Proxel-Based Simulation: Theory and Applications

Claudia Krull[1], Graham Horton[2]

Abstract

Discrete stochastic models are widely used to describe current engineering and logistics problems. The stochastic simulation of such models can get very expensive if the models are stiff or rare system events are of interest. Proxels are a state space-based simulation technique that does not have these drawbacks. They implicitly use a discrete-time Markov chain to deterministically discover all possible system states at discrete points in time. Some applications have shown that Proxels are especially suitable for the analysis of small stiff models, and can outperform stochastic simulation techniques in that area.

1 Introduction

Discrete stochastic models can be used to describe some current problems in the industry. Their analysis is often performed using discrete event-based simulation (DES). Unfortunately, DES can get very expensive. When stiff models and rare events are involved, many replications are required to gain statistically meaningful results. The performance of DES is dependent on the degree of stiffness of the model or rareness of the event of interest. Existing methods for rare event simulation try to relieve that by modifying either the model or the problem specification. However, these methods can be very complex and are usually problem dependent in their application. Proxel-based simulation is a recently developed state space-based simulation approach, which is based on discrete-time Markov chains (DTMC). It is a deterministic algorithm and does not suffer a significant performance decrease when rare events are involved. Proxels are especially suitable for the simulation of small stiff models, discovering all possible system developments in one run and assigning them probabilities. In contrast to partial or ordinary differential equations, Proxels are more intuitive to use and not inherently limited to specific model classes. Using a generic implementation, Proxels can in principle be applied to any discrete stochastic model, instead of stochastic simulation techniques. The paper describes the basic idea of Proxels, two successful applications and some current extensions.

[1]Otto-von-Guericke-University Magdeburg, E-mail: claudia@sim-md.de
[2]Otto-von-Guericke-University Magdeburg, E-mail: graham@sim-md.de

2 State of the Art

2.1 Stochastic Simulation of Rare Event Models

The stochastic simulation of models involving rare events can become unfeasibly expensive. Many replications are needed to discover the rare events and even more to obtain statistically significant results for them. In general, the cost of a DES is dependent on the number of state changes performed per simulation run and on the number of simulation runs necessary. This implies that the cost increases with increasing degree of rareness of the event. Rare event simulation methods try to relieve this problem. Importance sampling modifies the model definition by changing transitions specifications to make the event of interest more frequent. Importance splitting defines intermediate thresholds that have to be crossed before reaching the rare event. Development paths are split at these thresholds in order to increase the number of times a rare event is encountered within the simulation runs. Both methods require a subsequent rescaling of the results to make them applicable to the original model. However, both methods are mathematically complex and usually require problem knowledge to be applied properly. Their performance still suffers somewhat when the degree of rareness of the event increases.

2.2 Discrete-Time Markov Chains

Discrete-time Markov chains (DTMC) are a well researched area of mathematical modeling (see [1] for a thorough introduction). They can represent the state space of a model including the state transitions defined by one step transition probabilities. If one can build a DTMC representing a models states and behavior, then the solution of that DTMC is comparably easy using existing algorithms. However, the state transitions in a DTMC are memoryless, they can only directly represent discretized exponential or geometric distributions. Continuous non-Markovian distributions, such as Normal, Weibull or Lognormal, cannot be represented directly in a Markov chain. A direct DTMC representation of a real system involving time dependent behavior is often not detailed enough to draw conclusions about the systems dynamics. The easy solution of a Markov chain comes at the expense of loosing details of the time dependent system behavior. This limits the applicability of DTMC solutions to problems were the exact dynamic system behavior is not of much importance.

3 Proxel Background and Theory

One approach applying the advantages of Markov chains to the analysis of non-Markovian models are supplementary variables, which extend a system state by logging the age of that state. [2] This leads to partial differential equations (PDE) as system description that can then be solved. Extending this idea, it is possible to make any process memoryless by logging the ages of all currently activated or relevant transitions. Doing this at discrete points in time enables to use an algorithmic approach, rather than setting up and solving PDEs analytically. Using supplementary variables to make all non-Markovian processes of a model memoryless is the basic idea of the Proxel-based simulation method [3, 6].

A Proxel as defined in Equation (1) is a point S in the extended state space of the model – the discrete system state dS extended by the age of the relevant transitions $\vec{\tau}$ for a specific point in the simulation time t – , with the probability of that state p. Proxels are only generated at discrete points in time, which are multiples of the simulation time step. The probability to perform any active state change within one of these time steps can the be determined by the so-called instantaneous rate function (IRF), which is defined as in Equation (2).

$$P = (S,p) = ((dS,\vec{\tau},t),p) \quad (1)$$

$$\mu(\tau) = \frac{f(\tau)}{1-F(\tau)} \quad (2)$$

The following is a sketch of the Proxel-based simulation approach based on these ideas. `start` represents the initial system state, and `dt` the discrete simulation time step.

```
1  Create initial Proxel with initial system state
   at start of simulation time $((start,0,0),1)$
2  For each activated transition r of each Proxel at time t
3    Create Proxel for t+dt with the probability
     of state change r within dt, reset transition age r
4    Create Proxel for t+dt for the case of no state change
     with leftover probability, increase transition ages by dt
5    Store newly created Proxels in data structure
6  Repeat 2-5 until end of simulation time
```

The algorithm implicitly builds a DTMC of the reachable model state space. By extending the discrete system states by the transition activation times, all processes are made memoryless. This enables to determine a transient solution of the discrete stochastic model algorithmically.

The performance of the exact implementation largely depends on the data structure chosen for Proxel storage. In general, the Proxel approach is much more flexible than the original supplementary variables, because it does not require to setup and solve differential equations. In contrast to DES, the cost of the method is not influenced by the stiffness of the model. The algorithm deterministically discovers all possible states at discrete points in time. The smaller the simulation time step is, the more accurate these probabilities are. On the other hand, the simulation time step also determines the cost of the simulation. This enables a trade-off between accuracy and computation cost. Some further features and problems of Proxel-based simulation will be discussed in Section 5.

4 Two Example Applications of Proxels

This section describes two example applications of Proxel-based simulation to small stiff models. These nicely demonstrate the properties of Proxel-based simulation and exemplify the area where Proxels can outperform existing simulation methods.

4.1 Analysis of a Vehicle Warranty Model

The problem described here comes from an industry project carried out for the DaimlerChrysler AG (DC). [7] The task was to determine costs of different warranty strategies for the following scenario: The expiration of a car warranty is based on a maximum mileage and a maximum time (e.g. 10000 miles vs. 1 year). Failures within the warranty period incur costs for the manufacturer. The failure has a much smaller rate than the warranty expiration, however, the occurrence of a failure generates considerable cost. Therefore we are dealing with a stiff model.

The DES employed by DC needed a runtime of 20 to 30 hours to compute a cost estimate to the accuracy of one cent for one parameter set (years, mileage, cost per failure), using given failure and time to mileage distributions. The special-purpose Proxel-based algorithm developed for this case used mileage as the basic time unit and one mile as discrete simulation time step. The failure probability within the warranty period multiplied by the cost of a failure directly yielded the desired warranty cost. This approach was already quite fast, needing only few minutes to obtain a comparable result. In a second attempt, rough estimates obtained for larger simulation time steps were used to extrapolate a more accurate solution. This was possible because of a linear convergence of the solution parameter with decreasing simulation time step. This decreased the computation time to mere seconds for one parameter set. This application enabled DC to gain faster and more precise predictions for the warranty costs in only a fraction of the original time, eventually enabling a faster decision between warranty strategies.

4.2 Proxel-Based Queuing Simulation

Queuing analysis is an old subject in modeling and simulation. [1] It has recently become of interest again, since many problems in electronic communication can be described using queuing models. The goal of classical queuing analysis is to find an analytical expressions for the performance measures of a class of queuing systems. However, this is not always possible, depending on the system specification. As an alternative, DES can be employed, even though it is a lot less accurate and more expensive. Proxels can be a good alternative to DES, when no analytical solution is available. They are especially suitable for queuing simulation, because the discrete state space of a queuing model is usually small and the number of processes is limited to arrival and service of customers. Furthermore, queuing models can be very stiff, or rare system states are of interest, such as the overflow of a buffer in a switch and the resulting packet loss.

An example queuing system of a small call center is of type M/G/c/K. The exact problem specification is $Exp(1.25)/N(1;0.2)/2/17$ with a Markovian arrival process, a normally distributed service process, two call-center agents as servers and a holding queue capacity of 15, system capacity of 17. The rare event of interest is the filling up of the queue and the resulting possibility to loose incoming calls. The Proxel solution needed only seconds to produce a meaningful result for the queue overflow probability. In contrast, a discrete event-based simulation of the system needed 15 minutes of computation time. The event of the queue filling up did not happen often enough, resulting in an inappropriate confidence interval for the measure. The problem was by far to stiff for standard DES. See [5] for more details and examples.

For queuing simulation in general Proxels can be used to obtain exact results for analytically not tractable systems. They can also provide answers for systems that cannot be tackled using DES. Furthermore, Proxels can help obtain rough estimates for not yet formally analyzed problems.

5 Special Issues

This section discusses special issues and problems of the Proxel-based simulation method, as well as extensions that were already performed to reduce these problems. One major drawback of Proxel simulation and state space-based methods in general is the drastic increase in the number of system states due to the extension with supplementary variables. This so-called state space explosion limits the applicability of the methods to models with a small discrete state space. The effects of this state space explosion can be dampened somewhat by intelligent storage and retrieval strategies for Proxels. Two more fundamental strategies to tackle that problem have been implemented so far and will be described here briefly.

The first problem leading to state space explosion is that every continuous distribution is split into as many separate time steps as the support of the distribution needs, also covering very smooth parts with too many sampling points. Each one of those sampling points leads to a different age value and consequently to a separate Proxel that needs to be stored and processed. One solution to this is the combination with discrete phase-type distributions (DPH). [4] These can represent smooth distribution functions with much less sampling points, leading to a drastic reduction in the size of the expanded state space. This increases the size of the models that can be feasibly analyzed using Proxels. The combination of Proxels and DPH is possible because both are ways to represent a non-Markovian distribution with a segment of a DTMC.

The second problem leading to state space explosion is related to stiff models, because using the original algorithm, the fastest model transition determines the size of the time step that is used to discretize all distributions. If the model is stiff, this time step needs to be very small, and is inefficient for much slower transitions. The use of so-called variable time steps can help relieve that problem. [8] Here, every transition can be performed using a time step of optimal size. This strategy can reduce the computation cost for stiff models significantly, again enabling the analysis of larger models using Proxels.

The cost of a Proxel-based simulation algorithm increases with an increasing discrete state space and increasing number of concurrently activated transitions. It also increases with decreasing simulation time step size, leading to the above mentioned state space explosion, but on the other hand it also enables a trade-off between accuracy and cost of a computation. An extrapolation of the simulation results that were computed using larger time steps can be used to obtain more accurate results while reducing computation time. Summing up, current extensions and special purpose implementations of Proxels make the simulation method applicable to a significant group of real world problems.

6 Conclusion

Proxel-based simulation is a state space-based method well suitable for the analysis of small stiff models or models containing rare events. Two specific applications were described, demonstrating that. In contrast to DES and current methods for Rare event simulation, Proxels can deterministically discover all possible system states in one run and assign probabilities to them. The state space explosion inherent to this class of approaches limits the applicability to small models. However, it can be dampened somewhat through the use of discrete phase-type distributions or variable time steps. Proxels can be used to obtain accurate results in a limited computation time for some problems, where DES cannot be feasibly applied.

References

[1] Bolch G., Greiner S., de Meer H., Trivedi K. S. (1998) *Queuing Networks and Markov Chains*. John Wiley & Sons, New York.

[2] German R., Lindemann C. (1994) Analysis of stochastic petri nets by the method of supplementary variables. In *Proceedings of Performance Evaluation*, 20:317-335.

[3] Horton G. (2002) A new paradigm for the numerical simulation of stochastic petri nets with general firing times. In *Proceedings of the European Simulation Symposium 2002*, pp.129-136. SCS European Publishing House.

[4] Krull C. (2008) *Discrete-Time Markov Chains: Advanced Applications in Simulation*. PhD thesis, Otto-von-Guericke-University Magdeburg.

[5] Krull C., Horton G. (2007) Application of proxels to queuing simulation. In *Simulation and Visualization 2007*, Magdeburg, Germany, pp.299-310.

[6] Lazarova-Molnar S. (2005) *The Proxel-Based Method: Formalisation, Analysis and Applications*. PhD thesis, Otto-von-Guericke-University Magdeburg.

[7] Lazarova-Molnar S., Horton G. (2004) Proxel-based simulation of a warranty model. In *European Simulation Multiconference 2004*, pp.221-224. SCS European Publishing House.

[8] Wickborn F., Horton G. (2005) Feasible state space simulation: Variable time steps for the proxel method. In *Proceedings of the 2nd Balkan conference in informatics*, Ohrid, Macedonia, pp.446-453.