Micro and macro views of discrete-state Markov models and their application to efficient simulation with Phase-type distributions

Philipp Reinecke
philipp.reinecke@fu-berlin.de
Freie Universität Berlin
Institute for Computer Science
Takustraße 9
14195 Berlin, Germany

Miklós Telek
telek@hit.bme.hu
Budapest University of Technology and Economics
Dept. of Telecommunications
Magyar tudósok krt. 2
1117 Budapest

Katinka Wolter
katinka.wolter@fu-berlin.de
Freie Universität Berlin
Institute for Computer Science
Takustraße 9
14195 Berlin, Germany

1. DESCRIPTION OF MATERIAL

This proposal is for a three-hour tutorial. The tutorial is split into two parts of equal length. We first talk about the distinction of the macro and micro view in modelling and illustrate its use in efficient solution methods. We then provide a hands-on introduction to using phase-type simulations in discrete-event simulation, as one instance of using the distinction between the micro and macro view.

1.1 Micro and macro views of discrete state Markov models

A wide range of complex real life systems are modelled with Markov chains in order to evaluate various performance measures. There are sophisticated methodologies to compose the Markov models based on the elementary behavior of the components of the real system.

Examples of elementary behavior in a large system model include service, failure time distribution, arrival process of customers, etc. In Markov models these model elements are described with small Markov chains and the overall system model is composed by an appropriate combination of the small Markov chains.

With respect to evaluation, we may distinguish between two perspectives. If the elements of the underlying small Markov chains are known, we refer to this view as the *micro-level view* of the process. On the other hand, if we consider only the stochastic behaviour of the model elements, we call this perspective the *macro-level*. Due to the fact that the typical performance measures are related to the macro-level view of the system behaviour one can try to optimise the system representation by properly tuning the underlying small Markov chains. One interesting consequence of this optimisation procedure is that non-Markovian models can be used to compute Markovian systems in an efficient way.

The first part of the tutorial summarises the typical model elements used in discrete state system description, including phase-type distributions and Markov arrival processes. After that, it focusses on the problem of different representations. We list a few available results and a set of related open problems [3, 2, 4].

1.2 Efficient Simulation with Phase-type Distributions

Phase-type (PH) distributions are a valuable tool for representing real-world phenomena such as failure-times or response-times in an analytically tractable way. Recently, the application of phase-type distributions in simulation has received increasing attention. In simulation, phase-type distributions enable good representation of empirical distributions, even if the data does not follow one of the well-known statistical distributions. Furthermore, since phase-type distributions have Markovian representations, analytical approaches can be used to support simulation results.

In the theoretical communities, PH distributions are well-known and have been applied to derive analytical solutions in system evaluation for a long time. In contrast, they have seen little application in simulation. This is certainly partly due to the fact that the theoretical details of deriving and using PH distributions may appear rather intimidating to 'simulation guys'. A second reason may be the performance costs of generating PH-distributed random variates. Finally, the absence of support for PH distributions in popular discrete-event simulators such as OMNeT++ [10] and NS-2 [11] also constitutes a major hurdle.

In the second part we give a hands-on introduction to using PH distributions. We follow the workflow depicted in Figure 1: Given a measurement trace, a PH distribution is fitted to this data and used in simulation. We start with an accessible discussion of the essential theoretical basics. In this part we focus particularly on structural properties that are suitable for efficient fitting and simulation later on.

In the second step of this part we address the question of obtaining a PH distribution that fits a measurement trace. We focus on three important points: First, the approximating distribution must represent the data set well, second, the fitted distribution must be usable in efficient simulation, and third, fitting should be user-friendly. We will discuss several established fitting tools and illustrate how to obtain a good fit using our Hyper-* fitting tool for user-friendly and efficient cluster-based PH fitting [7].

The third step focusses on efficient random-variate generation from PH distributions. We introduce several algorithms and show how special structures can reduce the cost of random-variate generation [9]. We then discuss optimisation of PH representations with respect to simulation cost. We give results on optimal representations for the APH sub-

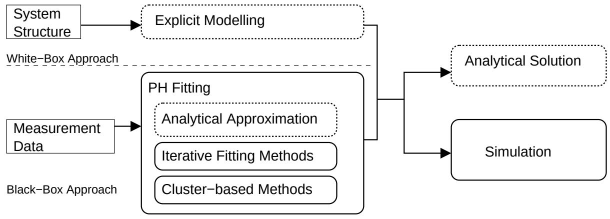

Figure 1: Basic workflow for PH distributions.

class [8] and consider extension to the general PH class [5].

In the final step we introduce the generic libphprng library for PH random-variate generation in simulation frameworks [6]. The library is part of the Butools package [1] and implements the efficient algorithms introduced in the previous part. Libphprng combines efficient methods with easy usage. Furthermore, libphrng integrates seamlessly with discrete-event simulators such as OMNeT++ without any changes to the library core. We illustrate application of the library in popular network simulators using a case-study. The case study [12] covers evaluation of syntonisation accuracy in the precision-time protocol (PTP). We show how the use of PH distributions can tremendously reduce simulation run-times and improve accuracy of the results.

2. INTENDED AUDIENCE

The intended audience of the tutorial are practice-oriented researchers, especially from the networking community, and users of discrete-event simulation. Furthermore, we also invite the attendance of researchers who are adept at using PH distributions in analytical approaches and may be interested in broadening their focus to simulation.

3. ASSUMED BACKGROUND OF ATTENDEES

We assume some working knowledge and practical experience with system evaluation, Markovian modelling and performance analysis.

4. BRIEF BIOGRAPHIES OF SPEAKERS

Philipp Reinecke received his Master's degree in Computer Science at the Humboldt-Universität zu Berlin in 2007. He is working in various areas of system modelling and evaluation, applying methods ranging from experiments in testbeds through simulation to analytical methods. He will submit his PhD thesis on the efficient use of phase-type distributions in the spring of 2012.

Miklós Telek graduated as an electrical engineer at the Faculty of Electrical Engineering, Technical University of Budapest in 1987. Since 1990 he has been with the Department of Telecommunications, Technical University of Budapest, where he is a professor now. His current research interests include various aspects of stochastic performance modeling and analysis of computer and communication systems.

Katinka Wolter Katinka Wolter received her Diploma and PhD degree from the Technical University Berlin in 1995 and 1999, respectively. She has worked on stochastic modelling techniques and retry mechanisms for reliability and gained her habilitation degree from Humboldt-University in 2008. Currently, she is guest-professor for Dependable Systems at Freie Universität Berlin and about to join Newcastle University, UK.

5. REFERENCES

[1] L. Bodrog, P. Buchholz, A. Heindl, A. Horváth, G. Horváth, I. Kolossváry, Z. Németh, P. Reinecke, M. Telek, and M. Vécsei. Butools: Program packages for computations with PH, ME distributions and MAP, RAP processes. http://webspn.hit.bme.hu/˜butools, October 2011.

[2] P. Buchholz, A. Horvath, and M. Telek. Stochastic petri nets with low variation matrix exponentially distributed firing time. *International Journal of Performability Engineering*, 7:441–454, 2011. Special issue on Performance and Dependability Modeling of Dynamic Systems.

[3] P. Buchholz and M. Telek. On minimal representation of rational arrival processes. In *Madrid Conference on Qeueuing theory (MCQT)*, June 2010.

[4] P. Buchholz and M. Telek. Composition and equivalence of markovian and non-markovian models. In *8th International Conference on Quantitative Evaluation of SysTems (QEST)*, pages 213–222, sept 2011.

[5] G. Horváth, P. Reinecke, M. Telek, and K. Wolter. Efficient Generation of PH-distributed Random Variates. Under review.

[6] P. Reinecke and G. Horváth. Phase-type Distributions for Realistic Modelling in Discrete-Event Simulation. In *OMNeT++ Workshop 2012*. ACM, 2012. To appear.

[7] P. Reinecke, T. Krauß, and K. Wolter. Cluster-based Fitting of Phase-type Distributions to Empirical Data. Under review.

[8] P. Reinecke, M. Telek, and K. Wolter. Reducing the Costs of Generating APH-Distributed Random Numbers. In B. Müller-Clostermann, K. Echtle, and E. Rathgeb, editors, *MMB & DFT 2010*, number 5987 in LNCS, pages 274–286. Springer-Verlag Berlin Heidelberg, 2010.

[9] P. Reinecke, K. Wolter, L. Bodrog, and M. Telek. On the Cost of Generating PH-distributed Random Numbers. In G. Horváth, K. Joshi, and A. Heindl, editors, *Proceedings of the Ninth International Workshop on Performability Modeling of Computer and Communication Systems (PMCCS-9)*, pages 16–20, Eger, Hungary, September 17–18, 2009 2009.

[10] A. Varga. The OMNeT++ Discrete Event Simulation System. In *Proceedings of the European Simulation Multiconference (ESM'2001)*, June 2001.

[11] Various authors. The Network Simulator ns-2. http://www.isi.edu/nsnam/ns/. (last seen May 11, 2010).

[12] K. Wolter, P. Reinecke, and A. Mittermaier. Model-based Evaluation and Improvement of PTP Syntonisation Accuracy in Packet-Switched Backhaul Networks for Mobile Applications. In N. Thomas, editor, *Computer Performance Engineering. Proceedings of the 8th European Performance Engineering Workshop, EPEW 2011*, number 6977 in LNCS, pages 219–234. Springer, October 2011.