Performability-Aware Computing (PaCo): Logics, Models, and Languages

Marco Bernardo

University of Urbino – Italy

General Information

- Call for project proposals: PRIN 2007 (issued by MIUR).
- Deadline: Oct 2007 (everything started at ICTCS 2007 in Rome).
- Research areas: Mathematics & Computer Science (100%).
- Approved: July 2008.
- Starting date: 22 Sep 2008.
- Closing date: 22 Sep 2010 (extended by 6 months).
- Cost: 102,526 euros (70% funded by MIUR).

Research Units

1. University of Urbino: *Bernardo*, Aldini, Padovani, ...
2. University of Firenze: *De Nicola*, Loreti, Latella, Massink, ...
3. University of Camerino: *Corradini*, Tesei, Di Berardini, ...
4. University of Torino: *Sproston*, Bernardi, Troina, Donatelli, ...
5. University of L'Aquila: *Cortellessa*, Berardinelli, Trubiani, ...

- Concurrency theory, performance evaluation, software engineering.

Project Meetings

1. Kick-off meeting: Bertinoro, Oct 2008 (organized by Urbino RU).
2. First mid-term meeting: Lucca, Jun 2009 (organized by Firenze RU).
3. Second mid-term meeting: L'Aquila, March 2010 (organized by L'Aquila RU).
4. Final meeting: Camerino, Sep 2010 (organized by Camerino RU, with ICTCS 2010).

Project Setting

- The design of software systems is increasingly influenced by the identification and the satisfaction of nonfunctional requirements, such as performance and dependability.
- Performance refers to the quality of service guaranteed by a system when the system works properly: throughput, utilization, queue length, response time, ...
- Dependability refers instead to the reasonable expectations about the service provided by a system, where the service is classified as being proper or improper depending on whether it is provided according to its specification or not: reliability, availability, safety, security, ...
- Performability expresses the level at which a system is able to perform.
- The performance of a system may degrade in the presence of faults, with the service remaining proper.

Project Objectives

- By following the modern model-driven software development view, the design of performability-aware systems requires:
  - Formalisms for modeling the systems themselves.
  - Formalisms for specifying the performability properties of interest.
  - Techniques for verifying those performability properties.
- Many formalisms exist ranging from UML and architectural description languages to modal/temporal logics, process algebras, Petri nets, and automata.
- Advancing the state of the art by establishing new theoretical result for the existing formalisms or making those formalisms more expressive.
- Integrating logics, models, and languages for the description and the analysis of performability-aware systems.

WP1: Temporal Logics and Model Checking

- RUs: Firenze, Torino, Urbino.
- Stochastic temporal logics integrating spatial modalities.
- Local and distributed model-checking algorithms for stochastic temporal logics.
- Efficient model-checking algorithms for subclasses of probabilistic timed automata.

WP2: Process Algebras and Behav. Equiv.

- RUs: Firenze, Urbino.
- Weak behavioral equivalences for Markovian process algebras.
- Approximated behavioral equivalences for Markovian process algebras.
- Characterizations of stochastic behavioral equivalences.
- New semantics and equivalences for stochastic process algebras with mobility.

WP3: Automata and Probabilistic Timed Models

- RUs: Camerino, L'Aquila, Torino.
- Generalization of time divergence in probabilistic timed automata.
- Abstraction and refinement mechanisms for probabilistic timed automata.
- Characterization of fairness, liveness, urgency, distribution, and location and context awareness in probabilistic timed models.

WP4: Specification of Performability Measures

- RUs: Camerino, Firenze, L'Aquila, Torino, Urbino.
- New UML profiles for performability domains.
- Offline and online composition mechanisms for performability attributes.
- Mechanisms for expressing asymptotical performability measures for nondeterministic timed systems.
- Rephrasing efficiency measures for timed systems in a probabilistic setting.
- User-friendly mechanisms for expressing path properties in UML.
- New language integrating MSL and CSL/PTCTL/MoSL.

WP5: Model Transformations and Type Systems

- RUs: Camerino, L'Aquila, Torino, Urbino.
- New transformation functions from UML models to performability models.
- New backward propagation functions of results from performability models to UML models.
- New transformation functions between performability models.
- New backward propagation functions of results between performability models.
- Type systems supporting the correct transformation of performability models.

What to Know More About PaCo?

- Please visit http://www.sti.uniurb.it/paco/:
  - Slides presented at the project meetings.
  - Papers accepted at conferences and workshops.
- Researchers involved in PaCo are open to work with people interested in quantitative modeling and evaluation of complex systems.