Journée Farman

Jeudi 8 octobre 2009

SIMOP

Synergie Simulation et Model-Checking Paramétré

Étienne André, Thomas Chatain, Emmanuelle Encrenaz, Laurent Fribourg

Laboratoire Spécification et Vérification

LSV, ENS de Cachan & CNRS, France

Saïd Amari, Olivier De Smet, Bruno Denis, Silvain Ruel

Laboratoire Universitaire de Recherche en Production Automatisée

LSV, ENS de Cachan, Cnam, France

Membres du projet SIMOP

- Projet Farman 2007–2009
- Laboratoires membres
  - Laboratoire Spécification et Vérification
  - Laboratoire Universitaire de Recherche en Production Automatisée

Saïd Amari
(LURPA)

Étienne André
(LSV)

Thomas Chatain
(LSV)

Olivier De Smet
(LURPA)

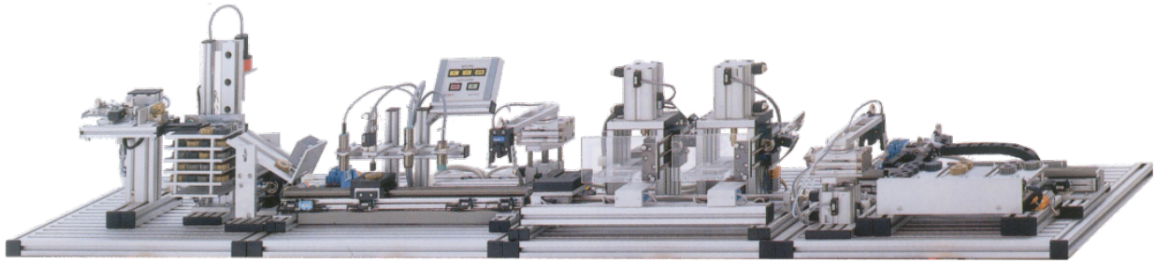
Bruno Denis
(LURPA)

Emmanuelle Encrenaz
(LSV / LIP6)

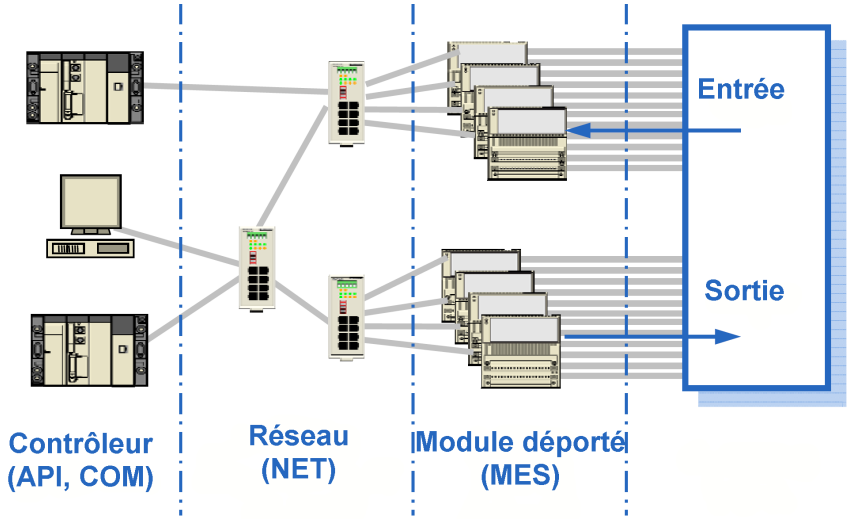
Laurent Fribourg
(LSV)

Silvain Ruel
(LURPA)

Contexte : vérification d'un système distribué



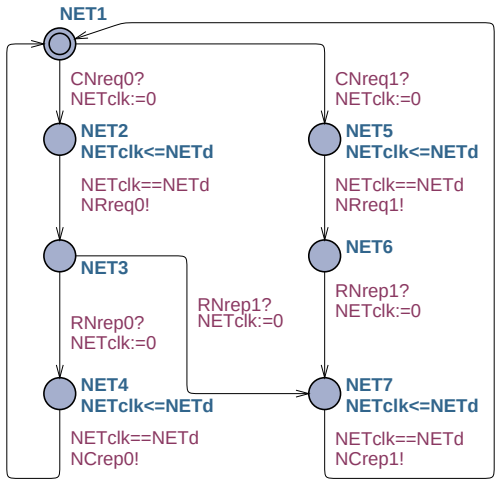
- Architecture d'automatisation en réseau



  - Contrôleurs modulaires (API) : processeur de calcul
  - Processeur de communication (COM) : interroge le MES
  - Réseau (NET) : communication entre API et MES
  - Module déporté d'entrées–sorties (MES)

Modélisation du système

- Automates temporisés
  - Automates d'états finis avec horloges (écoulement du temps)
  - Modélisation de chaque module par un automate temporisé



- Paramètres temporels du système (constantes ajustables)

PLCct *COMct* *SIGmrt* *PLCmtt*
RIOd *COMd* *NETd*

Objectif du projet

- Détermination d'une instance de référence des paramètres temporels

$$PLCct = 300 \quad COMct = 1000 \quad SIGmrt = 2071 \quad PLCmtt = 100$$
$$RIOd = 70 \quad COMd = 25 \quad NETd = 10$$

  - Vérification que ces valeurs correspondent à un bon comportement du système

- Question : le système sera-t-il toujours correct si l'on change des valeurs ?

Objectif du projet

- Détermination d'une instance de référence des paramètres temporels

$$PLCct = 300 \quad COMct = 1000 \quad SIGmrt = 2071 \quad PLCmtt = 100$$
$$RIOd = 70 \quad COMd = 25 \quad NETd = 10$$

  - Vérification que ces valeurs correspondent à un bon comportement du système

- Question : le système sera-t-il toujours correct si l'on change des valeurs ?

Objectif

Déterminer des instances des paramètres correspondant à un bon comportement du système.

Objectif du projet

- Détermination d'une instance de référence des paramètres temporels

$$PLCct = 300 \quad COMct = 1000 \quad SIGmrt = 2071 \quad PLCmtt = 100$$
$$RIOd = 70 \quad COMd = 25 \quad NETd = 10$$

  - Vérification que ces valeurs correspondent à un bon comportement du système

- Question : le système sera-t-il toujours correct si l'on change des valeurs ?

Objectif

Déterminer des instances des paramètres correspondant à un bon comportement du système.

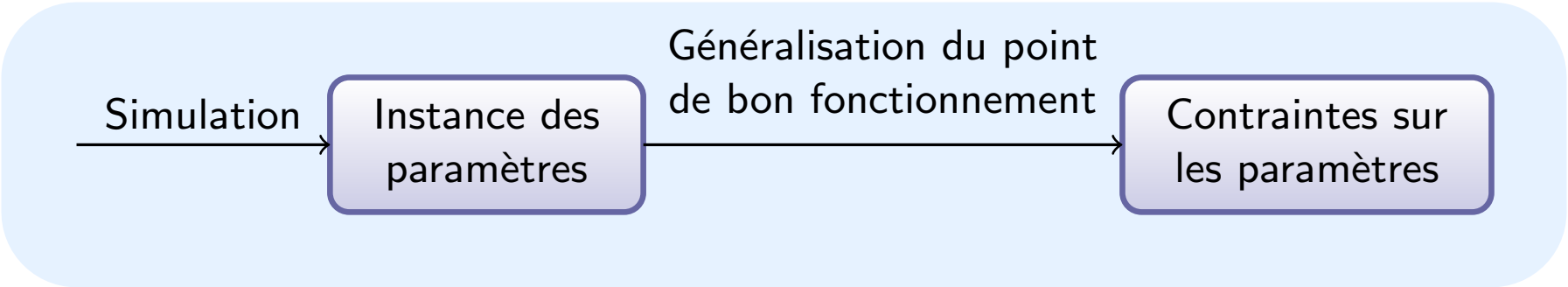
Deux approches :

1. Méthode dichotomique
2. Synthèse de contrainte

Méthode 1 : exploration dichotomique

- Utilisation de l'outil UPPAAL
  - Model-checker pour automates temporisés
- Permet de savoir, pour une instance des paramètres donnée, si le modèle a, ou non, un bon fonctionnement
- Écriture d'un script pour tester automatiquement un grand nombre de points
- Obtention d'un nuage de points de bon fonctionnement

Méthode 2 : synthèse de contrainte (1/2)



Méthode 2 : synthèse de contrainte (2/2)

- Modélisation à l'aide d'automates temporisés paramétrés
- Application de la méthode inverse

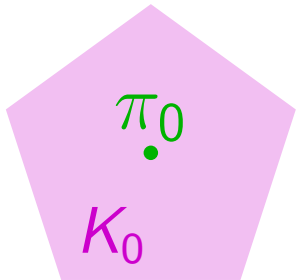
Méthode 2 : synthèse de contrainte (2/2)

- Modélisation à l'aide d'automates temporisés paramétrés
- Application de la méthode inverse
  - Entrées
    - ★ Le système modélisé par un automate temporisé paramétré
    - ★ L'instance de référence π_0

π_0

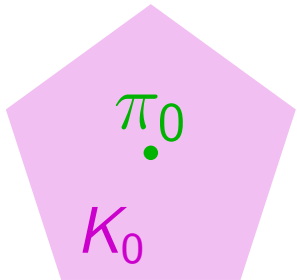
Méthode 2 : synthèse de contrainte (2/2)

- Modélisation à l'aide d'automates temporisés paramétrés
- Application de la méthode inverse
  - Entrées
    - ★ Le système modélisé par un automate temporisé paramétré
    - ★ L'instance de référence π_0
  - Sortie : généralisation de l'instance de référence
    - ★ Une contrainte K_0 sur les 7 paramètres généralisant le comportement de l'instance



Méthode 2 : synthèse de contrainte (2/2)

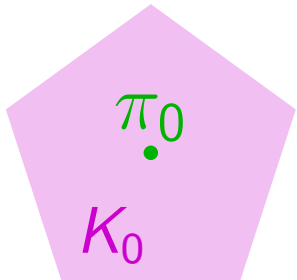
- Modélisation à l'aide d'automates temporisés paramétrés
- Application de la méthode inverse
  - Entrées
    - ★ Le système modélisé par un automate temporisé paramétré
    - ★ L'instance de référence π_0
  - Sortie : généralisation de l'instance de référence
    - ★ Une contrainte K_0 sur les 7 paramètres généralisant le comportement de l'instance



  - Exécution automatique : outil IMITATOR [And09]

Méthode 2 : synthèse de contrainte (2/2)

- Modélisation à l'aide d'automates temporisés paramétrés
- Application de la méthode inverse
  - Entrées
    - ★ Le système modélisé par un automate temporisé paramétré
    - ★ L'instance de référence π_0
  - Sortie : généralisation de l'instance de référence
    - ★ Une contrainte K_0 sur les 7 paramètres généralisant le comportement de l'instance



  - Exécution automatique : outil IMITATOR [And09]
- Notre contrainte K_0 en 3 dimensions ($COMct$, $PLCct$ et $SIGmrt$)

$$\begin{array}{rrcl} & SIGmrt & > & 2COMct + 70 \\ \wedge & 10PLCct & < & 3COMct + 10 \\ \wedge & 2COMct & < & 6PLCct + 205 \\ \wedge & 7PLCct & < & 2COMct + 105 \end{array} \quad \begin{array}{rrcl} \wedge & COMct & > & 3PLCct + 90 \\ \wedge & 7PLCct & > & 2COMct + 90 \\ \wedge & 3COMct & < & 10PLCct + 10 \end{array}$$

Comparaison entre les deux méthodes

- Méthode 1 : exploration dichotomique
  - Avantage : grand nombre de points
  - Inconvénient : pas d'information sur les valeurs entre les points traités
- Méthode 2 : synthèse de contrainte
  - Avantage : zone dense en 7 dimensions
  - Inconvénient : zone plus petite que le nuage de points de la méthode 1

Remarques finales

- Synergie entre simulation et model-checking
  - Approche dichotomique
  - Génération d'une contrainte
- Communications
  - Publication à MSR'09 : *Synthèse de contraintes temporisées pour une architecture d'automatisation en réseau* [ACD+09]
  - Thèse de Silvain Ruel : *Évaluation des bornes des performances temporelles des Architectures d'Automatisation en Réseau par preuves itératives de propriétés logiques* [Rue09]
- Directions de recherches futures
  - Améliorer la taille de la zone dense donnée par la contrainte synthétisée
    - ⋆ Méthode itérative en tirant profit de l'approche dichotomique

Références

Étienne André, Thomas Chatain, Olivier De Smet, Laurent Fribourg, and Silvain Ruel.
Synthèse de contraintes temporisées pour une architecture d'automatisation en réseau.
In Olivier H. Roux, editor, *MSR'09*, Nantes, France, November 2009. Hermès.

Étienne André.
IMITATOR : A tool for synthesizing constraints on timing bounds of timed automata.
In Martin Leucker and Carroll Morgan, editors, *ICTAC'09*, volume 5684 of *Lecture Notes in Computer Science*, pages 336–342, Kuala Lumpur, Malaysia, August 2009. Springer.

Silvain Ruel.

Évaluation des bornes des performances temporelles des architectures d'automatisation en réseau par preuves itératives de propriétés logiques, 2009.