Leveraging Parameters within Model Checking to Verify Distributed Software Deployment Properties by Composition

Hélène Coullon*, Didier Lime†

September 9, 2020

Keywords: petri Nets, Model Checking, Temporal Logic, Distributed systems, Deployment.

Description

With the advent of cloud computing, large computer systems made up of multiple components interacting across machines and networks have become commonplace. Because of their complexity, deploying such systems to reach a usable state is often a difficult task. It becomes even more complex when considering the possibility to perform some steps in parallel, a necessity to reduce deployment time and maximize service. Commercial tools are available to coordinate deployments, but they lack a proper execution model. It is therefore difficult to check that scripts for these tools do not contain errors that could result in failed deployments or incorrect configurations.

Madeus [1] is a formal model that can be used to represent the components of a distributed system and their evolution from an initial state to the end of a deployment. In this model, an *assembly* is a group of components that can execute actions and communicate with each other using ports. Multiple components can execute in parallel, and a single component can also perform multiple actions in parallel, which makes Madeus a very flexible model to describe efficient deployment procedures.

This model can be used to check essential properties of deployments, such as ensuring that the desired state is reached and verifying that the system never enters an invalid state. It can also help optimize deployments, for example by estimating the time needed before completion and identifying the critical path. To perform this analysis automatically, it is useful to transform the Madeus representation into an equivalent Petri net [6], another model widely used to represent distributed systems. There exists a range of model-checking tools that can be used to verify the properties of the resulting Petri net.

In particular we have proposed an automatic transformation to obtain a time Petri net [5] corresponding to a given Madeus assembly [2], and we have introduced qualitative and quantitative properties that a developer, or a system administrator, may found useful for helping her designing its parallel distributed software deployment. We have used the model checker Romeo to this purpose [4]. However, this approach is limited because of scalability issues. The purpose of the internship is to study mechanisms to enhance the scalability of the approach, in particular by leveraging *parameters* in time Petri nets [3] as a mechanism for compositional model checking.

Tasks

Depending on time, the following may be carried out during the internship:

1. understand the formalism of Madeus and its transformation to time Petri nets;

*STACK team, `http://stack.inria.fr/`, IMT Atlantique & Inria Rennes Bretagne Atlantique & LS2N
†team STR, `https://www.ls2n.fr/equipe/str/`, Ecole Centrale de Nantes & LS2N

2. study the related work of compositional model checking;
3. study how to use parameters of time Petri nets to handle compositional model checking for Madeus;
4. study the new transformation and prove its correctness;
5. conduct experiments to show the scalability of the approach.

Skills

The candidate must have an interest in formal models and their semantics. Basic knowledge on model-checking is appreciated, as well as an interest of the candidate for distributed software systems and their deployment.

Practical Information

The internship will be colocated in the STR team of the LS2N at École Centrale de Nantes, and the INRIA/IMT/LS2N team STACK, located at IMT Atlantique in Nantes. For further information about the project, please contact:

- Hélène Coullon, `helene.coullon@inria.fr`, `http://helene-coullon.fr/`
- Didier Lime, `didier.lime@ec-nantes.fr`, `http://pagesperso.ls2n.fr/~lime-d/`

References

[1] Maverick Chardet, Hélène Coullon, Dimitri Pertin, and Christian Pérez. Madeus: A formal deployment model. In *4PAD 2018 - 5th International Symposium on Formal Approaches to Parallel and Distributed Systems (hosted at HPCS 2018)*, pages 1–8, Orléans, France, July 2018.

[2] Hélène Coullon, Claude Jard, and Didier Lime. Integrated Model-checking for the Design of Safe and Efficient Distributed Software Commissioning. In *IFM 2019 - 15th International Conference on integrated Formal Methods*, Integrated Formal Methods, pages 120–137, Bergen, Norway, December 2019.

[3] Didier Lime, Olivier H. Roux, and Charlotte Seidner. Parameter synthesis for bounded cost reachability in time petri nets. In Susanna Donatelli and Stefan Haar, editors, *40th International Conference on Application and Theory of Petri Nets and Concurrency (Petri Nets 2019)*, volume 11522 of *Lecture Notes in Computer Science*, pages 406–425, Aachen, Germany, June 2019. Springer.

[4] Didier Lime, Olivier H. Roux, Charlotte Seidner, and Louis-Marie Traonouez. Romeo: A parametric model-checker for petri nets with stopwatches. In *Tools and Algorithms for the Construction and Analysis of Systems, 15th International Conference, TACAS 2009, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2009, York, UK, March 22-29, 2009. Proceedings*, 2009.

[5] P. M. Merlin. *A study of the recoverability of computing systems.* PhD thesis, Dep. of Information and Computer Science, University of California, Irvine, CA, 1974.

[6] Carl Adam Petri. *Kommunikation mit Automaten.* Dissertation, schriften des iim, Rheinisch-Westfälisches Institut für Instrumentelle Mathematik an der Universität Bonn, Bonn, 1962.