ACI Sécurité Informatique PERSÉE

www.labri.fr/Perso/~herbrete/persee/

---

rapport à mi-parcours

Ph. Schnoebelen

Lab. Spécification & Vérification
CNRS & ENS de Cachan

mai 2005, révisé novembre 2005

Table des matières

1 Description générale

Le projet PERSÉE vise à développer des techniques de vérification symbolique pour les *systèmes hétérogènes*, c.-à-d. des systèmes modélisés par un graphe de contrôle agissant sur des variables non-bornées et de types différents. Ces systèmes sont des modèles naturels pour les systèmes embarqués, les protocoles, les algorithmes distribués, les systèmes sur puce, etc.

Les principaux axes de travail envisagés concernent :

- la mise au point de modèles, de méthodes symboliques (en particulier à base d'accélérations), et de structures de données avancés ;
- la construction de modèles abstraits de haut niveau ;
- la proposition d'un cadre pour la combinaison de représentations symboliques ;
- le développement d'un environnement permettant d'utiliser différents outils symboliques en combinaison.

Le but du projet est aussi de favoriser des échanges et des synergies entre des équipes reconnues au plan international, dont les compétences sont complémentaires, mais qui n'avaient pas encore pris l'habitude de travailler en collaboration.

2 Participants

Les chercheurs participant au projet PERSÉE sont issus des trois équipes suivantes :

1. L'axe INFINI du Laboratoire Spécification & Vérification (LSV) à Cachan :
   - Ph. Schnoebelen, DR, CNRS ;
   - A. Finkel, PU, ENS Cachan ;
   - D. Nowak, CR, CNRS ;
   - A. Halbert, IE, ENS Cachan ;
   - Ch. Darlot, postdoctorant (de sep. 2003 à août 2004) ;
   - J. Leroux, postdoctorant[1] ;
   - S. Bardin, doctorant ;
   - N. Bertrand, doctorante (depuis oct. 2003).
2. L'équipe Modélisation et Vérification du Laboratoire d'Informatique Algorithmique, Fondements et Applications (LIAFA) à Paris 7 :
   - A. Bouajjani, PU Paris 7 ;
   - P. Habermehl, MC Paris 7 ;
   - Y. Jurski, MC Paris 7 ;
   - M. Sighireanu, MC Paris 7 ;
   - T. Touili, doctorante puis CR CNRS ;
   - A. Meyer, doctorant (depuis oct. 2003) ;
   - P. Moro, doctorant (depuis oct. 2003).
3. L'équipe MVTSI du Laboratoire Bordelais de Recherche en Informatique (LaBRI) à Bordeaux :
   - G. Sutre, CR CNRS ;
   - J.-M. Couvreur, MC Bordeaux (jusqu'à sep. 2005) ;
   - F. Herbreteau, MC Bordeaux ;

[1] J. Leroux, ancien doctorant du LSV, a effectué un séjour postdoctoral à Montréal en 2003–2004, puis un séjour postdoctoral à Rennes en 2004–2005. Durant ces deux années, il a poursuivi ses collaborations avec A. Finkel et S. Bardin dans le cadre de PERSÉE. Il a ensuite été recruté comme CR CNRS au LaBRI dans l'équipe MVTSI.

– A. Griffault, MC Bordeaux ;
– K. Musumbu, MC Bordeaux ;
– A. Vincent, doctorant puis MC Bordeaux ;
– M. Adélaïde, ATER (de sep. 2002 à sep. 2004) ;
– The Quang Tran, doctorant (depuis oct. 2004).

À ce jour, cinq réunions plénières rassemblant l'essentiel des participants ont eu lieu :

1. le 24 nov 2003 à Paris,
2. le 15 mar 2004 à Paris,
3. les 7 et 8 juin 2004 à Cachan,
4. les 14 et 15 septembre 2004 à Bordeaux,
5. le 26 mai 2005 à Paris.

Le programme de ces journées est disponible sur le site Web du projet.

D'autres réunions, n'impliquant en général qu'un groupe de travail, se sont tenues par ailleurs.

3 Résultats scientifiques

3.1 Calcul symbolique d'accessibilité

3.1.1 Théorie des accélérations

Une théorie des accélérations plates a été développée dans [BFLS05]. Ce travail a permis de formaliser certaines des notions sous-jacentes à l'approche développée au LSV pour l'analyse symbolique des systèmes complexes, et de dégager des principes heuristiques d'application générale, et qui sont en particulier appliqués dans l'outil FAST [BFL04].

Une retombée théorique très surprenante est due à Leroux et Sutre, qui ont montré qu'un très grand nombre de classes de systèmes à compteurs pour lesquels l'ensemble d'accessibilité avait été prouvé semilinéaire et effectif sont en fait des systèmes aplatissables [LS04, LS05]. Ainsi, notre algorithme générique d'accélération plate (et donc l'outil FAST) s'applique à ces systèmes, est garanti de terminer, et remplace utilement les algorithmes ad-hoc proposés indépendamment pour chacune des classes.

3.1.2 Au-delà des propriétés de sûreté

Les techniques classiques permettant de réduire la vérification des propriétés de vivacité à des questions d'accessibilité ne s'étendent pas aux systèmes infinis. Les réponses à cette difficulté sont en général en partie ad-hoc, dépendant à la fois d'une famille de modèles, d'une classe de propriétés, et d'une technique algorithmique de vérification.

L'étude des aspects pratiques de cette question dans un cadre de *regular model checking* est abordée dans [BLW04], où les configurations des systèmes peuvent être représentées par des mots finis arbitrairement longs, voire par des mots *infinis* dans le cadre du *omega-regular model checking*.

Une autre contribution est fournie par [BS04, ABRS05, BBS05, BBS06], où les modèles sont des processus de décision markoviens modélisant des protocoles asynchrones dans lesquels les pertes de messages obéissent à des lois probabilistes naturelles.

3.2 Représentation symbolique des ensembles de configurations

3.2.1 Nouvelles représentations symboliques

Pour l'analyse des programmes manipulant des pointeurs, les techniques de vérification automatique sont encore dans une phase très exploratoire. Dans le cadre du projet PERSÉE, les techniques de *regular model checking* ont été mises en œuvre via des représentations linéaires spéciales des configurations des pointeurs [BHMV05]. Une autre approche est basée sur les SMS, une structure de données symbolique originale, à base de graphes pondérés par des expressions arithmétiques [BFN04].

Pour l'analyse de programmes récursifs multi-threads, les approches symboliques sont en général des extensions des pushdown-systems pouvant aller jusqu'au *process rewrite systems* de Mayr et au-delà. Ces extensions peuvent varier en complexité suivant qu'on autorise ou non la création dynamique de threads et qu'on permet plus ou moins de communication et de synchronisation entre les threads. Autour d'A. Bouajjani, les chercheurs de l'équipe Modélisation et Vérification ont contribué puissamment à cette direction de recherche [BT03, BET04, BM04, BT05, BMOT05].

3.2.2 Algorithmique des représentations symboliques

L'amélioration des performances de nos model-checkers passe aussi par les progrès algorithmiques au niveau des représentations symboliques.

Dans cette direction, le projet PERSÉE est à l'origine de deux résultats importants :

1. la mise au point par J.-M. Couvreur d'une notion d'*automates partagés*, une représentation canonique des automates finis déterministes qui permet de voir un automate comme le dag de ses composantes fortement connexes et de partager les sous-arbres identiques entre ses dags [Cou05].
2. l'étude fine de l'algorithmique des *Unambiguous Binary Automata* (UBA), cf. [FL05, FL04]. Cette étude a conduit à un résultat dérivé exceptionnel, la résolution par J. Leroux du problème classique de la synthèse efficace d'une formule de Presburger définissant une partie reconnaissable de $\mathbb{N}^m$ [Ler05].

3.2.3 Combinaison de représentations symboliques

L'analyse symbolique de systèmes manipulant des variables de différents types pose la question de combiner les représentations symboliques associées aux différents types. Cette question est bien connue et soulève des problèmes algorithmiques ardus. Dans le cadre du model-checking symbolique, elle a été abordée sous l'angle des calculs de successeurs (ou prédécesseurs) immédiats, combinés aux tests d'inclusion et de vacuité.

Le projet PERSÉE a élargi cette question en prenant aussi en compte la dimension des algorithmes d'accélération. Ainsi, dans [BHJ03] les techniques d'accélérations combinent les aspects discrets et les aspects continus de systèmes hybrides.

La problématique générale est considérée dans [BF04] où sont donnés des premiers résultats prometteurs sous la forme de critères génériques suffisants pour la combinaison effective de résultats d'accélération.

3.3 Abstractions et méthodes symboliques

Pour l'analyse de systèmes de grande taille, il est parfois indispensable de mettre en œuvre des techniques d'abstractions les plus automatiques possible.

Dans un cadre de *regular model-checking*, [HV04] montre comment des techniques de généralisation d'automates à partir d'exemples et de contre-exemples permettent d'obtenir des approximations supérieures très pertinentes des ensembles d'accessibilité.

Dans le même cadre, [BHV04] montre comment combiner la démarche « *abstract–check–refine* » de [HJMS02, HJMS03] avec le *regular model checking*.

Dans [Ler04], les systèmes manipulent des compteurs et les approximations supérieures sont des disjonctions d'espace affines calculées automatiquement.

Dans [AS04], des systèmes hybrides d'un type particulier permettent d'analyser certains processus biologiques au moyen des techniques de model-checking symbolique.

3.4 Environnement intégré pour la vérification symbolique

Un des objectifs du projet PERSÉE est de construire un outil de model-checking symbolique "ouvert" permettant de combiner (en les réutilisant) les structures de données, les représentations symboliques et les algorithmes, d'outils tels que TReX [ABS01] et FAST [BFLP03] et Mec 5 [GV04] (pour commencer).

Dans cette direction, les travaux d'ores et déjà réalisés sont :

- La spécification formelle des interfaces génériques et du langage de scripts pour les stratégies symboliques [GHP+05].
- L'extension du langage AltaRica [GV03, GPV04] de façon à ce qu'il prenne en compte des types de données définies par l'utilisateur et puisse ainsi manipuler les modèles considérés au sein du projet.
- La refonte de l'architecture logicielle de TReX de façon à ce que cet outil s'adapte à l'interface définie plus haut.

La continuation de ce travail, prévue pour la fin du projet, est naturellement l'intégration des différents composants. La preuve de faisabilité du concept s'obtiendra en vérifiant un protocole complexe, par exemple le protocole BRP, au moyen de la chaîne d'outils intégrés.

4 Autres éléments d'appréciation

Journées Systèmes Infinis

Le projet PERSÉE a été l'organisateur des cinquièmes Journées Systèmes Infinis (JSI 2005), qui se sont tenues à Cachan les 10 et 11 mai 2005. Les JSI sont une manifestation irrégulière démarrée en 1998, qui permet de rassembler la communauté française travaillant sur la vérification des systèmes infinis, et de rencontrer quelques uns des chercheurs étrangers actifs dans le domaine.

Nous renvoyons à `www.lsv.ens-cachan.fr/~phs/jsi2005.php` pour le programme des journées.

Mobilité interne au projet

Le projet PERSÉE a favorisé la mobilité entre les équipes participantes :

- J.-M. Couvreur, MC Bordeaux au LaBRI, a effectué une délégation au CNRS au sein du LSV (de sep. 2002 à août 2004) ;

– J. Leroux, doctorant du LSV a rejoint le LaBRI en sep. 2005 via le concours de recrutement des CR CNRS.

Références

[ABRS05] P. A. Abdulla, N. Bertrand, A. Rabinovich, and Ph Schnoebelen. Verification of probabilistic systems with faulty communication. *Information and Computation*, 202(2) :141–165, 2005.

[ABS01] A. Annichini, A. Bouajjani, and M. Sighireanu. TReX : A tool for reachability analysis of complex systems. In *Proc. 13th Int. Conf. Computer Aided Verification (CAV 2001), Paris, France, July 2001*, volume 2102 of *Lecture Notes in Computer Science*, pages 368–372. Springer, 2001.

[ACBJ04] P. A. Abdulla, A. Collomb-Annichini, A. Bouajjani, and B. Jonsson. Using forward reachability analysis for verification of lossy channel systems. *Formal Methods in System Design*, 25(1) :39–65, 2004.

[AS04] M. Adélaïde and G. Sutre. Parametric analysis and abstraction of genetic regulatory networks. In *Proc. 2nd Workshop on Concurrent Models in Molecular Biology (BioCONCUR'04), London, UK, Aug. 2004*, Electronic Notes in Theor. Comp. Sci. Elsevier Science, 2004. To appear.

[BBS05] C. Baier, N. Bertrand, and Ph. Schnoebelen. Verifying nondeterministic probabilistic channel systems against ω-regular linear-time properties. Paper cs.LO/0511023, Computing Research Repository (CoRR), November 2005.

[BBS06] C. Baier, N. Bertrand, and Ph. Schnoebelen. A note on the attractor-property of infinite-state Markov chains. *Information Processing Letters*, 97(2) :58–63, 2006.

[BET04] A. Bouajjani, J. Esparza, and T. Touili. Reachability analysis of synchronized PA systems. In *Proc. 6th Int. Workshop on Verification of Infinite State Systems (INFINITY 2004), London, UK, Sep. 2004*, 2004.

[BF04] S. Bardin and A. Finkel. Composition of accelerations to verify infinite heterogeneous systems. In *Proc. 2nd Int. Symp. Automated Technology for Verification and Analysis (ATVA 2004), Taipei, Taiwan, Nov. 2004*, volume 3299 of *Lecture Notes in Computer Science*, pages 248–262. Springer, 2004.

[BFL04] S. Bardin, A. Finkel, and J. Leroux. FASTer acceleration of counter automata in practice. In *Proc. 10th Int. Conf. Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2004), Barcelona, Spain, Apr. 2004*, volume 2988 of *Lecture Notes in Computer Science*, pages 576–590. Springer, 2004.

[BFLP03] S. Bardin, A. Finkel, J. Leroux, and L. Petrucci. FAST : Fast Acceleration of Symbolic Transition systems. In *Proc. 15th Int. Conf. Computer Aided Verification (CAV 2003), Boulder, CO, USA, July 2003*, volume 2725 of *Lecture Notes in Computer Science*, pages 118–121. Springer, 2003.

[BFLS05] S. Bardin, A. Finkel, J. Leroux, and Ph. Schnoebelen. Flat acceleration in symbolic model checking. In *Proc. 3rd Int. Symp. Automated Technology for Verification and Analysis (ATVA 2005), Taipei, Taiwan, Oct. 2005*, volume 3707 of *Lecture Notes in Computer Science*, pages 474–488. Springer, 2005.

[BFN04] S. Bardin, A. Finkel, and D. Nowak. Toward symbolic verification of programs handling pointers. In Ramesh R. Bharadwaj, editor, *Proc. 3rd Int. Workshop on Automated Verification of Infinite-State Systems (AVIS 2004), Barcelona, Spain, Apr. 2004*, 2004.

[BHJ03] B. Boigelot, F. Herbreteau, and S. Jodogne. Hybrid acceleration using real vector automata. In *Proc. 15th Int. Conf. Computer Aided Verification (CAV 2003), Boulder, CO, USA, July 2003*, volume 2725 of *Lecture Notes in Computer Science*, pages 193–205. Springer, 2003.

[BHMV05] A. Bouajjani, P. Habermehl, P. Moro, and T. Vojnar. Verifying programs with dynamic 1-selector-linked structures in regular model checking. In *Proc. 11th Int. Conf. Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2005), Edinburgh, Scotland, UK, Apr. 2005*, volume 3440 of *Lecture Notes in Computer Science*, pages 13–39, 2005.

[BHV04] A. Bouajjani, P. Habermehl, and T. Vojnar. Abstract regular model checking. In *Proc. 16th Int. Conf. Computer Aided Verification (CAV 2004), Boston, MA, USA, July 2004*, volume 3114 of *Lecture Notes in Computer Science*, pages 372–386. Springer, 2004.

[BLW04] A. Bouajjani, A. Legay, and P. Wolper. Handling liveness properties in (omega-) regular model checking. In *Proc. 6th Int. Workshop on Verification of Infinite State Systems (INFINITY 2004), London, UK, Sep. 2004*, 2004.

[BM04] A. Bouajjani and A. Meyer. Symbolic reachability analysis of higher-order context-free processes. In *Proc. 24th Conf. Found. of Software Technology and Theor. Comp. Sci. (FST&TCS 2004), Chennai, India, Dec. 2004*, volume 3328 of *Lecture Notes in Computer Science*, pages 135–147. Springer, 2004.

[BMOT05] A. Bouajjani, M. Müller-Olm, and T. Touili. Regular symbolic analysis of dynamic networks of pushdown systems. In *Proc. 16th Int. Conf. Concurrency Theory (CONCUR 2005), San Francisco, CA, USA, Aug. 2005*, volume 3653 of *Lecture Notes in Computer Science*, pages 474–487. Springer, 2005.

[BS04] N. Bertrand and Ph. Schnoebelen. Verifying nondeterministic channel systems with probabilistic message losses. In Ramesh R. Bharadwaj, editor, *Proc. 3rd Int. Workshop on Automated Verification of Infinite-State Systems (AVIS 2004), Barcelona, Spain, Apr. 2004*, 2004.

[BT03] A. Bouajjani and T. Touili. Reachability analysis of Process Rewrite Systems. In *Proc. 23rd Conf. Found. of Software Technology and Theor. Comp. Sci. (FST&TCS 2003), Mumbai, India, Dec. 2003*, volume 2914 of *Lecture Notes in Computer Science*, pages 74–87. Springer, 2003.

[BT05] A. Bouajjani and T. Touili. On computing reachability sets of process rewrite systems. In *Proc. 16th Int. Conf. Rewriting Techniques and Applications (RTA 2005), Nara, Japan, Apr. 2005*, volume 3467 of *Lecture Notes in Computer Science*, pages 484–499. Springer, 2005.

[Cou05] J.-M. Couvreur. A BDD-like implementation of an automata package. In *Proc. 9th Int. Conf. Implementation and Application of Automata (CIAA 2004), Queen's University, Kingston, ON, Canada, July 2004*, volume 3317 of *Lecture Notes in Computer Science*, pages 310–311. Springer, 2005.

[FL04] A. Finkel and J. Leroux. Polynomial time image computation with interval-definable counters systems. In *Model Checking Software, Proc. 11th Int. SPIN Workshop, Bar-*

celona, Spain, Apr. 2004, volume 2989 of *Lecture Notes in Computer Science*, pages 182–197. Springer, 2004.

[FL05] A. Finkel and J. Leroux. The convex hull of a regular set of integer vectors is polyhedral and effectively computable. *Information Processing Letters*, 96(1) :30–35, 2005.

[GHP+05] A. Griffault, F. Herbreteau, G. Point, G. Sutre, A. Vincent (LaBRI), M. Sighireanu (LIAFA), S. Bardin, A. Finkel, and D. Nowak (LSV). Intégration des outils PERSÉE. Livrable RC1, Projet PERSÉE de l'ACI Sécurité Informatique, April 2005. 32 pages.

[GPV04] A. Griffault, G. Point, and A. Vincent. Vérification formelle des modèles AltaRica. In *Actes du Congrès LM (Maîtrise des risques et sûreté de fonctionnement, $\lambda\mu$14)*. Hermès, October 2004.

[GV03] A. Griffault and A. Vincent. Vérification de modèles AltaRica. In *MAJECSTIC : Manifestation des jeunes chercheurs STIC, Marseille, Oct. 2003*, 2003.

[GV04] A. Griffault and A. Vincent. The mec 5 model-checker. In *Proc. 16th Int. Conf. Computer Aided Verification (CAV 2004), Boston, MA, USA, July 2004*, volume 3114 of *Lecture Notes in Computer Science*, pages 488–491. Springer, 2004.

[HJMS02] T. A. Henzinger, R. Jhala, R. Majumdar, and G. Sutre. Lazy abstraction. In *Proc. 29th ACM Symp. Principles of Programming Languages (POPL 2002), Portland, OR, USA, Jan. 2002*, pages 58–70, 2002.

[HJMS03] T. A. Henzinger, R. Jhala, R. Majumdar, and G. Sutre. Software verification with BLAST. In *Model Checking Software, Proc. 10th Int. SPIN Workshop, Portland, OR, USA, May 2003*, volume 2648 of *Lecture Notes in Computer Science*, pages 235–239. Springer, 2003.

[HV04] P. Habermehl and T. Vojnar. Regular model checking using inference of regular languages. In *Proc. 6th Int. Workshop on Verification of Infinite State Systems (INFINITY 2004), London, UK, Sep. 2004*, 2004.

[Ler04] J. Leroux. Disjunctive invariants for numerical systems. In *Proc. 2nd Int. Symp. Automated Technology for Verification and Analysis (ATVA 2004), Taipei, Taiwan, Nov. 2004*, volume 3299 of *Lecture Notes in Computer Science*, pages 93–107. Springer, 2004.

[Ler05] J. Leroux. A polynomial time Presburger criterion and synthesis for number decision diagrams. In *Proc. 20th IEEE Symp. Logic in Computer Science (LICS 2005), Chicago, IL, USA, June 2005*, pages 147–156. IEEE Comp. Soc. Press, 2005.

[LS04] J. Leroux and G. Sutre. On flatness for 2-dimensional vector addition systems with states. In *Proc. 15th Int. Conf. Concurrency Theory (CONCUR 2004), London, UK, Aug.-Sep. 2004*, volume 3170 of *Lecture Notes in Computer Science*, pages 402–416. Springer, 2004.

[LS05] J. Leroux and G. Sutre. Flat counter automata almost everywhere ! In *Proc. 3rd Int. Symp. Automated Technology for Verification and Analysis (ATVA 2005), Taipei, Taiwan, Oct. 2005*, volume 3707 of *Lecture Notes in Computer Science*, pages 489–503. Springer, 2005.

[Mey04] A. Meyer. On term rewriting systems having a rational derivation. In *Proc. 7th Int. Conf. Foundations of Software Science and Computation Structures (FOSSACS 2004), Barcelona, Spain, Apr. 2004*, volume 2987 of *Lecture Notes in Computer Science*, pages 378–392. Springer, 2004.