Proposition de recherche doctorale

Vérification compositionnelle pour la Conception sûre de systèmes embarqués

Mots clés :

- **Directeur de thèse** : Emmanuelle Encrenaz
- **Co-encadrant(s)** :
- **Unité de recherche** : Laboratoire d'informatique de Paris 6
- **Ecole doctorale** : École Doctorale Informatique, Télécommunications, Électronique de Paris
- **Domaine scientifique principal**: Divers

Résumé du projet de recherche (Langue 1)

Contexte Les systèmes embarqués sont bien implantés dans la vie courante. On les retrouve dans toute l'électronique de loisir grand public, mais également dans de nombreux domaines d'applications critiques : l'assistance opératoire et le diagnostic médical, l'automobile (une automobile de moyenne gamme comprend plus d'une vingtaine de controleurs répartis sur le véhicule), le transport ferroviaire ou aérien, ... Ces systèmes embarqués correspondent à l'intégration sur un même circuit, d'un ensemble de fonctionnalités complexes, réalisées par un très grand nombre de composants hétérogènes. Les systèmes sur puce actuels comprennent plusieurs processeurs exécutant de multiples tâches coopératives, des coprocesseurs spécialisés (pour des traitements applicatifs, ou pour les communication réseau), des composants radio-fréquences. Ces systèmes sont fréquemment soumis à des impératifs de sûreté et de fiabilité. Selon leur domaine d'utilisation, leur défaillance peut causer de graves dommages. Il est important de pouvoir s'assurer, lors de leur conception, de leur correction vis-à-vis de leur fonctionalité. Du fait de leur très grande complexité, les méthodes de vérification actuelles ne permettent de vérifier que très partiellement ces systèmes. Les méthodes de validation des systèmes complexes actuellement utilisées reposent presque exclusivement sur le test (fonctionnel), qui n'est pas exhaustif. Les méthodes de vérification automatiques, basées sur l'exploration de l'espace d'état du système (méthodes de vérification par modèle), sont appliquées pour certaines phases bien identifiées du processus de conception des systèmes embarqués, mais force est de reconnaitre que leur emploi reste marginal. Un vérificateur de modèle (ou model-checker) est un programme qui détermine automatiquement si une propriété est vraie ou non pour un système fini donné. Le système fini est décrit comme une collection d'automates synchrones, représenté par une fonction de transition globale indiquant l'état du système à l'instant i+1 en fonction de son état à l'instant i, et la propriété caractérise un ensemble de comportements du système (désirables ou non), exprimée dans une logique temporelle (ici CTL). Les vérificateurs de modèle explorent le graphe des états du système (en utilisant l'état initial et la relation de transition) pour s'assurer que les comportements décrits par la propriété sont présents ou non dans le graphe. En général, les vérifications ne sont pas faites en parcourant les états un à un (il y en a trop), mais en parcourant simultanément des ensembles d'états, représentés symboliquement par leurs fonctions caractéristiques ; On parle de vérification de modèle symbolique. La représentation symbolique la plus utilisée est basée sur les diagrammes de décision binaire [McMillan 93]. Malgré les avancées apportées par cette représentation, les systèmes vérifiables restent de moyenne complexité. Une approche complémentaire consiste à adopter une démarche compositionnelle : on cherche à tirer partie de la structure en composants du système et de la propriété à vérifier, afin de réduire la taille de la représentation symbolique de l'espace d'états, et repousser ainsi les limites de la vérification par modèle. Le département ALSOC du LIP6 a proposé et automatisé une méthode d'abstraction de composants, basée sur l'ensemble des propriétés que vérifie chaque composant [Braunstein 07] [Bara 08]. On souhaite l'intégrer dans une démarche plus générale de Raffinement d'Abstraction Guidée par le Contre-Exemple (CEGAR, [Clarke00]) qui est maintenant bien implantée pour la vérification de programmes C ou JAVA. Par exemple, les outils SLAM ou BLAST, et tout récemment VCEGAR [VCEGAR 07] offrent des possibilités de vérification très intéressantes pour des programmes de moyenne complexité. **Objectifs de la thèse** L'objectif de la thèse est de définir et implanter une méthode de vérification compositionnelle automatique, en s'appuyant sur : la démarche de conception modulaire, dans laquelle chaque composant est défini, spécifié et vérifié avant d'être intégré dans un système plus complexe), la démarche de vérification par abstraction CEGAR, qui permet de repousser les limites de la vérification par modèle en éliminant les variables du systèmes non pertinentes pour la vérification de telle ou telle propriété. L'originalité de cette approche réside en la combinaison de ces deux démarches pour fixer un cadre de conception précis et rigoureux, qui s'inscrit dans une démarche qualité. La conception et la vérification sont intimement couplées : la formalisation de la spécification de chaque composant est nécessaire, et une fois chaque composant vérifié, cette spécification est utilisée pour simplifier la vérification de systèmes composés. Cette démarche sera appliquée à plusieurs cas d'étude, représentant notamment des parties de systèmes embarqués provenant du domaine de l'automobile ou du traitement de flux video. Environnement et collaborations Actuellement, ce thème de recherche n'est pas supporté par un projet. Plusieurs contacts sont néanmoins établis avec le LURPA (ENS-Cachan : conception et validation de systèmes automatisés) et l'équipe ARIS (TIMA – Grenoble : tests et qualification de circuits) qui montrent un intéret pour ces méthodes de conception et vérification. Des collaborations pourront être formalisées autour de ce thème durant la thèse. **Bibliographie** [Mc Millan 93] Symbolic Model-checking. K. Mc Millan. Kluwer Academish Publishers, 1993. [Clarke 00] Counter-Example Guided Abstraction Refinement. E. Clarke et al. CAV'2000. LNCS vol 1855. pp 154-169. Springer-Verlag. [Braunstein 07] Using CTL Formulae as Component Abstraction in a Design and Verification Flow. C. Braunstein, E. Encrenaz. ACSD'2007. pp 80-89. IEEE Computer Society Press. [Bara 08] Abstraction de Composants pour la Vérification par model-Checking. A. Bara. Mémoire de Diplôme Universitaire OMP, 2008, effectué au LIP6/SOC. [VCEGAR 07] VCEGAR : a Verilog Counter-example Guided Abstraction Refinement. H. Jain, D. Kroening et al. TACAS'07, LNCS vol 4424. pp 583-586. Springer-Verlag. [VIS 96] VIS : a System for Verification and Synthesis. R. Brayton et al. CAV'96, LNCS vol 1102, pp 428-432. Springer-Verlag.

Résumé du projet de recherche (Langue 2)

La vérification est un enjeu important dans le processus de développement, de fiabilisation et de certification des systèmes complexes. La thèse propose l'amélioration de la vérification en combinant l'exploitation de la structure du système (vérification compositionnelle) et d'abstractions du système (retirant des informtaions non pertinentes vis à vis d'une propriété à vérifier. Ces points ont été exploités séparément, mais peu de travaux de recherche les combine efficacement. Le LIP6 a proposé une méthode combinant ces deux aspects et souhaite investiguer le potentiel de cette combinaison.