TIPPtool: Compositional Specification and Analysis of Markovian Performance Models

H. Hermanns*, V. Mertsiotakis**, and M. Siegle

Informatik 7, Universität Erlangen-Nürnberg, Erlangen, Germany

Abstract. In this short paper we briefly describe a tool which is based on a Markovian stochastic process algebra. The tool offers both model specification and quantitative model analysis in a compositional fashion, wrapped in a user-friendly graphical front-end.

1 Compositional Performance Modelling

Classical process algebras have been designed as compositional description formalisms for concurrent systems. In *stochastic* process algebras temporal information is attached to actions in the form of continuous random variables representing activity durations, making it possible to specify and analyse both qualitative and quantitative properties. This short paper is about the TIPPtool [5], a tool that emerged from the TIPP project which focussed on a basic framework supporting both functional specification and performance evaluation in a single, process algebraic formalism [6]. The formalism is basically a superset of LOTOS [1], including means to specify exponentially distributed delays. It hence provides a bridge between qualitative and quantitative evaluation, the latter based on Markov chain analysis. More precisely, the underlying semantics of the specification language gives rise to homogeneous continuous time (semi-)Markov chains that can be analysed numerically by means of efficient techniques. Besides some support for analysis of functional aspects, the tool offers algorithms for numerical performance analysis of a given process algebraic specification. Exact and approximate evaluation techniques are provided to calculate various measures of interest. The tool also offers semi-automatic compositional minimisation of complex models based on equivalence-preserving transformations.

2 Model specification and analysis

The specification language of the TIPPtool is a superset of LOTOS[1]. In particular, a distinguished type of prefix, $(a, r);\ P$, is supported, denoting that action a occurs after a delay Δ which is exponentially distributed with rate parameter r (i.e. $Prob(\Delta \leq t) = 1 - e^{-rt}$); afterwards the process behaves as P.

Actions arising from ordinary prefix $a; P$ are called *immediate* actions. They happen as soon as possible if not prevented by the environment, following the *maximal progress* assumption. In particular, internal (or hidden) immediate actions are assumed

* Current affiliation: *Systems Validation Centre, University of Twente, Enschede, The Netherlands*

** Current affiliation: *Development Access, Lucent Technologies, Nürnberg, Germany*

[1] Data types are treated more liberally than in standard LOTOS; integers are a built-in data type.

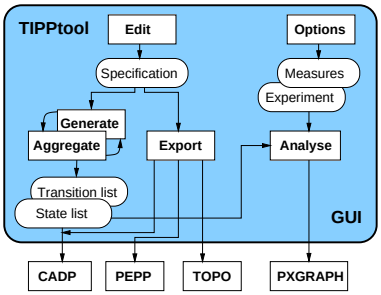
to happen immediately when enabled. In addition to the basic language elements, *process instantiation*, *parametric processes* and *inter-process communication* can be used to model complex dependences, such as value passing or mobility.

Conservatively extending classical process algebras, a labelled transition system (LTS) is generated from the system specification using structural operational rules [6]. Corresponding to timed and immediate actions there are two types of transitions between states: timed transitions and immediate transitions. The LTS can hence be regarded as a semi-Markov process. Under certain conditions (checked by the tool) the semi-Markov process can be transformed into a continuous time Markov chain. Verifying these properties involves equivalence preserving transformations, based on a stochastic variant of Milner's observational congruence [6]. Since this relation is compositional, it can be applied to minimise the state space of a specification in a componentwise fashion. This minimisation abstracts from internal immediate steps and it aggregates the Markov chain based on the concept of lumpability [10], while preserving functional and stochastic information. For a particular Markov chain, a system of ordinary differential equations needs to be solved in order to obtain the state probabilities at a particular time instant t (transient analysis). Alternatively, solving a linear system of equations leads to the state probabilities in the equilibrium (stationary analysis). These limiting probabilities (where $t \to \infty$) are known to exist for arbitrary finite (homogeneous, continuous time) Markov chains.

3 Tool features and structure

In its current version 2.3, the TIPPtool provides the following functionality:

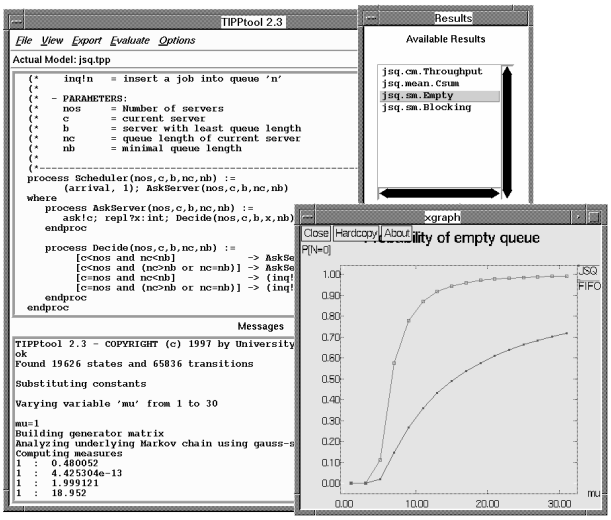
- Model description by means of a LOTOS-based notation,
- Reachability analysis based on the operational semantics,
- Algorithms for deadlock detection and tracing to a given state,
- Algorithms for checking bisimulation-style equivalences and for the minimisation of (sub-)models,
- Stationary and transient analysis of the underlying Markov chain,
- Functions for the calculation of performance and dependability measures,
- Support of experiment series,
- Output of numerical results using the tool PXGRAPH,
- Interfacing with other tools.



The tool consists of several components whose interaction is shown in the figure on the right. Specifications can be created with an editor (Edit component). The Generate/Aggregate component is responsible for parsing the specification, for the generation of the LTS and for its minimisation according to an equivalence notion. The user may currently choose between four (stochastic variants of) classical congruences. This minimisation is known to be particularly beneficial if it is applied to components of a larger specification in a stepwise, compositional fashion. In the TIPPtool, semi-automatic compositional minimisation is supported in an elegant way: By highlighting a certain fragment of the specification with the mouse, it is

possible to invoke compositional minimisation of that fragment. When the minimised representation is computed, a new specification is generated automatically, where the selected fragment has been replaced by its minimised representation.

Via the Options, the user can specify various measures to be calculated, such as the probability of the system being in a certain subset of states, or the throughput (i.e. the mean frequency of occurrence) of some action. An experiment description contains information about model parameters to be varied during analysis. A series of experiments can be carried out automatically in an efficient manner, generating numerical results for different values of a certain model parameter, while the state space only needs to be generated once. Models can be analysed with the Analyse module. This module offers various numerical solution algorithms for the underlying stochastic process, among them two approximate methods [9, 12]. After an experiment series has been carried out, the results are presented graphically with the tool PXGRAPH from UC Berkeley, cf. the screenshot on the right. The Export module of the tool provides interfaces to three other tools, PEPP [4], TOPO [11], and CADP [2]. The former interface generates stochastic task graphs [8], for which the tool PEPP offers a wide range of both exact and approximate analysis algorithms, some of which work even for general distributions. The second interface provides support for the translation of specifications into a format suitable for the LOTOS tool TOPO. Among other functionalities, TOPO is capable of building C-programs from LOTOS specifications. The third interface can be used to exploit the bisimulation equivalence algorithms of the tool ALDEBARAN, as well as other tools (FC2, AUTOGRAPH), for visualisation or functional verification purposes. Here, the interface is on the level of the state space.



We used the programming language STANDARD ML for implementing the parser, the semantics, the bisimulation algorithms and for the approximate Markov chain solution methods. The numerical analysis part is written in C, on top of a library which provides data structures for sparse matrices (SparseLib1.3 from Kenneth Kundert, UC Berkeley). This library has been extended by iterative solution methods for stationary and transient analysis. The clear interface of the library makes it easy to integrate other solution methods into the tool. The communication with the state space generator is done via ASCII-files. For computing the measures, shell-scripts are used, which are based on standard UNIX-tools such as GREP, AWK and SED. Finally, the graphical user interface has been implemented using the scripting language TCL/TK. The communication between the GUI and the other tools is done via UNIX-pipes.

4 Conclusion

In this short paper, we have presented the status quo of the TIPPtool. We have described the particular features of a stochastic process algebra based specification formalism, together with the distinguishing components of the tool. To the best of our knowledge, the TIPPtool is the only existing tool offering compositional minimisation of Markov chain models. TIPPtool is available free of charge for non-commercial institutions, more details can be found at `http://www7.informatik.uni-erlangen.de/tipp/`. Among others, the tool has been applied to the study of performance and dependability aspects of the plain old telephony system [7], a robot control system [3], and a hospital information system [13]. So far, models with up to 10^7 states have been tackled compositionally.

References

1. T. Bolognesi and E. Brinksma. Introduction to the ISO Specification Language LOTOS. In *The Formal Description Technique LOTOS*, pages 23–73, Amsterdam, 1989. North-Holland.
2. Jean-Claude Fernandez, Hubert Garavel, Alain Kerbrat, Radu Mateescu, Laurent Mounier, and Mihaela Sighireanu. Cadp (cæsar/aldebaran development package): A protocol validation and verification toolbox. In *Proceedings of the 8th Conference on Computer-Aided Verification (CAV 96)*, Springer LNCS 1102:437–440. August 1996.
3. S. Gilmore, J. Hillston, R. Holton, and M. Rettelbach. Specifications in Stochastic Process Algebra for a Robot Control Problem. *International Journal of Production Research* 34(4):1065–1080, 1996.
4. F. Hartleb and A. Quick. Performance Evaluation of Parallel Programms — Modeling and Monitoring with the Tool PEPP. In *Proceedings of 7th GI-ITG Fachtagung "Messung, Modellierung und Bewertung von Rechen- und Kommunikationssystemen"*, pages 51–63. Informatik Aktuell, Springer. September 1993.
5. H. Hermanns, U. Herzog, U. Klehmet, V. Mertsiotakis, and M. Siegle. Compositional Performance Modelling with the TIPPtool. In *10th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation (TOOLS 98)*, Springer LNCS 1469:51–62. September 1998.
6. H. Hermanns, U. Herzog, and V. Mertsiotakis. Stochastic Process Algebras – Between LOTOS and Markov Chains. *Computer Networks and ISDN Systems*, 30(9-10):901–924, 1998.
7. H. Hermanns and J.P. Katoen. Automated Compositional Markov Chain Generation for a Plain Old Telephony System. *Science of Computer Programming*. to appear.
8. U. Herzog. A Concept for Graph-Based Stochastic Process Algebras, Generally Distributed Activity Times and Hierarchical Modelling. In *Proc. of the 4th Workshop on Process Algebras and Performance Modelling*, pages 1–20. Universitá di Torino, CLUT, 1996.
9. J. Hillston and V. Mertsiotakis. A Simple Time Scale Decomposition Technique for SPAs. In *The Computer Journal*, 38(7):566–577, 1995.
10. J.G. Kemeny and J.L. Snell. *Finite Markov Chains*. Springer, 1976.
11. J.A. Manas, T. de Miguel, and J. Salvachua. Tool Support to Implement LOTOS Specifications. *Computer Networks and ISDN Systems*, 25(7):815–839, 1993.
12. V. Mertsiotakis and M. Silva. Throughput Approximation of Decision Free Processes Using Decomposition. In *Proc. of the 7th Int. Workshop on Petri Nets and Performance Models*, pages 174–182, St. Malo, June 1997. IEEE CS-Press.
13. M. Siegle, B. Wentz, A. Klingler, and M. Simon. Neue Ansätze zur Planung von Klinikkommunikationssystemen mittels stochastischer Leistungsmodellierung. In *42. Jahrestagung der Deutschen Gesellschaft für Medizinische Informatik, Biometrie und Epidemiologie (GMDS)*, pages 188 – 192, Ulm, September 1997. MMV Medien & Medizin Verlag.