Future Generation Computer Systems 56 (2016) 360–374

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Predictive analytics using statistical, learning, and ensemble methods to support real-time exploration of discrete event simulations

Walid Budgaga[a], Matthew Malensek[a,*], Sangmi Pallickara[a], Neil Harvey[b], F. Jay Breidt[c], Shrideep Pallickara[a]

[a] Department of Computer Science, Colorado State University, Fort Collins, CO, USA
[b] School of Computer Science, University of Guelph, Guelph, Ontario, Canada
[c] Department of Statistics, Colorado State University, Fort Collins, CO, USA

HIGHLIGHTS

- Our approach enables fast, accurate forecasts of discrete event simulations.
- The framework copes with high dimensionality and voluminous datasets.
- We facilitate simulation execution with cycle scavenging and cloud resources.
- We create and evaluate several predictive models, including ensemble methods.
- Our framework is made accessible to end users through an interactive web interface.

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form
18 June 2015
Accepted 24 June 2015
Available online 15 July 2015

Keywords:
Discrete event simulation
Latin Hypercube Sampling
Distributed execution
Cloud infrastructure

ABSTRACT

Discrete event simulations (DES) provide a powerful means for modeling complex systems and analyzing their behavior. DES capture all possible interactions between the entities they manage, which makes them highly *expressive* but also compute-intensive. These computational requirements often impose limitations on the breadth and/or depth of research that can be conducted with a discrete event simulation.

This work describes our approach for leveraging the vast quantity of computing and storage resources available in both private organizations and public clouds to enable real-time exploration of discrete event simulations. Rather than directly targeting simulation execution speeds, we autonomously generate and execute novel *scenario variants* to explore a representative subset of the simulation parameter space. The corresponding outputs from this process are analyzed and used by our framework to produce models that accurately forecast simulation outcomes in real time, providing interactive feedback and facilitating exploratory research.

Our framework distributes the workloads associated with generating and executing scenario variants across a range of commodity hardware, including public and private cloud resources. Once the models have been created, we evaluate their performance and improve prediction accuracy by employing dimensionality reduction techniques and ensemble methods. To make these models highly accessible, we provide a user-friendly interface that allows modelers and epidemiologists to modify simulation parameters and see projected outcomes in real time.

1. Introduction

The behavior of complex, real-world systems is often difficult to predict or fully understand. These systems may be influenced by any number of internal or external stimuli, and direct experimentation is often prohibitively expensive, time-consuming, or simply not feasible. In these situations, computer simulation is a compelling solution. Specifically, discrete event simulations (DES) model all possible interactions between entities in a system, making them highly *expressive*. To model uncertainty in these interactions, *stochastic* discrete event simulations associate probabilities with each of their events. However, this expressiveness comes at the cost of increased computational complexity and prolonged execution times.

* Corresponding author.
E-mail addresses: budgaga@cs.colostate.edu (W. Budgaga), malensek@cs.colostate.edu (M. Malensek), sangmi@cs.colostate.edu (S. Pallickara), neilharvey@gmail.com (N. Harvey), jbreidt@stat.colostate.edu (F.J. Breidt), shrideep@cs.colostate.edu (S. Pallickara).

http://dx.doi.org/10.1016/j.future.2015.06.013
0167-739X/© 2015 Elsevier B.V. All rights reserved.

Our subject discrete event simulation, the North American Animal Disease Spread Model (NAADSM) [1] is an epidemiological model of disease outbreaks in livestock populations. Livestock are simulated as individual *herds* and interact with their environment through events; for instance, an *exposure* event may occur when a particular herd has come into contact with a disease of interest. The simulation has been applied in studies of several different diseases, including foot-and-mouth disease [2], avian influenza [3], and pseudorabies [4]. NAADSM is a stochastic DES: simulations are run many times, with each *iteration* contributing to an overall representation of the output variables' *probability distributions*. Iterations often require several hours of CPU time to execute depending on how events unfold.

The computational complexity of these stochastic iterations makes it difficult for planners and epidemiologists to perform exploratory "what if" analysis that plays an important role in planning and preparedness. For instance, a planner may make subtle adjustments to quarantine procedures or the number of vaccines available in order to analyze economic consequences or how disease spread might change. Each modification of the input parameters requires a new set of iterations to be run. Dividing the target simulation into several units and executing them in parallel is one way to improve overall execution times [5,6], but generally does not enable real-time exploration. In this work, we target real-time computational guarantees that involve providing subsecond, interactive responses to the user as simulation parameters are changed.

This paper describes our approach for retaining the expressiveness of stochastic DES while addressing the weaknesses in the timeliness of their outcomes. We achieve this by utilizing voluminous epidemic simulation data to glean insights and derive relationships between scenarios and outcomes. We then use this information to create models that can forecast the results for an entire class of input parameters, enabling our system to provide real-time answers to exploratory investigations.

1.1. Research challenges

We consider the problem of generating fast, accurate DES forecasts for a given subset of the input parameter space. These forecasts are generated in lieu of compute-intensive simulation runs. Challenges involved in accomplishing this include:

1. *Data Dimensionality*: Each input parameter represents a dimension, the number of which can be quite high (approximately 1800–2500 in this particular study). Furthermore, input parameters come in a variety of types: integers, floats, or even probability distributions.
2. *Interactive Exploration*: The "what if" scenarios in question must provide immediate feedback during exploration; every parameter change will result in slightly different outputs that must be forecast in real time.
3. *Accuracy*: Outputs produced during exploration must be reasonably accurate to ensure their usefulness. Once a planner has determined parameters of interest, he or she may decide to perform a set of actual simulation runs.

1.2. Research questions

Specific research questions we explore include:

1. How can we minimize the number of iterations required to build our models while still ensuring statistical coverage of the parameter space?
2. What are the implications of our execution model, and how can we obtain necessary processing resources?
3. A large amount of training data is necessary for making predictions. How can this data be managed in a scalable and fault-tolerant manner?
4. How can we deal with increases in dimensionality as the number of input parameters grows?
5. What prediction models can provide both **accurate** and **real-time** results?
6. How can we improve model performance? What impact does the relative error, feature correlations, and input dataset size have on predictive performance?
7. Once the models are built, how can we make their insights available to users in an accessible and efficient manner?

1.3. Summary of approach

Our approach treats the DES in question as a black box and focuses on deriving relationships between the inputs and outputs. Given a disease spread scenario, our framework views input tuples as points in the multidimensional parameter space. We first derive bounds for each of the dimensions from both historical data and subject-matter experts, and then sample within this parameter space to create novel *scenario variants*. Our objective is two-fold: we wish to ensure adequate coverage of the parameter space, while also controlling the size of computational workloads.

For each scenario, we inspect the variances of key output variables to derive the number of iterations that must be executed. Both the variant generation and their subsequent simulation iterations are implemented as MapReduce [7] jobs that are orchestrated by our *Forager* framework. Forager deals with highly elastic resource pools and can scavenge for CPU cycles on both physical and virtual machines, including spot instances in the cloud. These simulation runs generate a large amount of data, often producing terabytes of outputs in a few hours. To cope with these storage demands, we use a distributed storage system to manage the data in a scalable and fault-tolerant manner.

Once the simulation iterations have been executed, we model the relationships between inputs and outputs. To facilitate predictions, we create a model for each output variable. We consider both linear (multivariate linear regression) and non-linear (artificial neural networks) methods to construct these models, and use *k*-fold cross-validation to assess their generalizability. To further improve predictive performance, we investigate the use of ensemble methods to reduce model bias (gradient boosting) and variance (random forests). We also consider the effects of dimensionality and collinearity in the input dataset to reduce model noise and creation times.

The technologies discussed in this study enable our system to provide accurate answers to "what if" scenarios in real time. We make this information accessible to planners and epidemiologists through a web-based user interface that targets a broad range of devices and platforms. This allows interactive modification of scenario parameters with direct feedback.

1.4. Paper contributions

This paper describes our approach for supporting interactive exploration of discrete event simulations. The research involves several key features, including the use of analytics to ensure accurate and timely forecasts that account for statistical coverage of the parameter space, orchestration of workloads, generation and management of training data, correlations between inputs and outputs, dimensionality reduction, and the use of learning structures. Our specific contributions include:

- *Applicability*: The framework is broadly applicable to other compute-intensive simulations. We treat a given simulation as a black box and focus on deriving the relationship between inputs and outputs.