Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S. -H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds.

SUPPORTING A MODELING CONTINUUM IN SCALATION: FROM PREDICTIVE ANALYTICS TO SIMULATION MODELING SUPPLEMENT

John A. Miller
Michael E. Cotterell
Stephen J. Buckley

Department of Computer Science
University of Georgia
Athens, GA, 30602, USA

IBM Thomas J. Watson Research Center
Yorktown Heights, NY, USA

ABSTRACT

Predictive analytics and simulation modeling are two complementary disciplines that will increasingly be used together in the future. They share in common a focus on predicting how systems, existing or proposed, will function. The predictions may be values of quantifiable metrics or classification of outcomes. Both require collection of data to increase their validity and accuracy. The coming era of big data will be a boon to both and will accelerate the need to use them in conjunction. This paper discusses ways in which the two disciplines have been used together as well as how they can be viewed as belonging to the same modeling continuum. Various modeling techniques from both disciplines are reviewed using a common notation. Finally, examples are given to illustrate these notions.

This supplememnt to the paper provides all figures, tables and substantial code listings, all of which would not fit into the paper due to page limitations. First paragraphs of sections are included to establish proper context.

1 INTRODUCTION

Two disciplines, predictive analytics and simulation modeling, are currently expanding their scope and are likely to increase their commonalities in the near future. On the one hand, predictive analytics attempts to make sense of data by finding patterns or fitting statistical models. On the other hand, simulation modeling attempts to mimic reality. Simulation requires data for fitting distributions and estimating parameters. One may view the two disciplines as two ends of the same continuum. Although somewhat of an overstatement, one end could be described as data-rich and knowledge-poor, while the other could be viewed knowledge-rich and data-poor.

...

The rest of the paper is organized as follows: Using a common notation, sections 2 and 3 provide some necessary background on predictive analytics and simulation modeling, respectively. Section 4 makes the case for a modeling continuum based on the richness of data and knowledge utilized by various modeling techniques. Example problems illustrating commonalities and trade-offs between the various techniques are discussed in sections 5 and 6. Finally, conclusions and future work are given in section 7. Due to space limitations, all figures and many code listings are provided in the on-line supplement (see http://www.cs.uga.edu/~jam/scalation/examples/simopt).

2 PREDICTIVE ANALYTICS

As the name predictive analytics indicates, the purpose of techniques that fall in this category is to develop models to predict outcomes. For example, the distance a golf ball travels y when hit by a driver depends on several factors or inputs $\mathbf{x}$ such as club head speed, barometric pressure, and smash factor (how square the impact is). The models can be developed using a combination of data (e.g., from experiments) and knowledge (e.g., Newton's Second Law). The modeling techniques discussed in this section tend to emphasize the use of data more than knowledge.

...

2.1 Time-Independent Models

In time-independent models, the time argument t is removed from the prediction function. Although these techniques are thought of as time-independent, it is still possible to interpret one of the x_is as time. It is just that time is not a dominate feature as it is in the next section on time-dependent models.

...

2.2 Time-Dependent Models

If time t is a key feature involved in the modeling, there are a variety of modeling techniques that can be applied to time series data, starting with the classical ARMA models.

...

2.3 Models Based on Classifiers

When the output/response y is defined on small domains, e.g., $\mathbb{B}$ or $\mathbb{Z}_k = \{0, \; ... \; k-1\}$, then some classifiers used in data mining can be used for predictive analytics.

...

3 SIMULATION MODELING

The most recent version of the Discrete-event Modeling Ontology (DeMO) lists five modeling paradigms or world-views for simulation. So far in this paper, discussion has focused on functions with two vectors, the input $\mathbf{x}$ and output $\mathbf{y}$, and a scalar time t. Simulation modeling adds to these the notion of state, represented by a vector-valued function $\mathbf{s}(t)$. Knowledge about the internals of a system or process is used to define state as well as how state can change over time. Theoretically, this should make such models more accurate, more robust, and have more explanatory power. More about this in the next section. Ultimately, we may still be interested in how inputs affect outputs, but to increase the realism of the model with the hope of improving its accuracy, much attention must be directed in the modeling effort to state and state transitions. This is true to a degree with most simulation modeling paradigms or world views. These paradigms are briefly discussed below and explained in detail in (Silver et al. 2011).

```
// Golf Ball Flight Dynamics Model
val n  = 100                                // maximum number of time points
val tm =   5.                               // simulate for a maximum of tm sec
val g  =   9.80665                          // gravitational force in m/sec^2
val m  =  45.93                             // mass of a golf ball in grams
val aa =  15.00                             // launch angle in degrees
val ss = 100.00                             // swing speed in miles/hour
val sf =   1.49                             // smash factor
val s  = ss * sf * 1609.344 / 3600          // initial ball speed in m/sec
val a  = aa * Pi / 180.                     // launch angle in radians
val y0 = new VectorD (0., 0.)               // initial position (y_0, y_1)
```

```
val v0 = new VectorD (s*cos(a), s*sin(a))  // initial velocity at t0

// define the system of Ordinary Differential Equations (ODEs)
def dy1_dt (t: Double, y: VectorD) = v0(0)              // ODE 1
def dy2_dt (t: Double, y: VectorD) = v0(1) - g * t      // ODE 2
val odes: Array [Derivative] = Array (dy1_dt, dy2_dt)

val dt = tm / n                                        // time step
var t  = dt                                            // next time point to examine
var y  = new VectorD (2)                               // current ball position

breakable { for (i <- 1 to n) {
   y = DormandPrince.integrateV (odes, y0, t)  // compute new position
   if (y(1) < 0.) break                        // quit after hitting ground
   println ("> at t = " + "%4.1f".format (t) + " y = " + y)
   t += dt
}} // for
```

...

3.1 Simulation Optimization

Simulation optimization is becoming more popular and may be used for optimizing designs or for improving the models themselves (Pasupathy and Henderson 2011). One can think of a simulation model as having a parameter vector **b** that needs to be estimated or fit based on pairings of input and output vectors $\{\mathbf{y}, \mathbf{x}\}$. In some cases, such as biochemical pathways, kinetics parameters are hard to measure directly, so an alternative is to adjust them by using simulation optimization to bring simulation results in line with experimental data. This is completely analogous to want happens in machine learning, where a training set of data is used calibrate or adjust parameters in a model (e.g., the weights W and V in Neural Nets). The optimization techniques themselves may be very similar. ScalaTion supports the development of simulation optimization solutions and includes several optimization algorithms, e.g., Linear Programming (Simplex), Integer Programming (Branch and Bound), Quadratic Programming (Quadratic Simplex), Nonlinear Programming (Steepest Descent, Conjugate Gradient and Quasi-Newton), and Heuristics (Tabu Search and Genetic Algorithm). Furthermore, the SoPT ontology (Han et al. 2011) can assist users developing such solutions.

4 MODELING CONTINUUM

Having examined techniques in both predictive analytics and simulation modeling utilizing a common notation, the paper now considers how these techniques can be viewed to belong to the same continuum.
...

5 APPLICATION TO HEALTHCARE

In the healthcare domain, one problem to be addressed for emergency departments/urgent care facilities is that of staffing. The solutions provided below are simplified to better illustrate the techniques. For more information on problem of this type, please see (Tan et al. 2012) Given an estimated demand, how many of various types of staff members should be hired, i.e., how many triage nurses, registered nurses, nurse practitioners, doctors and administrative clerks should be hired. The model includes $l = 2$ types of patients (regular and severe) and $m = 5$ types of employees.

The goal is to maximize a utility function based on profit as well as patient satisfaction that factors in a dis-utility proportional to patient waiting times. The optimization problem may be formulated as follows:

$$\textbf{max } u(\mathbf{x}) \textbf{ subject to } \mathbf{x} \in \mathbb{Z}_+^m$$

Table 1: Model Definitions

Variable	**Obtained**	**Description**	**Units**
λ_k	input	arrival rate for patients of type k	hr^{-1}
μ_{jk}	input	service rate at resource j for patients of type k	hr^{-1}
f_k	input	fee charged to patients of type k	\$
d	input	patients dis-utility of waiting	\$ / hr
s_j	input	salary/wage for staff of type j	\$ / hr
x_j	optimize	staffing level for type j employees	none
n	output	treatment rate for patients	hr^{-1}
w	output	waiting time for patients	hr
c	$\mathbf{s} \cdot \mathbf{x}$	operating cost	\$ / hr
r	$\mathbf{f} \cdot \mathbf{n}$	revenue for patient service	\$ / hr
p	$r - c$	net profit	\$ / hr
u	$p - dnw$	overall utility	\$ / hr

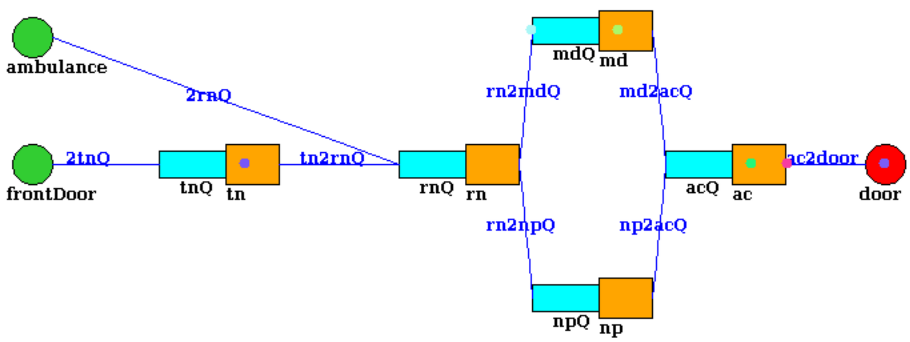

Figure 1: Screenshot of the Animation Produced by ScalaTion.

This is Integer Nonlinear Programming Problem (INLP) where unless assumptions/approximations are made there is no closed-form expression for the objective function $u(\mathbf{x})$. For this modeling/optimization problem, the following three techniques are utilized: Process-Interaction Simulation Models, Queueing Networks and Multiple Linear Regression.

```
\\ Utility Function for Healthcare Application
def u (x: VectorI, s: Double, d: Double): Double = {
    val m = new EmergencyModel(x)        // create model
    val results = m.simulate()           // simulate and gather stats
    val w = sumOfMeans(m.tnQ, m.rnQ,     // sum of average waiting times
                       m.mdQ, m.npQ,
                       m.acQ)
    val n = new VectorD (m.low, m.high) // patients served of each type
    val c = x dot s                      // operating cost
    val r = f dot n                      // revenue for patient service
    val p = r - c                        // net profit
    p - d * w * n.mean                   // return overall utility
} // u
```

...

6 APPLICATION TO SUPPLY CHAIN MANAGEMENT

In this paper we claim that predictive analytics is more reliant on data, while simulation modeling is more reliant on knowledge. Effectively combining data and knowledge can lead to more accurate and informative modeling. Supply chain management is one of the most mature fields of analytics and provides excellent support for our claims. A wide variety of time-dependent predictive analytics techniques are used in supply chain management to forecast product demand (Box and Jenkins 1976). As shown in Figure 2 (see http://www.cs.uga.edu/~jam/scalation/examples/simopt), forecasts of product demand feed the overall supply chain process, whose goal is to provide inventory to satisfy demand on a continuing basis. Simulation is often used to assess whether a supply chain will truly satisfy demand in the presence of a variety of uncertainties such as forecast error, supplier lead time, manufacturing lead time, and manufacturing yield. Here are a few of the many examples of supply chain simulation:

...

7 CONCLUSIONS AND FUTURE WORK

Analytics and modeling is a vast landscape with numerous competing and complementary techniques. Positioning these techniques along a modeling continuum as well as creating taxonomies and ontologies to describe and inter-relate them can help illuminate this vast landscape. From the available/obtainable knowledge and data and the purpose of a particular modeling study, appropriate techniques can be chosen from the modeling continuum.

...

REFERENCES

Box, G., and G. Jenkins. 1976. *Time Series Analysis: Forecasting and Control*. Upper Saddle River, NJ, USA: Holden-Day.

Chen-Ritzo, C., L. An, S. Buckley, P. Chowdhary, T. Ervolina, N. Lamba, Y. Lee, and D. Subramanian. 2007. “Pandemic Business Impact Modeler”. In *Proceedings of the 2007 INFORMS Simulation Society Research Workshop*, ?–?: The Society for Computer Simulation.

Feigin, G., C. An, D. Connors, and I. Crawford. 1996. “Shape Up, Ship Out”. *ORMS Today*.

Fletcher, C. 2012. “Innovation Insight: Price Optimization Leverages CRM, Big Data and In-Memory Computing to Increase Profits and Innovation in the Customer Experience”. Technical report. Gartner Report G00234031.

Glover, F., and M. Laguna. 1998. *Tabu Search*, Volume 1. Kluwer Academic Pub.

Han, J., J. A. Miller, and G. A. Silver. 2011. “SoPT: Ontology for Simulation Optimization for Scientific Experiments”. In *Proceedings of the 2011 Winter Simulation Conference*, 2914–2925: IEEE.

IBM 2012. “BM Quality Early Warning System”. Technical report.

Lin, G., M. Ettl, S. Buckley, S. Bagchi, D. Yao, B. Naccarato, R. Allan, K. Kim, and L. Koenig. 2000. “Extended-enterprise supply-chain management at IBM personal systems group and other divisions”. *Interfaces* 30 (1).

Pasupathy, R., and S. Henderson. 2011. “SimOpt: A Library of Simulation Optimization Problems”. In *Proceedings of the 2011 Winter Simulation Conference (WSC)*, 4075–4085. IEEE.

Silver, G. A., J. A. Miller, M. Hybinette, G. Baramidze, and W. S. York. 2011. “DeMO: An Ontology for Discrete-event Modeling and Simulation”. *Simulation* 87 (9): 747–773.

Tan, W.-C., P. J. Haas, R. L. Mak, C. A. Kieliszewski, P. G. Selinger, P. P. Maglio, S. Glissman, M. Cefkin, and Y. Li. 2012. “Splash: A Platform for Analysis and Simulation of Health”. In *Proceedings of the 2nd ACM SIGHIT Symposium on International Health Informatics*, 543–552. ACM.