# The ipc/HYDRA Tool Chain for the Analysis of PEPA Models

Jeremy T. Bradley William J. Knottenbelt
Department of Computing, Imperial College London
180 Queen's Gate, London SW7 2BZ, United Kingdom.
{jb,wjk}@doc.ic.ac.uk

## Abstract

*PEPA is a popular stochastic process algebra and a powerful formalism for describing performance models of communication and computer systems. We augment the current state-of-the-art in the analysis of PEPA models by presenting a tool set that can not only perform steady-state and transient analysis, but also response time analysis. Response time densities and quantiles are important performance metrics which are used to specify service level agreements (SLAs) and benchmarks. HYDRA is a tool specialising in response time analysis of large Markov systems based on stochastic Petri nets. By using the Imperial PEPA compiler (*ipc*), we can generate a HYDRA model from a PEPA model and obtain steady-state, transient and response time measures based on the original PEPA description.*

## 1. Introduction

Traditionally, performance analysis of concurrent systems has focused on steady-state analysis—that is, on calculating the long-run probability that a system will be in each of its reachable states. This is adequate to predict standard resource-based measures, such as throughput and utilisation, but is inadequate to answer questions about transient state distributions, such as "What is the probability that a router has 10 packets in its buffer 20 seconds after startup?", or questions about response time quantiles, such as "What is the probability that a packet will pass through a router in under 5 milliseconds?" The latter type of question is particularly important since response time quantiles are increasingly specified as key quality-of-service metrics in benchmarks and Service Level Agreements.

PEPA [5] is a popular Markovian process algebra for specifying compositional performance models. PEPA models reduce to an underlying Markov chain, which can be analysed using appropriate Markovian techniques. While PEPA Workbench can produce both steady-state and transient results, we present the ipc/HYDRA tool chain which can perform steady-state, transient and response time results from a PEPA model.

The HYpergraph-based Distributed Response time Analyser or HYDRA [3] is a parallel tool that uses an efficient uniformization-based technique to compute transient distributions and response time densities and quantiles in Markov chains with large state spaces of the order of $10^7$ states. It is an extension of DNAmaca, a tool for the steady-state analysis of large Markov chains. By using state-of-the-art hypergraph partitioning techniques HYDRA yields excellent scalability on distributed memory parallel computers and is able to utilise effectively the compute power and RAM provided by a network of workstations.

In this paper, we describe how response time quantities can be calculated from PEPA models using the *Imperial PEPA compiler* (ipc) and the HYDRA tool. ipc translates a PEPA specification (described in Section 2) at the component level (as opposed to the global state-space level) to a stochastically identical HYDRA specification (Section 3). We then use HYDRA to extract the steady-state, transient or response time densities and quantiles of interest (Section 4).

## 2. PEPA

PEPA is a parsimonious stochastic process algebra that can describe compositional stochastic models. These models consist of components whose actions incorporate random exponential delays. Full details of the PEPA process algebra can be found in [5]. In brief, the syntax of a PEPA component, $P$, is represented by:

$$P \quad ::= \quad (a,\lambda).P \;\big|\; P+P \;\big|\; P \underset{S}{\bowtie} P \;\big|\; P/L \;\big|\; A$$

$(a,\lambda).P$ is an action prefix operation. $P_1+P_2$ is a choice operation between two components. $P_1 \underset{S}{\bowtie} P_2$ is the cooperation operator between two components which synchronise over a set of actions, $S$. $P/L$ is a hiding operator of a set of actions, $L$. $A$ is a constant label.

## 3. ipc Tool Architecture

ipc enables additional solution procedures and response time analysis capabilities over existing PEPA tool sets such as PEPA Workbench, PRISM [6] and Möbius [2]. Additionally, ipc is also the only tool to support a complete implementation of Hillston's apparent rate synchronisation [5].

ipc performs the translation from PEPA to the stochastic Petri net formalism that HYDRA employs. ipc converts each PEPA component submodel into an SPN. The models are combined using shared SPN transitions to represent the shared actions of the PEPA synchronisation. ipc calculates the apparent rate of each component as it is combined in order to represent the complete PEPA synchronisation rate.

ipc uses the stochastic probe measurement mechanism from [1]. A stochastic probe is a fragment of process algebra (say a component $Q$) that facilitates the expression of stochastic measures in a PEPA model, say $M$. By synchronising the probe with the model, $M \underset{L}{\bowtie} Q$, specified model behaviour (from set $L$) can be observed before turning a measurement on (having the probe change to a *running state*) and further user-defined behaviour (also in $L$) is looked for before switching the measurement off (or the probe switches to an *off state*). As HYDRA is a state-oriented formalism, ipc encodes the measurement (response time, transient, or steady-state) in terms of the observed state of the probe.

Figure 1 shows a cumulative distribution function from an 83,000 state PEPA active badge model from [4]; passage time quantiles can easily be derived from this.

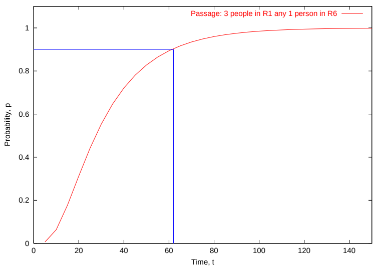


**Fig. 1. Cumulative response time distribution and quantile for any 3 people ending up in room 6 in the corridor model**

## 4. The HYDRA/DNAmaca Analyser

DNAmaca is a modelling language for Markov and semi-Markov chains. The HYDRA version is an efficient hypergraph-based implementation of DNAmaca which operates on an underlying Generalised Stochastic Petri Net. HYDRA uses *uniformization* to calculate transient measures in Markov models. For the calculation of response time densities, HYDRA adds an extra absorbing state to the chain, which is made the sole successor of all the response time target states. Uniformization is applied to the modified chain and the modification ensures we obtain the *first* response time density. Both density function and cumulative distribution can be produced with this technique.

## 5. Conclusion

In this paper, we have briefly introduced the tool chain ipc/HYDRA for performing efficient steady-state, transient and response time analysis on PEPA models. By translating PEPA models into the stochastic Petri net formalism of HYDRA, we can make use of HYDRA's many steady-state solution techniques while also using uniformization to gather transient and response time metrics.

## References

[1] A. Argent-Katwala, J. T. Bradley, and N. J. Dingle. Expressing performance requirements using regular expressions to specify stochastic probes over process algebra models. In *Proc. of the 4th Int. Workshop on Software and Performance*, pages 49–58. ACM, January 2004.

[2] G. Clark and W. Sanders. Implementing a stochastic process algebra within the Möbius modeling framework. In *Proc. of the 1st PAPM-PROBMIV Workshop*, volume 2165 of *LNCS*, pages 200–215. Springer-Verlag, 2001.

[3] N. J. Dingle, W. J. Knottenbelt, and P. G. Harrison. HYDRA: HYpergraph-based Distributed Response-time Analyser. In *PDPTA'03*, pages 215–219, June 2003.

[4] S. Gilmore, J. Hillston, and G. Clark. Specifying performance measures for PEPA. In *Proc. of the 5th ARTS Workshop*, volume 1601 of *LNCS*, pages 211–227. Springer-Verlag, 1999.

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

[6] M. Kwiatkowska, G. Norman, and D. Parker. Probabilistic symbolic model checking with PRISM: A hybrid approach. In *TACAS'02 Workshop*, volume 2280 of *LNCS*, pages 52–66. Springer-Verlag, 2002.