Projet IOP
Intégration d’outils à CosyVerif

Étienne André, Fabrice Kordon, Alban Linard, Laure Petrucci


11 juin 2014
Motivation

Dissemination of Verification Tools

- Application of formal methods to dedicated cases studies
- Towards technological transfer to industry
- Tools organised around formalisms

Academics ≠ Developers 😕

- Need to share effort (platform, interfaces, distribution mechanisms)
- Need to share definitions (typically formalisms)
- Coordinated effort to better handle a complex context of interrelated formal notations
  - Variants of Petri nets
  - Variants of automata
  - etc.
Principles of the *CosyVerif* platform

- Distributed and open [AHHKLLP13]
  - Developed at ENS Cachan, Paris 13, UPMC, etc.
- Supports different families of formalisms
  - automata
  - Petri nets
- 12 concrete formalisms
- 2-layered XML-based description language: [ABDHKLP13]
  - FML, Formalism Markup Language (modelling language description)
  - GrML, Graph Markup Language (actual model description)
- Reuse of existing formalisms
- Open to new tool contributions
- Tools invoked through web services transparent to the user
- Graphical user interface: Coloane
- Repository of models
The CosyVerif Verification Platform

The CosyVerif Architecture

Cluster 1
- server
- super server

Cluster 2
- super server

Cluster N
- super server

Authentication Server

Formalisms & Models Repository

BenchKit, a benchmark tool

SOAP Web Service Bus

Web Client

Tool and command-line library

Graphical User interface
The Coloane User Interface

The *Coloane* User interface (Graphical client)

Command-line client is also available for script-based access to tools
## Formalisms and Tools

<table>
<thead>
<tr>
<th>Formalisms</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Petri Nets</strong></td>
<td><strong>PROD (Univ. Helsinki, Symmetric nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>PNXDD (LIP6, Symmetric nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Crocodile (LIP6, Symmetric nets w. bags)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Cunf (LSV, P/T nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Cosmos (LSV, Stochastic Petri nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>GreatSPN invariants (Univ. Torino, P/T nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Helena (LIPN, HL nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>ModGraph (LIPN, HL nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>ObsGraph (LIPN, HL nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Structural bounds (LIP6, P/T nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Unfold into P/T nets (LIP6, Symmetric nets)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Various exports (LIP6, P/T nets)</strong></td>
</tr>
<tr>
<td><strong>Automata</strong></td>
<td><strong>Imitator (LIPN, Timed automata)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Modgraph (LIPN, Synchronised automata)</strong></td>
</tr>
</tbody>
</table>

**References:**
- [Kordon et al., 2012]
- [Colange et al., 2011]
- [Baldan et al., 2012]
- [Ballarini et al., 2011]
- [hel, 2014]
- [Lakos and Petrucci, 2004]
- [Klai and Ochi, 2012]
Goal of the GDR GPL Project

Goals

1. **Integrate more tools** into the *CosyVerif* platform
2. **Establish an integration procedure** that will benefit for other tools to be integrated in *CosyVerif*

Use of the fundings: 2 interns hired:

- Henoc Khouilla (LIPN)
- Idrissa Sokhona (LIP6)
Tool for invariant computation for Petri nets

Status before the work

- Not integrated, but integrated into CPN-AMI (ancestor of CosyVerif)

Work achieved

- Integrating the functions of the tool (Place invariants, Transition invariants, Minimal syphon, Minimal traps)
- Translating the internal CosyVerif format into the one of CPN-AMI, thus enabling the reuse of the previous translators
- Testing and benchmarking was done to access the new integration’s results compared to ones provided by CPN-AMI
Integration of ModGraph

Tool for construction and analysis of modular state spaces [Lakos and Petrucci, 2004]

Status before the work

- Previously integrated in CosyVerif, but it provided only a poor user interface

Work achieved

- Upgrade the ModGraph service to the latest version of the tool;
- Enhance the user interface provided by the service.
BDD-based tool implementing a verification approach for workflows using Symbolic Observation Graphs [Haddad et al., 2004, Klai and Ochi, 2012]

Status before the work
- Previously integrated in CosyVerif, but it provided only a poor user interface

Work achieved
- Upgrade the ObsGraph service to the latest version of the tool;
- Enhance the user interface provided by the service;
- Upgrade the service by interaction with the tool developer, for instance the addition of new services above the ObsGraph tool.
Integration of Helena

Explicit state model checker (a High-level Petri net is used for models) [hel, 2014]

Status before the work
- Not integrated

Work achieved
- First attempt to the integration of Helena in CosyVerif
- A prototype was obtained, but not polished enough to be released yet. (Integrating Helena is difficult because a translation from the CosyVerif model format to Helena’s one must be defined.)
Univ. Pierre & Marie Curie
- course attended by 25 students
- students had to provide a small individual project as homework
- only issue (first practical session only, since patch was then provided): misuse of the permissions leading to a crash, due to Eclipse embedded libraries for Coloane
- students downloaded the bundle and provided their project on time

Univ. Paris 13
- course attended by 20 students
- anonymous aftercourse evaluation
- 87% satisfied or very satisfied by their experience
Recent and Ongoing Evolutions

- Asynchronous tool invocation
  - Get the result later (e.g. by email)

- Federation of servers and use of clusters
  - Enable load balancing

- Repository of formalisms and models

- Command-line version of the underlying platform
Future Evolutions

- **Enhanced interaction** between tools
  - Output of a tool as input of another one

- Handling **semantics** (bridges between formalisms)
  - Also allows system simulation

- Handling **heterogeneous models** (mixing different formalisms)

[Try it!](http://cosyverif.org/)
Future Evolutions

- **Enhanced interaction** between tools
  - Output of a tool as input of another one

- **Handling semantics** (bridges between formalisms)
  - Also allows system simulation

- **Handling heterogeneous models** (mixing different formalisms)

Try it!

http://cosyverif.org/
Bibliography
(2014).
Helena.

A modular approach for reusing formalisms in verification tools of concurrent systems.

IMITATOR 2.5: A tool for analyzing robustness in scheduling problems.

CosyVerif: An open source extensible verification environment.

Teaching formal methods: Experience at UPMC and UP13 with CosyVerif.
To appear.


Modular analysis of systems composed of semiautonomous subsystems.