et al. "Uniform Design: Theory and Application." *Technometrics*, vol. 42, no. 3, 2000, pp. 237–48.

Florian, Aleš. "An Efficient Sampling Scheme: Updated Latin Hypercube Sampling." *Probabilistic Engineering Mechanics*, vol. 7, no. 2, 1992, pp. 123–30, doi:10.1016/0266-8920(92)90015-a.

Friedman, Jerome H. "Multivariate Adaptive Regression Splines." *The Annals of Statistics*, vol. 19, no. 1, 1991, pp. 1–141.

Iman, Ronald L., and W. J. Conover. "A Distribution-Free Approach to Rank Correlation among Input Variables." *Communications in Statistics - Simulation and Computation*, vol. 11, no. 3, 1982, pp. 311–34, doi:10.1080/03610918208812265.

Kutner, Michael H., et al. *Applied Linear Regression Models*. McGraw-Hill, 2004.

Lekivetz, Ryan, and Bradley Jones. "Fast Flexible Space-Filling Designs for Nonrectangular Regions." *Quality and Reliability Engineering International*, vol. 31, no. 5, 2015, pp. 829–37, doi:10.1002/qre.1640.

---. "Fast Flexible Space-Filling Designs with Nominal Factors for Nonrectangular Regions." *Quality and Reliability Engineering International*, vol. 35, no. 2, 2019, pp. 677–84, doi:10.1002/qre.2429.

Little, Zachary C., et al. "Second-Order Extensions to Nearly Orthogonal-and-Balanced (NOAB) Mixed-Factor Experimental Designs." *Journal of Simulation*, vol. 00, no. 00, Taylor & Francis, 2018, pp. 1–12, doi:10.1080/17477778.2018.1533794.

Matheron, Georges. “Principles of Geostatistics.” *Economic Geology*, vol. 58, no. 8, 1963, pp. 1246–66.

McKay, Michael D., et al. “Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code.” *Technometrics*, vol. 21, no. 2, 1979, pp. 239–45, doi:10.2307/1268522.

Owen, Art B. “Controlling Correlations in Latin Hypercube Samples.” *Journal of the American Statistical Association*, vol. 89, no. 428, 1994, pp. 1517–22, doi:10.2307/2291014.

Tang, Boxin. “A Theorem for Selecting OA-Based Latin Hypercubes Using a Distance Criterion.” *Communications in Statistics - Theory and Methods*, vol. 23, no. 7, 1994, pp. 2047–58, doi:10.1080/03610929408831370.

van der Herten, J., et al. “A Fuzzy Hybrid Sequential Design Strategy for Global Surrogate Modeling of High-Dimensional Computer Experiments.” *SIAM Journal on Scientific Computing*, vol. 37, no. 2, 2015, pp. A1020–39, doi:10.1137/140962437.

Vieira Jr., H., et al. “Efficient, Nearly Orthogonal-and-Balanced, Mixed Designs: An Effective Way to Conduct Trade-off Analyses via Simulation.” *Journal of Simulation*, vol. 7, no. 4, Nature Publishing Group, 2013, pp. 264–75, doi:10.1057/jos.2013.14.