

## Publication II

Jirka Poropudas and Kai Virtanen. 2010. Simulation metamodeling in continuous time using dynamic Bayesian networks. In: Björn Johansson, Sanjay Jain, Jairo Montoya-Torres, Joe Hagan, and Enver Yücesan (editors). Proceedings of the 2010 Winter Simulation Conference (WSC 2010). Baltimore, Maryland, USA. 5-8 December 2010. Pages 935-946. ISBN 978-1-4244-9864-2.

© 2010 Institute of Electrical and Electronics Engineers (IEEE)

Reprinted, with permission, from IEEE.

This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of Aalto University's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to [pubs-permissions@ieee.org](mailto:pubs-permissions@ieee.org).

By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

*Proceedings of the 2010 Winter Simulation Conference*  
B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, eds.

## SIMULATION METAMODELING IN CONTINUOUS TIME USING DYNAMIC BAYESIAN NETWORKS

Jirka Poropudas  
Kai Virtanen

Systems Analysis Laboratory  
Aalto University  
School of Science and Technology  
P.O.Box 11100, FIN - 00076 Aalto, Finland

### ABSTRACT

The application of dynamic Bayesian networks (DBNs) is a recently introduced approach to simulation metamodeling where the probability distribution of simulation state is represented as a function of time. The DBN metamodels reveal the time evolution of simulation and enable alternative what-if analyses unlike previous metamodels that imitate the simulation model as an input-output mapping. In earlier studies, the analysis of DBNs is restricted to discrete time instants selected beforehand in the construction phase of the metamodel. This paper introduces an extension to the framework of DBN metamodeling that employs multivariate interpolation and allows the analysis in continuous time. In practice, an approximation for the probability distribution of the simulation state is calculated by interpolating between conditional probabilities given by the DBN. The utilization of multivariate interpolation in the context of DBN metamodeling is illustrated by examples dealing with Poisson arrival processes and air combat simulation.

### 1 INTRODUCTION

Discrete event simulation (DES) is a flexible and widely used methodology for analyzing complicated real-world systems that include stochastic elements (Law 2006). In most cases, however, conducting analysis directly with the simulation model is complicated and analyzing raw simulation data is inconvenient. Therefore, simulation data are often analyzed using simulation metamodels, i.e., auxiliary models representing the simulation model as an input-output mapping (Friedman 1996). There exists various approaches to simulation metamodeling: regression models (Blanning 1974, Kleijnen 1979), spline models (Barton 1998), artificial neural networks (Fonseca, Navarrese, and Moynihan 2003), radial basis functions (Hussain, Barton, and Joshi 2002), Kriging or spatial correlation models (Kleijnen 2009), frequency domain models (Schruben and Cogliano 1987), response surfaces as in (Barton 1998) and (Kleijnen and Sargent 2000), and game models (Poropudas and Virtanen 2009a). These metamodels are used for simplifying and interpreting simulation models, conducting sensitivity and what-if analyses (Friedman 1996), and optimizing simulation output (Cheng and Currie 2004).

The metamodels listed above are mappings between simulation input and output that overlook the changes in the simulation state during DES. This deficiency has been overcome by using dynamic Bayesian networks (DBNs) (Dean and Kanazawa 1990) as simulation metamodels allowing the study of time evolution of simulation (Poropudas and Virtanen 2007, Poropudas and Virtanen 2010). In DBN metamodels, the simulation output under consideration is a time series representing the changes in the state of DES. The DBNs are constructed to represent the time dependence of the probability distribution of the simulation state, i.e., its time evolution. In what-if analyses, DBNs are used to calculate the probability distributions conditional on a fixed value of the simulation state at a given time instant. Such analyses display the dependence between the specified simulation state and the state of the simulation at other time instants. Similar analyses could also be performed based on raw simulation data but the utilization of DBNs makes the analysis less time consuming and more efficient compared to the repeated re-screening of the data (Poropudas and Virtanen 2007). A detailed

discussion on the construction of DBN metamodels based on simulation data as well as their analysis possibilities is available in (Poropudas and Virtanen 2010).

The existing DBN metamodels operate in discrete time despite the continuous nature of DES. They are used to study the probability distribution of the simulation state only at discrete time instants that must be fixed during the construction of the DBN. Note that, specific time instants can be included in the DBN during its construction phase, if it is known in advance that they are relevant to the analysis of the simulation model. However, if the time instants of interest are not known in the construction phase, the construction procedure has to be repeated in order to focus on other time instants. Furthermore, the inclusion of extraneous time instants in the model increases its size unnecessarily.

This paper extends the analysis of DBN metamodels into continuous time. The extension is based on probabilities calculated using the DBN and multivariate interpolation. An approximation for the probability distribution of the simulation state at an arbitrary time instant is acquired by calculating the closest known probabilities with the DBN and interpolating between these values. The interpolation gives approximate results effortlessly and without a need to access simulation data or construct additional DBN metamodels.

The presented extension can be used in the analysis of continuous time simulations as well as other continuous time phenomena. Another approach for the analysis of continuous time stochastic processes would be the utilization of continuous time Bayesian networks (CTBNs) as proposed in (Nodelman, Shelton, and Koller 2002). The CTBNs can also be constructed based on data (Nodelman, Shelton, and Koller 2003) but, unfortunately, their application area is restricted to time-homogenous Markov-processes (Nodelman, Shelton, and Koller 2002). Such limitations do not apply to the DBN metamodels discussed in this paper. In addition, they offer similar analysis possibilities as the CTBNs.

This paper is structured as follows. A short introduction to construction principles of DBN metamodels is given in Section 2. In addition, the utilization of DBNs for describing the evolution of simulation and conducting what-if analyses is reviewed. Section 3 introduces the extension of DBN metamodels into continuous time by using multivariate interpolation. Then, the entire duration of the simulation can be taken into account when applying DBNs in simulation studies. Both the formulation and utilization of the interpolation are presented. The use of the extended DBN metamodels is illustrated in Section 4 where two simulation models are analyzed. Finally, conclusions are given in Section 5.

## 2 SIMULATION METAMODELING USING DYNAMIC BAYESIAN NETWORKS

### 2.1 Construction Based on Simulation Data

In a DES model, simulation state  $S(t)$  at time  $t \in [t_0, t_f]$  is defined using time dependent random variables  $x_1(t), \dots, x_n(t)$ . Now, it is assumed that each state variable  $x_k(t)$ , where  $k \in K = \{1, \dots, n\}$ , has a discrete set of values  $X_k$ . These variables are taken as the variables of DBN metamodel and simulations are performed to produce necessary simulation data, i.e., time series representing the values of the state variables during the simulation. Let us denote the probability of the value  $i \in X_k$  of the variable  $x_k(t)$ , where  $k \in K$  and  $t \in [t_0, t_f]$ , by  $p_i^k(t) := P(x_k(t) = i)$ . The probabilities  $p_i^k(t)$  as functions of time are estimated from the simulation data and now they are called probability curves.

The DBN metamodel is a discrete time model that represents the simulation state at time instants  $t \in T = \{t_0, t_1, \dots, t_f\}$ . Once finished, the constructed metamodel produces the probability distribution of the simulation state at the time instants  $T$ . To study probabilities involving time instants not included in the DBN, one has to use an approximation such as piecewise linear interpolation

$$\hat{p}_i^k(t) := p_i^k(t_-) + \frac{t - t_-}{t_+ - t_-} [p_i^k(t_+) - p_i^k(t_-)], \quad (1)$$

where  $t_- = \max\{v \in T \mid v \leq t\}$  and  $t_+ = \min\{v \in T \mid v \geq t\}$ . In order to achieve the best representation for the simulation model, the time instants  $T$  have to be selected deliberately. Thus, the construction of the DBN metamodel involves an optimization problem regarding the placement of the time instants.

In the optimization problem, the number of time instants is fixed and the time instants  $T = \{t_0, \dots, t_f\}$  are selected so that the approximation error is minimized (Poropudas and Virtanen 2010). This results

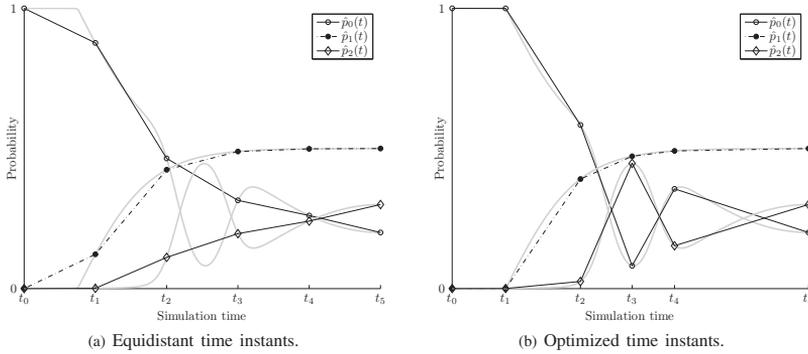


Figure 1: Grey lines present the probability curves  $p_i(t) = P(x(t) = i)$ , where  $i \in \{0, 1, 2\}$ , estimated from simulation data. The approximation of the probabilities  $\hat{p}_i(t)$  is calculated using interpolation based on the probabilities corresponding to the time instants  $t \in T = \{t_0, t_1, \dots, t_5\}$ .

in an optimization problem of the form

$$\min_T M(T), \quad \text{where} \quad M(T) := \max_{k \in K, i \in X_k, t \in [t_0, t_f]} \left| p_i^k(t) - \hat{p}_i^k(t) \right|. \quad (2)$$

The problem is solved using, e.g., a genetic algorithm (Goldberg 1989). Figures 1(a) and 1(b) illustrate the time evolution of a simulated system with one state variable  $x(t)$  and three values of the simulation state  $X = \{0, 1, 2\}$ . Note that the probability curves behave approximately linearly between the optimized time instants of the DBN. Additionally, all the discussed probabilities are estimated from the simulation data, i.e., so far there is no need for a DBN

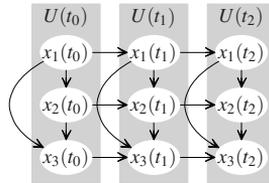


Figure 2: Example DBN metamodel representing the simulation state at time instants  $t \in T = \{t_0, t_1, t_2\}$ . The simulation state at time  $t \in T$  is represented by a time slice consisting of three simulation state variables  $U(t) = \{x_1(t), x_2(t), x_3(t)\}$ . Each node is associated with a conditional probability table. Together, the structure of the DBN and the probability tables give the joint probability distribution of the simulation state variables  $U = \bigcup_{t \in T} U(t)$ .

A DBN is a graphical presentation for the joint distribution of a sequence of random variables (see, e.g., Dean and Kanazawa 1990, Jensen and Nielsen 2007). In order to study the time evolution of the simulation, a DBN metamodel is constructed based on the simulation data to represent the joint distribution of simulation state variables during the course of the simulation. In the DBN, the simulation state at time  $t \in T$  is represented by a time slice  $U(t) = \{x_1(t), \dots, x_n(t)\}$  (see, e.g., Figure 2).

Once the variables and the optimal time instants have been chosen, the structure of the DBN is determined. The DBN consists of nodes representing the random variables  $x_k(t)$ , where  $k \in K$ , used in the definition of the simulation state. The arcs between the nodes denote dependencies between the variables. In the construction of the DBN metamodel, an initial network structure is designed by a subject matter expert and additional arcs may be added based on the dependencies found in the simulation data.

To complete the DBN metamodel, a conditional probability table is estimated from the simulation data for each node. The conditional probability tables contain the conditional probability distribution

of the corresponding variable for each combination of the values of its predecessors. Together with the network structure, the probability tables are used to describe the joint distribution for the simulation state variables  $U = \bigcup_{t \in T} U(t)$  (Pearl 1991, Jensen and Nielsen 2007).

The DBN metamodel is validated by comparing the conditional probabilities given by the DBN with probability curves estimated directly from the simulation data. The conditional probabilities calculated with the DBN should match the probability curves which can be examined visually. If no inaccuracies are detected, the DBN metamodel is taken as a valid representation for the simulation model.

In the construction, a number of available easy-to-use software (Korb and Nicholson 2004) such as Genie (Decision Systems Laboratory 2010) and Hugin (Andersen, Olesen, and Jensen 1990), can be utilized. The construction and validation of DBN metamodels are presented in full detail in Poropudas and Virtanen 2010.

## 2.2 Utilization of DBN Metamodels in Simulation Studies

In the following, it is assumed that  $S(t) = x(t)$ , i.e., only one variable is needed to define the simulation state. The DBN metamodel is constructed based on simulation data involving the values of this variable during the simulation. The DBN can be used to calculate the joint distribution of values of the variable  $x(t)$  at time instants  $t \in T$  that consists of the probabilities  $P(x(t_0) = i_0, \dots, x(t_f) = i_f)$  for the combinations of the values  $i_0, \dots, i_f \in X$ . Furthermore, the DBN produces the marginal and conditional probability distributions for the variable at various time instants. The probabilities of interest include, e.g.,  $P(x(t) = j)$ ,  $P(x(t) = j, x(s) = i)$ ,  $P(x(t) = j | x(s) = i)$ , and  $P(x(t_f) = j | x(s) = i)$ , where  $s, t, t_f \in T$  and  $i, j \in X$ .

The probabilities  $P(x(t) = j)$  depict the time evolution of the simulation state. In other words, the DBN is used to calculate the probability of the state variable having a given value  $j \in X$  at time instant  $t \in T$ . In what-if analyses, the DBN is used to determine conditional probabilities, e.g.,  $P(x(t) = j | x(s) = i)$  or  $P(x(t) = j | x(s) = i, x(r) = k)$  where  $s, t, r \in T$  and  $i, j, k \in X$ . Note that the fixed values of the random variables in the conditional probabilities correspond to observing the values of these variables. Thus, the conditional probabilities reveal the dependence between the observed simulation state and the time evolution of the simulation. On the other hand, the conditional probabilities such as  $P(x(t) = j | x(s) = i)$  can be interpreted as transition probabilities from state  $i$  at time  $s$  to state  $j$  at time  $t$ .

The simulation data can also be used to calculate probabilities representing the time evolution of the simulation, i.e.,  $P(x(t) = j)$ , as well as conditional probabilities such as  $P(x(t) = j | x(s) = i)$ . In practice, the probabilities are estimated by screening the simulation data for replications that fulfill the imposed conditions and, thus, the analysis of several alternative conditions may be time consuming. The unnecessary re-screening of the simulation data may be circumvented by using the DBN metamodel to calculate the probabilities.

Overall, the DBN metamodels expedite the analysis of the time evolution and allow the utilization of the available software tools such as GeNIe (Decision Systems Laboratory 2010) or Hugin (Andersen, Olesen, and Jensen 1990). Yet, it should be noted that there exists a limitation to the simulation studies using DBN metamodels. If the interesting time instants are not included in the DBN, i.e.,  $s, t, r \notin T$ , the DBN can not be directly used to calculate the conditional probabilities. If it is known in advance that some given time instant  $t$  is of interest, this time instant can be added as constraint to the set of time instants  $T$  in the optimization problem (2). This ensures that the said time instant is included in the constructed DBN and can be incorporated in the analysis. However, this is not an effective solution for all time instants  $t \in [t_0, t_f]$  as a new DBN would have to be constructed for each relevant time instant. In the following, this shortcoming is tackled using multivariate interpolation.

## 3 INFERENCE IN CONTINUOUS TIME USING DBN METAMODELS

### 3.1 Multivariate Interpolation

In the construction of a DBN, the location of time instants is optimized in order to find a good piecewise linear fit to probability curves. The actual probability curves are approximately linear between the time instants if there are enough time instants and they have been selected correctly. The DBN metamodel is used to calculate probabilities  $p_j(t) := P(x(t) = j)$  where  $t \in T$ , which are in the following referred to as the exact probabilities. For other time instants  $t \in [t_0, t_f]$ , approximative

probabilities are produced using interpolation. The interpolation formula presented in Eq. (1) can be re-written in a more generalizable form as a first order Lagrange interpolating polynomial (see, e.g., Phillips 2003)

$$\hat{p}_j(t) := \frac{1}{d(t_-, t_+)} \sum_{\tau=t_-}^{t_+} (d(t_-, t_+) - d(t, \tau)) p_j(\tau), \quad (3)$$

where  $t_-$  and  $t_+$  are defined as in Eq. (1) and  $d(t, \tau) := |t - \tau|$  is the distance between time instants  $t$  and  $\tau$ . This interpolation describes the time evolution of the simulated system within  $[t_0, t_f]$  and approximates the probability of a given value of the simulation state at time  $t \in [t_0, t_f]$ .

Similar reasoning is applicable also in the approximation of conditional probabilities such as  $p_{j|i}(t|s) := P(x(t) = j | x(s) = i)$  or  $p_{j|i,k}(t|s, r) := P(x(t) = j | x(s) = i, x(r) = k)$  where  $t, s, r \in [t_0, t_f]$  but not necessarily  $t, s, r \in T$ . The DBN provides the exact probabilities  $p_{j|i}(t_-|s_-)$ ,  $p_{j|i}(t_+|s_-)$ ,  $p_{j|i}(t_-|s_+)$ , and  $p_{j|i}(t_+|s_+)$  where

$$\begin{aligned} s_- &= \max\{v \in T \mid v \leq s\}, & s_+ &= \min\{v \in T \mid v \geq s\}, \\ t_- &= \max\{v \in T \mid v \leq t\}, & t_+ &= \min\{v \in T \mid v \geq t\}. \end{aligned} \quad (4)$$

Thus, the time instants in Eq. (4) are the time instants closest to  $s$  and  $t$  for which the exact probabilities can be calculated using the DBN. This calculation is efficient and takes only a fraction of the time needed to estimate the conditional probabilities from the simulation data.

An approximation for  $p_{j|i}(t|s)$ , denoted by  $\hat{p}_{j|i}(t|s)$ , is calculated using the exact probabilities produced by the DBN and a first order Lagrange interpolating polynomial with two variables (Phillips 2003)

$$\hat{p}_{j|i}(t|s) = \frac{1}{d(s_-, s_+)d(t_-, t_+)} \sum_{\sigma=s_-}^{s_+} \sum_{\tau=t_-}^{t_+} (d(s_-, s_+) - d(s, \sigma))(d(t_-, t_+) - d(t, \tau)) p_{j|i}(\tau|\sigma). \quad (5)$$

The resulting approximation is applied for inference regarding all time instants  $s \in [s_-, s_+]$  and  $t \in [t_-, t_+]$ . If time instants beyond these intervals are to be studied, the appropriate values for the closest time instants are found using Eq. (4). Then, the DBN is used to calculate the corresponding probabilities and a new interpolation is obtained using Eq. (5).

When the time instants  $s, t \in T$ , the approximation matches the exact probabilities given by the DBN, i.e.,  $\hat{p}_{j|i}(t|s) = p_{j|i}(t|s)$ . Furthermore, the bivariate interpolation in Eq. (5) reduces to the univariate interpolation in Eq. (3), if one of the variables belongs to  $T$ . For example, if  $s = s_- \in T$ , Eq. (5) gives

$$\hat{p}_{j|i}(t|s_-) := \frac{1}{d(t_-, t_+)} \sum_{\tau=t_-}^{t_+} (d(t_-, t_+) - d(t, \tau)) p_{j|i}(\tau|s_-).$$

It holds that  $\sum_{j \in X} \hat{p}_{j|i}(t|s) = 1$  for all  $i, j \in X$  and  $t, s \in [t_0, t_f]$ . This is not necessarily true for other interpolation methods such as higher order interpolating polynomials or splines (Phillips 2003) which makes their use in interpolation of probabilities undesirable.

The interpolation with two variables presented in Eq. (5) is an extension of the previously used linear interpolation to two dimensions. The interpolation is generalized in similar manner to higher dimensions in order to approximate probabilities conditional on more observed simulation states such as  $p_{j|i,k}(t|s, r)$ . The probabilities are approximated by adding more variables into the interpolation (Phillips 2003) which gives in the case of three variables

$$\begin{aligned} \hat{p}_{j|i,k}(t|s, r) &:= \frac{1}{d(s_-, s_+)d(r_-, r_+)d(t_-, t_+)} \\ &\times \sum_{\sigma=s_-}^{s_+} \sum_{\rho=r_-}^{r_+} \sum_{\tau=t_-}^{t_+} (d(s_-, s_+) - d(s, \sigma))(d(r_-, r_+) - d(r, \rho))(d(t_-, t_+) - d(t, \tau)) p_{j|i,k}(\tau|\sigma, \rho), \end{aligned}$$

where  $r_-$  and  $r_+$  are defined similarly to Eq. (4). The interpolation can also be used to approximate probabilities such as  $p_{j,k|i}(t,r|s) := P(x(t) = j, x(r) = k | x(s) = i)$ .

### 3.2 Simulation Analysis Using DBN Metamodels and Interpolation

The multivariate interpolation extends the utilization of DBN metamodels from the discrete time instants  $T = \{t_0, t_1, \dots, t_f\}$  to the time interval  $[t_0, t_f]$  describing the entire duration of the simulation. By using the multivariate interpolation, all what-if analyses discussed in Section 2.2 can be performed without restrictions to the time instants under consideration. For example, conditional probabilities such as  $p_{j|i}(t|s)$  and  $p_{j|i,k}(t|s,r)$ , where  $s, t, r \in [t_0, t_f]$  and  $i, j, k \in X$ , can be approximated in order to study dependence between simulation states at different times.

If conditional probabilities are approximated with  $n$  fixed values of state variables, the interpolation necessitates the calculation of  $2^n$  conditional probabilities using the DBN. For example, for any given pair of values  $i, j \in X$  of the simulation state variable at the given time instants  $s, t \in [t_0, t_f]$ , four probabilities are calculated using the DBN and the interpolation (1) is used to produce the approximation  $\hat{p}_{j|i}(t|s)$  for the conditional probability  $p_{j|i}(t|s)$ .

Despite the increased number of calculations compared to calculating a single probability, the combination of DBNs and multivariate interpolation is faster than alternative methods for calculating conditional probability distributions. Such methods include estimating probabilities based on simulation data or constructing a new DBN metamodel with time slices representing the time instants of interest. The repeated screening of the simulation data has proved to be time consuming (Poropudas and Virtanen 2007, Poropudas and Virtanen 2010) and the construction of a new DBN takes the same amount of computational effort as the construction of the original DBN metamodel. Thus, combining DBN metamodels with multivariate interpolation is the most effective and preferable technique for calculating the required probabilities.

## 4 EXAMPLE ANALYSES

### 4.1 Poisson-process with Time-Dependent Arrival Intensity

In the first example, the simulated system is a Poisson arrival process (Taylor and Karlin 1998) where the state of the simulation  $S(t)$  is determined by a single state variable  $x(t)$  which represents the number of arrivals up to time  $t \in [0.00, 10.00]$ . The arrival intensity  $\lambda(t)$  is time-dependent and defined as

$$\lambda(t) = \begin{cases} 0.40 & 2.00 \leq t < 6.00, \\ 1.20 & 8.00 \leq t < 10.00, \\ 0.00 & \text{otherwise.} \end{cases} \quad (6)$$

To gather the necessary simulation data, the system is simulated 10000 times. In the process under consideration, the value of the state variable has no theoretical upper limit. During the simulation, the highest observed number of arrivals was 13. Thus, in the DBN, the set of possible values of the state variable is defined as  $X = \{0, 1, \dots, 13\}$ . A DBN metamodel is constructed and validated using the principles discussed in Section 2.1. The resulting DBN is presented in Figure 3. The DBN represents the simulation state at 13 time instants. The simulation state at a given time instant depends on the previous state which is signified by the arcs between the nodes of the network.

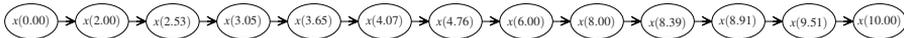
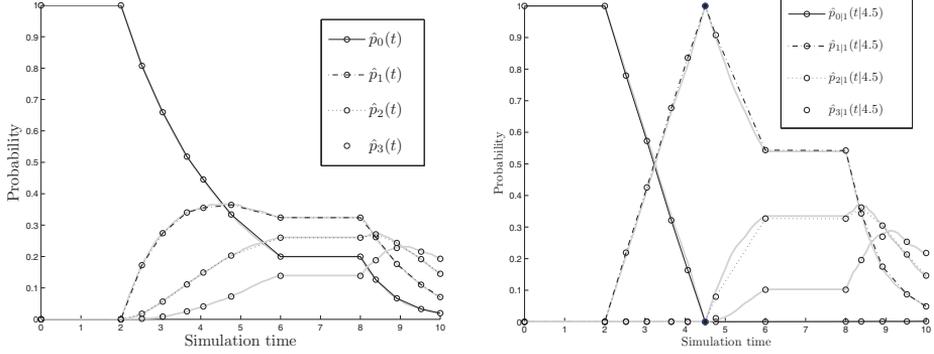


Figure 3: DBN metamodel for the Poisson process with the time-dependent arrival intensity. The DBN consists of 13 time slices representing the simulation state  $x(t)$  at the time instants  $T = \{0.00, 2.00, 2.53, 3.05, 3.65, 4.07, 4.76, 6.00, 8.00, 8.39, 8.91, 9.51, 10.00\}$ .

In the following, the DBN metamodel is utilized for studying the time evolution of the simulation as well as for conducting what-if analyses. The probabilities given by the DBN are combined with the interpolation (3) to obtain the approximations of the probabilities  $\hat{p}_j(t)$  at an arbitrary time instant. These probabilities and the respective probability curves estimated directly from the simulation data are presented in Figure 4(a) for values  $j \in \{0, 1, 2, 3\}$ . Figure 4(a) implies that the DBN recreates the probability curves accurately for the entire time interval  $[0.00, 10.00]$ . It is also noted that the probability curves remain constant in the time intervals where the arrival intensity is zero. The

probability  $\hat{p}_0(t)$  is equal to 1 at the beginning of the simulation and drops after time  $t = 2.00$  as the arrivals cumulate. Overall, the approximation gives an identical description for the progress of the simulation when compared with the original simulation data.



(a) Probabilities  $p_j(t)$  and their approximations  $\hat{p}_j(t)$  where  $j \in \{0, 1, 2, 3\}$ .

(b) Conditional probabilities  $p_{j|1}(t|4.50)$  and their approximations  $\hat{p}_{j|1}(t|4.50)$  where  $j \in \{0, 1, 2, 3\}$ . The black markers correspond to the assumption that  $x(4.50) = 1$ .

Figure 4: Probabilities and conditional probabilities for values of the simulation state in the Poisson arrival process. The grey lines depict the probability curves, the circles denote the exact probabilities given by the DBN, and the other lines denote the approximations given by the interpolation.

In what-if analysis, the state of the simulation is fixed at some time instant and the conditional evolution of the simulation is studied by updating the probability distribution of the simulation state at other time instants. For example, if it assumed that  $x(4.50) = 1$ , the conditional probabilities of the values of the simulation state are calculated to study how this condition affects the simulation. The conditional probability curves  $p_{j|1}(t|4.50)$  where  $j \in \{0, 1, 2, 3\}$  and the respective approximations  $\hat{p}_{j|1}(t|4.50)$  are presented in Figure 4(b). Note that the interpolation is based on the exact probabilities given by the DBN as well as the assumption that  $x(4.50) = 1$ . Overall, the approximations give an accurate representation for the conditional probabilities estimated from the simulation data. This fact is also demonstrated in Table 1 where the conditional probabilities  $p_{j|1}(8.50|4.50)$  estimated from the simulation data are compared with the approximations  $\hat{p}_{j|1}(8.50|4.50)$  for all values  $j \in X$ . The final row of Table 1 presents the probability  $p_1(4.50)$  and its approximation. The comparison of the conditional probability distributions implies that the interpolation gives a good approximation for the simulation data even though the time instants 4.50 and 8.50 do not belong to  $T$ .

Table 1: Probabilities  $p_{j|1}(8.50|4.50)$ , where  $j \in \{0, 1, \dots, 13\}$ , calculated based on the simulation data and the corresponding approximations  $\hat{p}_{j|1}(8.50|4.50)$  given by the DBN.

value $j$	Simulation data $p_{j 1}(8.50 4.50)$	Approximation $\hat{p}_{j 1}(8.50 4.50)$
0	0.000	0.001
1	0.293	0.307
2	0.367	0.350
3	0.219	0.211
4	0.086	0.088
5	0.026	0.028
6	0.001	0.003
7	0.001	0.001
$\geq 8$	0.000	0.001
$p_1(4.50)$	0.366	0.356

There is a slight non-linearity in the conditional probability  $\hat{p}_{2|1}(t|4.50)$  between the time instants 4.50 and 6.00 (see, Figure 4(b)). The non-linearity makes the approximation of the conditional probabilities within this time interval less accurate which is also seen in Table 2 where the conditional

probability curves  $p_{j|1}(5.00|4.50)$  are compared with the approximation  $\hat{p}_{j|1}(5.00|4.50)$ . Nevertheless, even in this case, the probabilities given by the interpolation correspond quite well with the probabilities estimated from the simulation data. Thus, one can conclude that the presented interpolation yields a good approximation with little computational effort.

Table 2: Probabilities  $p_{j|1}(5.00|4.50)$ , where  $j \in \{0, 1, \dots, 13\}$ , calculated based on the simulation data and the corresponding approximations  $\hat{p}_{j|1}(5.00|4.50)$  given by the DBN.

value $j$	Simulation data $p_{j 1}(5.00 4.50)$	Approximation $\hat{p}_{j 1}(5.00 4.50)$
0	0.000	0.000
1	0.816	0.838
2	0.167	0.127
3	0.016	0.029
4	0.001	0.005
5	0.000	0.001
$\geq 6$	0.000	0.001
$p_1(4.50)$	0.366	0.356

#### 4.2 Air Combat Simulation

In the second example, a DBN metamodel is used to analyze simulation data produced by simulating 1 vs. 1 air combat with a DES model called X-Brawler (L-3 Communications Analytics Corporation 2002). The sides of the air combat are named blue and red. The state of the simulation at time  $t$  is described by two variables  $S(t) = \{x_1(t), x_2(t)\}$ . The variables  $x_1(t)$  and  $x_2(t)$  are indicator variables for "blue is alive at time  $t$ " and "red is alive at time  $t$ ", respectively. As the state variables are binary, the set of their values is  $X = \{0, 1\}$ . The state variables describe the state of the air combat as

- *neutral*, i.e., both sides are alive,  $x_1(t) = 1$  and  $x_2(t) = 1$
- *blue advantage*, i.e., blue is alive and red has been killed,  $x_1(t) = 1$  and  $x_2(t) = 0$
- *red advantage*, i.e., blue has been killed and red is alive,  $x_1(t) = 0$  and  $x_2(t) = 1$
- *mutual disadvantage*, i.e., both sides have been killed,  $x_1(t) = 0$  and  $x_2(t) = 0$

The maximum duration of the simulation is set as 500 (seconds). The combat is simulated 10000 times in order to gather the necessary simulation data. Then, a DBN metamodel for the simulated air combat is constructed and validated as in Section 2.1. The final DBN is presented in Figure 5. The structure of the DBN indicates that the values of the state variables depend on the previous values of both variables which is designated by the arcs in Figure 5. The additional dependencies found in the simulation data are denoted by white arrowheads.

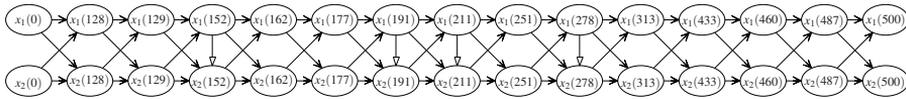


Figure 5: DBN metamodel for the simulated air combat. The DBN consists of 15 time slices representing the simulation state at the time instants  $T = \{0, 128, 129, 152, 162, 177, 191, 211, 251, 278, 313, 433, 460, 487, 500\}$ .

Once constructed, the DBN metamodel is utilized for analyzing the time evolution of the air combat and conducting what-if analyses. The time evolution is studied by considering probabilities  $p_{(j_1, j_2)}(t) := P(x_1(t) = j_1, x_2(t) = j_2)$ , where  $j_1, j_2 \in X$ , as a function of time. Figure 6(a) presents the probability curves  $p_{(j_1, j_2)}(t)$  and the exact probabilities given by the DBN as well as the corresponding approximations  $\hat{p}_{(j_1, j_2)}(t)$ . The approximations coincide with the probability curves, i.e., they lead to similar inference and produce an identical portrayal of the progress of the simulation.

The simulation begins in the *neutral* state, i.e., at the beginning of the simulation both sides are alive (see, Fig 6(a)). After 128 seconds, the probability of *blue advantage* rises as blue has an opportunity to shoot down red. Then, after  $t = 162$  also red may have shot down its opponent and the probability of *red advantage* increases. After 278 seconds the air combat is over and the probabilities of the

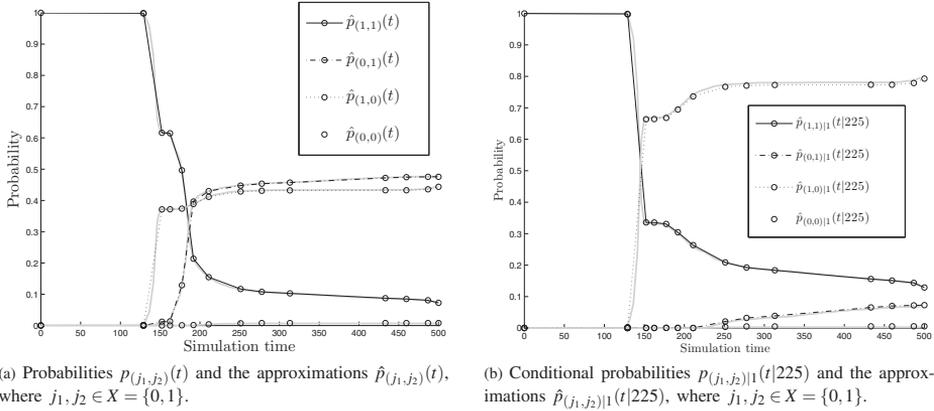


Figure 6: Probabilities and conditional probabilities for the state of the air combat as a function of time. The grey lines depict the probability curves, the circles denote the exact probabilities given by the DBN, and the other lines denote the approximations given by the interpolation.

states of the simulation remain approximately constant. At the end of the simulation, the probability of *blue advantage* is 0.444, *red advantage* 0.476, *mutual disadvantage* 0.008, and *neutral* 0.073. In this way, the progress of the simulation may be studied effortlessly using the probabilities given by the DBN.

Next, what-if analysis is conducted in order to study dependencies between the values of the simulation state at different time instants. For example, one can assume that blue is alive at time instant  $t = 225$ , i.e.,  $x_1(225) = 1$ . Then, the DBN is used with interpolation to calculate the approximations of the conditional probabilities  $\hat{p}_{(j_1, j_2)|1}(t|225)$  for time instants  $t \in [0, 500]$  and  $j_1, j_2 \in X$ . The approximations are presented in Figure 6(b) alongside the conditional probability curves estimated from the simulation data. In this case, the interpolation produces a good match with the probability curves.

The conditional probabilities in Figure 6(b) show that the initial jump in the probability of *blue advantage* is almost twice as large compared to the unconditioned situation in Figure 6(a), i.e., the assumption that blue is alive at time  $t = 225$  increases also the probability of blue shooting down red at the beginning of the air combat. On the other hand, the probability of *red advantage* remains almost nil up to time  $t = 251$  as blue is assumed to be alive at time  $t = 225$ . Additionally, the probability of *red advantage* remains low throughout the remainder of the simulation, i.e., if red has not shot blue down by this time instant, it is unlikely to do so in the following air combat. The assumption that blue is alive at time  $t = 225$  alters the probability distribution of the final simulation state so that the probability of *blue advantage* is 0.792, *red advantage* 0.073, *mutual disadvantage* 0.006, and *neutral* 0.129. Overall, the application of the DBN eases the calculation of conditional probabilities and gives an effective way for conducting what-if analyses.

Finally, the DBN is applied for what-if analysis regarding the probability distribution of the final simulation state. Now, the value of  $x_1(s)$  is set equal to 1, i.e., it is assumed that blue is alive at time  $s \in \{125, 150, 175, 200, 225, 250, 275, 300\}$ . Then, the conditional probability distribution of the final simulation state is calculated. Table 3 presents both the conditional probabilities  $p_{(j_1, j_2)|1}(500|s)$  estimated based on the simulation data and the respective approximations  $\hat{p}_{(j_1, j_2)|1}(500|s)$  given by the DBN. The table also includes the probabilities  $p_{(1, \cdot)}(s) := P(x_1(s) = 1)$  and their approximations  $\hat{p}_{(1, \cdot)}(s)$ . The both probabilities imply that the probability of the survival of blue is a decreasing function of time. It is also evident that the approximation concurs with the probabilities estimated from the simulation data. The final row of Table 3 displays the unconditional probability distribution of the final simulation state. The conditional probabilities  $\hat{p}_{(1, 0)|1}(500|s)$  and  $\hat{p}_{(1, 1)|1}(500|s)$  increase with the value of  $s$ . Thus, the longer blue survives, the more favorable it finds the probability distribution of the final simulation state. If blue survives the initial 300 seconds of the simulation, it has a 0.962 probability of surviving the air combat. Furthermore, if blue survives up to the time instant  $s = 300$ ,

Table 3: Probability distribution of the final air combat state  $p_{(j_1, j_2)|1}(500|s)$ , where  $j_1, j_2 \in X = \{0, 1\}$ , conditional on blue being alive at the time instants  $s \in \{125, 150, 175, 200, 225, 250, 275, 300\}$  estimated based on the simulation data. The approximations  $\hat{p}_{(j_1, j_2)|1}(500|s)$  are calculated using the DBN and interpolation. The final row represents the unconditioned probability distribution of the final state.

		Simulation data				Approximation				
		$p_{(j_1, j_2) 1}(500 s)$				$\hat{p}_{(j_1, j_2) 1}(500 s)$				
$s$	$p_{(1, \cdot)}(s)$	(1,0)	(0,1)	(0,0)	(1,1)	$\hat{p}_{(1, \cdot)}(s)$	(1,0)	(0,1)	(0,0)	(1,1)
125	1.000	0.442	0.477	0.007	0.074	1.000	0.444	0.476	0.008	0.073
150	0.989	0.447	0.471	0.007	0.075	0.989	0.449	0.471	0.008	0.073
175	0.898	0.493	0.418	0.007	0.083	0.888	0.500	0.410	0.008	0.082
200	0.580	0.763	0.101	0.008	0.128	0.587	0.755	0.114	0.009	0.123
225	0.556	0.796	0.068	0.003	0.134	0.559	0.792	0.073	0.006	0.129
250	0.546	0.811	0.052	0.001	0.136	0.545	0.811	0.054	0.003	0.132
275	0.539	0.820	0.042	0.000	0.138	0.539	0.822	0.043	0.001	0.134
300	0.536	0.825	0.037	0.000	0.139	0.536	0.827	0.037	0.001	0.135
-	-	0.442	0.477	0.007	0.074	-	0.444	0.476	0.008	0.073

the conditional probability  $\hat{p}_{(0,1)|1}(500|300)$  is only 0.037, i.e., the probability of the simulation ending in state *red advantage* is small. Overall, Table 3 illustrates that the DBN metamodel is an accurate representation of the simulation model as the probability distributions given the interpolation and estimated from the simulation data are nearly identical. Therefore, the conditional evolution of the simulation narrated by the approximated probabilities is consistent with the simulation data even though the discussed time instants do not belong to  $T$ .

### 5 CONCLUSIONS

By using DBN metamodels, the time evolution of DES can be analyzed in a transparent manner. The DBN metamodeling concept has also proved to be an effective way for conducting various what-if analyses. The previously proposed analysis capabilities of DBNs are confined into discrete time, i.e., the analysis of only a discrete set of time instants has been attainable. In this paper, the combination of DBN metamodeling and multivariate interpolation is applied to describe the time evolution of DES. This extension of DBN metamodeling utilizes first order Lagrange interpolating polynomials for approximative inference in continuous time. The paper illustrated the combination of DBNs and interpolation with two example analyses. According to these examples, the interpolation method discussed in this paper gives a simple and intuitive approximation for the probability distribution of the simulation state. The approximation can be calculated efficiently and it provides an accurate representation for the original simulation data. Thus, when compared with the brute force analysis of the simulation data, the proposed metamodeling concept leads to more effective inference with similar results and implications.

Simulation studies using DBN metamodels can be performed with software designed for the analysis of Bayesian networks. Unfortunately, multivariate interpolation is beyond the scope of such software and, thus, interpolation calculations presented in this paper have been carried out using MATLAB (Mathworks 2010). In order to promote and alleviate future studies, it might be worthwhile to develop an automated tool designed for construction and utilization of DBN metamodels together with multivariate interpolation.

In the earlier Bayesian networks literature, stochastic continuous time phenomena have been studied using CTBNs that are DBNs describing processes that evolve over continuous time. The presented multivariate interpolation schema extends their application area of discrete time DBNs into continuous time and offers a potential alternative for CTBNs. In future, the performance of DBNs and interpolation could be compared with CTBNs constructed from data that is produced by simulation of a suitable system, viz., a time-homogenous Markov process. This comparison would give a more comprehensive understanding about the advantages and disadvantages of these alternative approaches. However, one should note that, unlike CTBNs, the proposed combination of DBNs and interpolation is not limited to time-homogenous Markov-processes.

The DBN metamodels have also been used in simulation-based optimization as a part of influence diagram metamodels (Poropudas and Virtanen 2009b). In such metamodels, the DBN reveals the consequences of decision alternatives, i.e., the time evolution of a simulated system at discrete

time instants with given values of simulation parameters. Clearly, the application of multivariate interpolation could also be applied for representing the effects of simulation parameters within the simulation-optimization framework in continuous time.

To summarize, the DBN metamodels present the time evolution of simulation which can not be analyzed using other simulation metamodels. The DBNs expedite simulation analyses as they remove the need for repetitive re-screening of simulation data in the estimation of conditional probabilities. The computational advantages of the DBNs enable various alternative what-if analyses providing additional insight into the behavior of the simulated system. The addition of multivariate interpolation to the application of DBN metamodels eliminates the limitation of discrete time and allows one to study the progress of simulation in continuous time.

## REFERENCES

- Andersen, S. K., K. G. Olesen, and F. V. Jensen. 1990. *Hugin – a shell for building Bayesian belief universes for expert systems*. San Francisco, CA: Morgan Kaufmann.
- Barton, R. R. 1998. Simulation metamodels. In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. Medeiros, E. Watson, J. Carson, and M. Manivannan, 167–174. Washington, DC: IEEE Computer Society Press.
- Blanning, R. W. 1974. The sources and uses of sensitivity information. *Interfaces* 4 (4): 32–38.
- Cheng, R. C., and S. S. Currie. 2004. Optimization by simulation metamodelling methods. In *Proceedings of the 2004 Winter Simulation Conference*, ed. R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, 473–478. Washington, DC: IEEE Computer Society Press.
- Dean, T., and K. Kanazawa. 1990. A model for reasoning about persistence and causation. *Computational Intelligence* 5 (3): 142–150.
- Decision Systems Laboratory 2010. GeNIe (graphical network interface). Available via <<http://genie.sis.pitt.edu/>> [accessed March 22, 2010].
- Fonseca, D. J., D. O. Navarrese, and G. P. Moynihan. 2003. Simulation metamodeling through artificial neural networks. *Engineering Applications of Artificial Intelligence* 16 (3): 177–183.
- Friedman, L. W. 1996. *The simulation metamodel*. Norwell, MA: Kluwer Academic Publishers.
- Goldberg, D. E. 1989. *Genetic algorithms in search, optimization, and machine learning*. Upper Saddle River, NJ: Addison-Wesley Professional.
- Hussain, M., R. Barton, and S. Joshi. 2002. Metamodeling: Radial basis functions, versus polynomials. *European Journal of Operational Research* 138 (1): 142–154.
- Jensen, F. V., and T. D. Nielsen. 2007. *Bayesian networks and decision graphs*. New York, NY: Springer-Verlag.
- Kleijnen, J. P. C. 1979. Regression metamodels for generalizing simulation results. *IEEE Transactions on Systems, Man, and Cybernetics* 9 (2): 93–96.
- Kleijnen, J. P. C. 2009. Kriging metamodeling in simulation: A review. *European Journal of Operational Research* 192 (3): 707–716.
- Kleijnen, J. P. C., and R. G. Sargent. 2000. A methodology for fitting and validating metamodels in simulation. *European Journal of Operational Research* 120 (1): 14–29.
- Korb, K. B., and A. E. Nicholson. 2004. *Bayesian artificial intelligence*. Boca Raton, FL: Chapman & Hall/CRC.
- Law, A. 2006. *Simulation modeling and analysis*. Fourth ed. New York, NY: McGraw-Hill Science/Engineering/Math.
- Mathworks 2010. MATLAB. Available via <<http://www.mathworks.com/>> [accessed March 22, 2010].
- L-3 Communications Analytics Corporation 2002. *The X-Brawler air combat simulator management summary*. Vienna, VA: L-3 Communications Analytics Corporation.
- Nodelman, U. D., C. R. Shelton, and D. Koller. 2002. Continuous time bayesian networks. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, 378–387. Acapulco, Mexico.
- Nodelman, U. D., C. R. Shelton, and D. Koller. 2003. Learning continuous time bayesian networks. In *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence*, 451–458. Edmonton, Canada.
- Pearl, J. 1991. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Mateo, CA: Morgan Kaufmann.
- Phillips, G. M. 2003. *Interpolation and approximation by polynomials*. New York, NY: Springer-Verlag.

- Poropudas, J., and K. Virtanen. 2007. Analyzing air combat simulation results with dynamic Bayesian networks. In *Proceedings of the 2007 Winter Simulation Conference*, ed. S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 1370–1377. Washington, DC.
- Poropudas, J., and K. Virtanen. 2009a. Game theoretic validation and analysis of air combat simulation models. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*. 40 (5): 1057–1070.
- Poropudas, J., and K. Virtanen. 2009b. Influence diagrams in analysis of discrete event simulation data. In *Proceedings of the 2009 Winter Simulation Conference*, ed. M. D. Rossetti, R. R. Hill, and B. J. 696–708. Austin, TX.
- Poropudas, J., and K. Virtanen. 2010. Simulation metamodeling through dynamic Bayesian networks. Submitted for publication. Available via <<http://www.sal.hut.fi/Publications/m-index.html>> [accessed March 16, 2010].
- Schruben, L. W., and V. J. Cogliano. 1987. An experimental procedure for simulation response surface model identification. *Communication of the Association for Computing Machinery* 30 (8): 716–730.
- Taylor, H. M., and S. Karlin. 1998. *An introduction to stochastic modeling*. Third ed. San Diego, CA: Academic Press.

#### AUTHOR BIOGRAPHIES

**JIRKA POROPUDAS** received his M.Sc. degree in systems and operations research from the Helsinki University of Technology, Espoo, Finland, in 2005. He is currently working on his doctoral thesis at the Systems Analysis Laboratory in the Aalto University School of Science and Technology, Espoo, Finland. His research interests include statistics, simulation, simulation metamodeling, and statistical analysis of basketball. His e-mail address is <[Jirka.Poropudas@tkk.fi](mailto:Jirka.Poropudas@tkk.fi)>.

**KAI VIRTANEN** received the M.Sc. and Dr. Tech. degrees in systems and operations research from the Helsinki University of Technology (HUT), Espoo, Finland, in 1996 and 2005, respectively. He is currently Adjunct Professor at the Systems Analysis Laboratory in the Aalto University School of Science and Technology, Espoo, Finland. His research interests include optimization, decision and game theory with particular attention to aerospace applications as well as discrete-event simulation. He is the author of about 40 publications in scientific journals and conferences on these fields. His e-mail address is <[Kai.Virtanen@tkk.fi](mailto:Kai.Virtanen@tkk.fi)>.