Let's Evaluate Performance, Algebraically

Marco Bernardo
Università di Bologna, Dipartimento di Scienze dell'Informazione
Mura Anteo Zamboni 7, 40127 Bologna, Italy
E-mail: bernardo@cs.unibo.it

We discuss the benefits of using process algebras in the field of performance modeling and evaluation. We briefly present problems that have been tackled, theoretical results, and future developments.

Many computing systems consist of a possibly huge number of components that not only work independently but also communicate with each other from time to time. Examples of such systems are communication protocols, operating systems, embedded control systems for automobiles, airplanes, and medical equipment, railway signaling systems, air traffic control systems, distributed systems and algorithms, computer architectures, and integrated circuits.

The catastrophic consequences of failures, such as loss of human lives, environmental damages, and financial losses, in many of these critical systems compel computer scientists to develop formal description techniques for ensuring that these systems are implemented correctly despite of their complexity.

Formal methods are conceived to allow the correctness of a system design to be formally verified because the design can be described in a mathematically precise fashion, correctness criteria can be specified in a similarly precise way, and the design can be rigorously proved to meet or not the stated criteria.

Although a number of description techniques and related software tools have been developed to support the formal modeling and verification of functional properties of systems, only in recent years temporal characteristics have received attention. This has required extending formal description techniques by introducing the concept of time, represented either in a deterministic way or in a stochastic way. In the deterministic case the focus typically is on verifying the satisfaction of real time constraints, i.e. the fact that the execution of specific actions is guaranteed by a fixed deadline after some event has happened. As an example, if a train is approaching a railroad crossing, then bars must be guaranteed to be lowered on due time.

In the stochastic case, instead, systems are considered whose behavior cannot be deterministically predicted as it fluctuates according to some probability distribution. Due to economic reasons, such stochastically behaving systems are referred to as shared resource systems because there is a varying number of demands competing for the same resources. The consequences are mutual interference, delays due to contention, and varying service quality. Additionally, resource failures significantly influence the system behavior. The purpose of performance evaluation is to investigate and optimize the time varying behavior within and among individual components of such shared resource systems. As an example, if we consider again a railway system, we may be interested in minimizing the average train delay or studying the characteristics of the flow of passengers.

The desirability of evaluating the performance of a system in the early stages of its design has been widely recognized [8]. Nevertheless, it often happens that a system is tested for efficiency only after it has been fully designed and tested for functionality. This results in two problems. On the one hand, the detection of poor performance causes the system to be designed again, so the cost of the project increases. On the other hand, the performance related analysis is done on a model of the system extracted from the implementation while the functionality related analysis is usually performed on a model derived from a system design. As an example, functional verification could be conducted on a Petri net [21] or an algebraic term [19] describing the system, while performance could be evaluated on a Markov chain or a queueing network model [17] of the system. As a consequence, great care must be taken to ensure that these models are consistent with one another, i.e. do reflect (different aspects of) the same system.

Despite of the solid theoretical foundation and a rich practical experience, performance evaluation is still an art mastered by a small group of specialists. This is particularly true when systems are large or there are sophisticated interdependencies. The problem is that a linguistic support for describing performance aspects is not provided. Performance is usually modeled by resorting to stochastic processes such as Markov chains or graphical formalisms such as stochastically timed Petri nets [1], which yield low level models hard to understand as far as complex systems are concerned, or notations such as queueing networks, which are more abstract but do not allow all the important aspects of a system to be modeled in a formal way. From the designer viewpoint, it would be extremely advantageous to profit from a general purpose description technique resulting in readable performance models, possibly specified in a compositional way, which are unambiguous and can be formally analyzed in an efficient way.

To solve the problems mentioned above, in the 90's several proposals have been made to extend process algebras with performance modeling features, resulting in stochastically timed process algebras. Like classical process algebras, they are algebraic languages endowed with a small set of powerful operators whereby it is possible to systematically construct process terms from simpler ones. Moreover, stochastically timed process algebras comprise a family of actions which permit to express both the type and the duration of the activities executed by the systems being modeled. As a consequence, integrated semantic models (such as action labeled transition systems) underlying process terms contain both functional and performance information, so from them projected models (such as action type labeled transition systems and Markov chains) can be derived which are guaranteed to be consistent with each other and are used to investigate functional and performance characteristics, respectively, such as deadlock freeness and resource utilization. Also integrated semantic models are useful from the analysis standpoint as they allow to investigate mixed characteristics, such as mean time to deadlock, which refer to both functionality and performance.

Stochastically timed process algebras thus provide a linguistic, compositional, general purpose means to formally describe and analyze functional, performance, and mixed properties of systems. They also turn out to be advantageous compared to widely used stochastically timed extensions of Petri nets [1] because of the linguistic support they provide and the fact that they enable the designer to conduct an integrated system analysis by means of suitable notions of equivalence, which relate terms describing systems with the same functional and performance properties.

It is quite surprising that Nounou and Yemini [20] already presented in 1985 most ideas of stochastically timed process algebras, but they did not continue research in this direction. The seminal papers in the area of stochastically timed process algebras were written by Herzog and appeared in 1990 [13; 14]. Starting from that work, several research groups elaborated their algebra and the related tool support. Among the various proposals, we cite TIPP [10] (University of Erlangen), PEPA [15] (University of Edinburgh), and EMPA [4] (University of Bologna).

The stochastically timed process algebras mentioned above concentrate on the case in which action durations are expressed by means of exponentially distributed random variables. From a foundational point of view, the importance of the memoryless property of the exponential distribution has been recognized. By virtue of such a property, the semantics for the algebras above can simply be defined in the interleaving style (as an exponentially timed activity can be thought of as being started in the state where it is terminated) and the performance models underlying process terms turn out to be Markov chains, so the related theory can be exploited to effectively derive efficiency measures. Moreover, it has been discovered that a smooth extension of the probabilistic bisimulation equivalence proposed by Larsen and Skou [18] is the right notion of equivalence in this setting. The Markovian bisimulation equivalence, which is defined on the integrated semantic model, is shown to be a congruence with respect to the operators of the algebras, has a clear relationship with the Markov chain aggregation known as ordinary lumping, and turns out to be the coarsest congruence contained in the intersection of a purely functional bisimulation equivalence and a purely performance bisimulation equivalence, thereby stressing the need of an integrated semantic model in order for compositional reasoning to be permitted.

From the applicative viewpoint, software tools based on the Markovian process algebras above have been developed which provide an automated support to both functional verification and performance evaluation. These tools differ not only for the expressive power of the algebras they rely on, but also for their architecture. The TIPP-tool [12] and the PEPA Workbench [9] contain their own analysis routines, while TwoTowers [3] builds on already existing tools which carry out functional or performance analysis. Such tools are being used to conduct more and more complex case studies about the performance of communication protocols and distributed algorithms, in order to emphasize the adequacy of the Markovian process algebra approach.

Implementing tools has called for further theoretical investigations. First, a method to formally specify performance measures was needed. To this purpose, the technique of rewards has been proposed both in [6], where rewards are expressed through modal logic, and in [2], where an algebraic theory of rewards is worked out. Second, efficient solution techniques for the Markov chains underlying process terms had to be devised, possibly exploiting the syntactical structure of terms. An overview of such techniques can be found in [16].

Stochastically timed process algebras have been so far deeply investigated in the Markovian case. The future theoretical developments are thus expected in the treatment of activities with generally distributed durations, in order to be able to formally model and analyze a wider range of systems. When arbitrarily distributed durations come into play, the memoryless property no longer holds. From a foundational point of view, this means that activities can no more be thought of as being started in the states where they are terminated and that performance models are no longer Markov chains. In other words, it is necessary to understand how to define the semantics for stochastically timed process algebras with general distributions and what performance models underlie their process terms. To cope with this, several proposals have recently appeared which are quite different from one another. Among such proposals, we cite those stochastically timed process algebras with general distributions for which a notion of equivalence (which is peculiar of the process algebra approach) has been developed. CCS+ [11] solves the problem of identifying the start and the termination of an activity at the syntactic level by means of suitable operators which represent the random setting of a timer and the expiration of a timer, respectively. Semantic models are infinite transition systems from which performance measures can be derived via simulation. SPADES [7] solves the problem above in a similar way but the underlying semantic models are given by stochastic automata equipped with clocks. Performance measures can be obtained by simulating such automata. GSMPA [5] solves instead the problem of identifying the start and the termination of an activity at the semantic level through the ST approach. Additionally, the underlying performance models are explicitly defined to be generalized semi Markov processes, which can be analyzed not only via simulation but also numerically sometimes.

We conclude by noting that introducing general distributions should hopefully lead to bring together in one uniform framework the characteristics of deterministically timed process algebras, exploited to model and analyze real time systems, and stochastically timed process algebras, adopted for performance evaluation purposes.

REFERENCES

[1] M. Ajmone Marsan, *"Stochastic Petri Nets: An Elementary Introduction"*, in *Advances in Petri Nets '89*, LNCS 424:1-29, 1990

[2] M. Bernardo, *"An Algebra-Based Method to Associate Rewards with EMPA Terms"*, in Proc. of the *24th Int. Coll. on Automata, Languages and Programming (ICALP '97)*, LNCS 1256:358-368, Bologna (Italy), 1997

[3] M. Bernardo, W.R. Cleaveland, S.T. Sims, W.J. Stewart, *"TwoTowers: A Tool Integrating Functional and Performance Analysis of Concurrent Systems"*, to appear in Proc. of the *IFIP Joint Int. Conf. on Formal Description Techniques and Protocol Specification, Testing, and Verification (FORTE/PSTV '98)*, Paris (France), 1998

[4] M. Bernardo, L. Donatiello, R. Gorrieri, *"A Formal Approach to the Integration of Performance Aspects in the Modeling and Analysis of Concurrent Systems"*, in Information and Computation 144:83-154, 1998

[5] M. Bravetti, M. Bernardo, R. Gorrieri, *"Towards Performance Evaluation with General Distributions in Process Algebras"*, in Proc. of the *9th Int. Conf. on Concurrency Theory (CONCUR '98)*, LNCS 1466:405-422, Nice (France), 1998

[6] G. Clark, *"Formalising the Specification of Rewards with PEPA"*, in Proc. of the *4th Int. Workshop on Process Algebras and Performance Modelling (PAPM '96)*, CLUT, pp. 139-160, 1996

[7] P.R. D'Argenio, J.-P. Katoen, E. Brinksma, *"An Algebraic Approach to the Specification of Stochastic Systems"*, in Proc. of the *IFIP Working Conf. on Programming Concepts, Methods and Calculi (PROCOMET '98)*, Chapman & Hall, pp. 126-147, Shelter Islands (NY), 1998

[8] D. Ferrari, *"Considerations on the Insularity of Performance Evaluation"*, in IEEE Trans. on Software Engineering 12:678-683, 1986

[9] S. Gilmore, J. Hillston, *"The PEPA Workbench: A Tool to Support a Process Algebra-based Approach to Performance Modelling"*, in Proc. of the *7th Int. Conf. on Modelling Techniques and Tools for Computer Performance Evaluation*, LNCS 794:353-368, Wien (Austria), 1994

[10] N. Götz, U. Herzog, M. Rettelbach, *"Multiprocessor and Distributed System Design: The Integration of Functional Specification and Performance Analysis Using Stochastic Process Algebras"*, in Proc. of the *16th Int. Symp. on Computer Performance Modelling, Measurement and Evaluation (PERFORMANCE '93)*, LNCS 729:121-146, Rome (Italy), 1993

[11] P.G. Harrison, B. Strulo, *"Stochastic Process Algebra for Discrete Event Simulation"*, in *Quantitative Methods in Parallel Systems*, Springer, pp. 18-37, 1995

[12] H. Hermanns, V. Mertsiotakis, M. Rettelbach, *"A Construction and Analysis Tool Based on the Stochastic Process Algebra TIPP"*, in Proc. of the *2nd Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS '96)*, LNCS 1055:427-430, Passau (Germany), 1996

[13] U. Herzog, *"Formal Description, Time and Performance Analysis – A Framework"*, in *Entwurf und Betrieb verteilter Systeme*, Informatik Fachberichte 264, Springer, 1990

[14] U. Herzog, *"EXL: Syntax, Semantics and Examples"*, Tech. Rep. IMMD7-16/90, University of Erlangen (Germany), 1990

[15] J. Hillston, *"A Compositional Approach to Performance Modelling"*, Cambridge University Press, 1996

[16] J. Hillston, *"Exploiting Structure in Solution: Decomposing Composed Models"*, in Proc. of the *6th Int. Workshop on Process Algebras and Performance Modelling (PAPM '98)*, pp. 1-15, Nice (France), 1998

[17] L. Kleinrock, *"Queueing Systems"*, John Wiley & Sons, 1975

[18] K.G. Larsen, A. Skou, *"Bisimulation through Probabilistic Testing"*, in Information and Computation 94:1-28, 1991

[19] R. Milner, *"Communication and Concurrency"*, Prentice Hall, 1989

[20] N. Nounou, Y. Yemini, *"Algebraic Specification-Based Performance Analysis of Communication Protocols"*, in Proc. of the *5th Int. Conf. on Protocol Specification, Testing and Verification (PSTV V)*, North Holland, pp. 541-560, 1985

[21] C.A. Petri, *"Communication with Automata"*, Tech. Rep. RADC-TR-65-377, Griffiss Air Force Base (NY), 1966