

# Sujet de Stage Master recherche

## Vérification modulaire de systèmes temporisés : aspects structurels et dynamiques

Kais Klai et Laure Petrucci

LIPN, CNRS UMR 7030  
Université Paris 13  
99 avenue Jean-Baptiste Clément  
F-93430 Villetaneuse, France  
{kais.klai,laure.petrucci}@lipn.univ-paris13.fr

### 1 Contexte scientifique et problématique

La vérification et la validation de systèmes informatiques constitue aujourd'hui un enjeu crucial. C'est en particulier le cas pour les systèmes critiques dont les pannes peuvent avoir des conséquences irréversibles et dramatiques sur leur environnement (explosion d'Ariane 5, systèmes de suivi médical, systèmes de missions avioniques, ...). Dans ce cadre, une solution consiste à concevoir une spécification comportementale du système étudié, à l'aide de méthodes formelles, puis à analyser le modèle obtenu.

Divers formalismes permettent d'écrire de telles spécifications. Parmi ceux-ci, les réseaux de Petri présentent l'avantage d'écrire des modèles compacts et lisibles tout en ayant une sémantique précise. Cela permet de disposer d'outils d'analyse pour vérifier les propriétés que le système doit satisfaire.

Les techniques d'analyse principales se répartissent en deux catégories : comportementales et structurelles. Les premières consistent à examiner de manière exhaustive l'ensemble des états que le système peut atteindre (model-checking), à partir d'un état initial, alors que les techniques structurelles n'en dépendent pas, et ne passent pas par cette énumération d'états atteignables. Le phénomène de "l'explosion combinatoire de l'espace d'états", auquel on se retrouve confronté lorsque l'on étudie des cas de taille réelle peut être contré en utilisant diverses techniques de réduction. Une des approches consiste à tirer parti de la structure modulaire de la spécification, lors de l'analyse.

Différentes techniques modulaires ont été développées pour des systèmes modélisés à l'aide de réseaux de Petri (voir par exemple [3,1]). Elles mettent en jeu à la fois des aspects structurels et comportementaux.

Les systèmes complexes tels que ceux issus des transports requièrent non seulement des validations qualitatives, qui peuvent être assurées par les techniques existantes, mais aussi une validation quantitative prenant en compte des contraintes temporelles (timeout, ...). Les réseaux de Petri temporisés permettent de modéliser ces systèmes. Une première extension des travaux de [3] au cas temporisé a été présentée dans [2]. Toutefois, la vérification de propriétés n'est pas encore résolue. L'approche développée dans [1] s'appuie également sur des techniques structurelles, qui ne sont pas disponibles dans le cadre temporisé.

### 2 Objectif du stage

L'objectif de ce stage de master est d'étudier l'extension des méthodes d'analyse modulaire, citées ci-dessus, aux réseaux de Petri temporisés. L'ajout du temps complexifie nettement le problème. En particulier, les contraintes temporelles peuvent empêcher des comportements qui pourraient avoir lieu dans un cas non temporisé.

### 3 Profil souhaité

Le candidat recherché devra avoir des connaissances de base sur le modèle des réseaux de Petri, sur la logique temporelle à temps linéaire et sur la vérification de systèmes (model-checking). Il devra également maîtriser un langage de programmation.

## Références

1. Kais Klai, Laure Petrucci, and Michel Reniers. An incremental and modular technique for checking LTL-X properties of Petri nets. In *Formal Techniques for Networked and Distributed Systems - FORTE, 27th IFIP WG 6.1 International Conference*, volume 4574 of *Lecture Notes in Computer Science*, pages 280–295. Springer, 2007.
2. C. Lakos and L. Petrucci. Modular state space exploration for timed Petri nets. *Int. J. Softw. Tools Technol. Transf.*, 9(3) :393–411, 2007.
3. Charles Lakos and Laure Petrucci. Modular analysis of systems composed of semiautonomous subsystems. In *4th International Conference on Application of Concurrency to System Design (ACSD 2004), 16-18 June 2004, Hamilton, Canada*, pages 185–196, 2004.