INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Action TICK

Etude et implémentation des systèmes réactifs synchrones

Sophia Antipolis

THÈME 1C

Table des matières

1. Composition de l'équipe 1
2. Présentation et objectifs généraux du projet 1
2.1. (Sans titre) 1
2.1.1. Contexte et Objectifs du projet 1
2.1.2. Axes de recherche 2
2.1.3. Relations internationales et industrielles 2
3. Fondements scientifiques 2
3.1. Programmation Réactive Synchrone et Esterel 2
3.2. Méthodes d'analyse et de vérification automatique de systèmes réactifs synchrones 3
4. Domaines d'application 4
4.1. Télécommunications 4
4.1.1. Programmation réactive synchrones de protocoles 4
4.1.2. Spécification de microcontrôleurs dédiés 4
4.2. Systèmes embarqués en avionique/automobile 4
4.3. Synthèse de circuits 4
4.4. Conception conjointe matériel / logiciel 4
5. Logiciels 5
5.1. Esterel 5
5.2. Xeve 5
6. Résultats nouveaux 6
6.1. Exécution « efficace » logicielle de programmes Esterel 6
6.2. Traduction d'Esterel vers SynDex 6
6.3. Partitionnement structurel pour le calcul symbolique d'états atteignables 6
6.4. Spécification simple de propriétés de correction 7
6.5. Verification par analyse de causalité 7
6.6. Esterel multiphases 7
6.7. Traduction vers des formalismes de description de circuits ou d'architectures de systèmes embarqués 7
6.8. Abstraction de registres dans l'analyse de programmes 7
6.9. Conception « synchrone » d'un système de téléphonie par Internet 8
6.10. Synthèse de programmes structurés 8
6.11. Vérification de moteurs pour les systèmes à base de connaissances 8
7. Contrats industriels 9
7.1. Programmation synchrone pour les télécommunicatons 9
8. Actions régionales, nationales et internationales 9
8.1. Actions nationales 9
8.1.1. Action de recherche coopérative S-JAVA 9
8.1.2. Action Color-2001 Mûre 9
9. Diffusion des résultats 9
9.1. Animation de la Communauté scientifique 9
9.2. Enseignement 10
10. Bibliographie 10

1. Composition de l'équipe

Responsable scientifique
Robert de Simone [directeur de recherches, Inria]

Responsable permanent
Annie Ressouche [chargé de recherche]

Assistante de projet
Catherine Juncker [attachée d'administration de la recherche]

Ingénieur expert
Xavier Thirioux [jusqu'au 31 août]

Chercheurs doctorants
Yannis Bres [allocataire MENESR et moniteur UNSA]
Loïc Henry-Gréard [allocataire AMX MENESR, jusqu'au 30 octobre]
Fabrice Peix [allocataire MENESR et moniteur UNSA]
Dumitru Potop-Butucaru [boursier Eiffel]
Olivier Tardieu [Ingénieur du Corps des Mines, en détachement depuis le 1er septembre]
Eric Vecchié [boursier région PACA]

Collaborateurs extérieurs
Charles André [professeur à l'UNSA]
Férial Virolleau [professeur ESIEE]

Autres personnels
Laurence Pierre [Maître de Conférences UNSA, en délégation]

Stagiaires
Fabrice Giocanti [stagiaire ESIEE, du 1er avril au 31 juillet]
Jérôme Millon [stagiaire EMERIR Perpignan, du 1er avril au 30 septembre]
Ludovic Segarra [stagiaire ESIEE, du 1er avril au 31 juillet]

2. Présentation et objectifs généraux du projet

2.1. (Sans titre)

2.1.1. Contexte et Objectifs du projet

L'action Tick a comme objectif l'étude et l'analyse des systèmes réactifs synchrones, ainsi que de leur implantation effective et efficace. Nous nous basons principalement sur le langage Esterel comme formalisme de représentation. Nous poursuivons l'étude de méthodes algorithmiques, basées sur son environnement de développement et sa sémantique mathématique formelle, ainsi que de logiciels d'analyse et de vérification associés.

La programmation réactive synchrone est particulièrement adaptée aux systèmes temps réel, manipulant des événements logiques avec préemption. Le même formalisme peut modéliser des circuits digitaux, des contrôleurs logiciels, ou des systèmes mixtes incluant logiciel et matériel. Les domaines d'applications principaux sont les systèmes embarqués à forte composante contrôle.

Déjà utilisé chez Dassault Aviation pour l'informatique embarquée de ses avions de combat, par Texas Instruments pour la conception de circuits dédiés en téléphonie mobile, et par Cadence pour la conception conjointe de matériel et de logiciel, Esterel est actuellement en phase d'industrialisation et de commercialisation par la société Esterel Technologies. Nous poursuivons en parallèle des études sur de nouvelles techniques avancées de compilation et d'analyse.

Une version graphique du langage, dénommée SyncCharts, a été conçue par Charles André, membre du projet SPORTS de l'Université de Nice Sophia-Antipolis, et collaborateur extérieur de notre équipe.

2.1.2. Axes de recherche

La compilation actuelle de programmes Esterel utilise une traduction des programmes réactifs en systèmes d'équations booléennes, un formalisme très proche des modèles de circuits digitaux en portes logiques. Ce format intermédiaire est ensuite manipulé par des algorithmes d'analyse, d'optimisation, puis de compilation (logicielle) ou de synthèse (matérielle). Ces algorithmes ont une complexité parfois importante sur de gros programmes, et nous travaillons essentiellement à la conception de nouvelles techniques permettant leur passage à l'échelle, ainsi que la production de code (logiciel) efficace. Ces améliorations exploitent les informations structurelles syntaxiques des programmes. Elles procèdent de techniques d'analyse statique qui doivent être concues pour ce domaine spécifique de la programmation synchrone.

Par ailleurs, nous étudions des extensions au langage pour augmenter son expressivité pratique. Nous étudions également des extensions de nos techniques d'analyse à base de « model-checking » symbolique, en particulier pour le traitement des données par des abstractions appropriées. Enfin, nous étudions l'extension de nos travaux hors du cadre strictement synchrone, en particulier les réseaux asynchrones de processus synchrones, dans des buts d'implantation ou d'analyse.

Nous nous intéressons à la relation entre nos modèles et d'autres formalismes dans lesquels ils peuvent se traduire, en particulier les langages de description de circuits comme VHDL et Verilog, ainsi que les environnements de cosimulation matérielle/logicielle comme VCC ou SynDex.

2.1.3. Relations internationales et industrielles

Le compilateur Esterel (version binaire) est disponible par ftp avec son environnement de simulation graphique Xes et son logiciel d'analyse et de vérification Xeve. Nous avons créé le site `http://www.esterel.org` pour rassembler la communauté d'utilisateurs et organiser la distribution de nos logiciels.

Nous participons au projet RNRT Syntel, avec Simulog, Cadence et Thomson CSF. Nous avons participé au projet Esprit LTR Syrf, sur le développement des formalismes réactifs synchrones.

Nous collaborons régulièrement avec les sociétés Esterel Technologies, Thomson CSF, Cadence et Intel.

Nous participons à l'action de développement AEE, coordonnée par le projet INRIA SOSSO.

3. Fondements scientifiques

3.1. Programmation Réactive Synchrone et Esterel

Participants : Robert de Simone, Loïc Henry-Gréard, Fabrice Peix, Dumitru Potop, Annie Ressouche, Olivier Tardieu, Eric Vecchié.

Cette activité est un thème de collaboration avec l'équipe SPORTS de l'I3S (CNRS/UNSA), dirigée par Charles André.

Mots clés : *Esterel, programmation réactive synchrone, sémantique, causalité, temps réel, compilation, optimisation, circuit digital, conception conjointe*.

Le langage Esterel permet la programmation structurée de systèmes réactifs synchrones. La syntaxe impérative du langage est adaptée aux systèmes dominés par le contrôle. Elle repose sur des primitives spécifiques de parallélisme et de préemption hiérarchisée. La sémantique formelle permet la définition exacte des comportements de programmes, la traduction vers des formats adaptés à la synthèse de logiciel ou de matériel, l'optimisation de cette synthèse et la vérification de propriétés de programmes. On désigne comme **réactifs** les systèmes dont la caractéristique principale est d'interagir avec leur environnement extérieur au rythme de cet environnement. Les systèmes réactifs **synchrones** s'appuient sur les notions d'horloge globale, de diffusion instantanée d'informations, de parallélisme déterministe et de préemption pour fournir un modèle de programmation cohérent et adapté. Esterel propose les primitives syntaxiques correspondantes, en complément d'un langage impératif traditionnel. Les opérations de manipulations de données sont reportées vers un langage hôte, par exemple C.

Les applications d'Esterel sont les contrôleurs temps-réel, les systèmes embarqués, les protocoles de communication, les interfaces homme-machine, les parties contrôle de circuits digitaux et, plus généralement, les systèmes réactifs dominés par le contrôle.

Le comportement d'un programme Esterel est défini par une sémantique mathématique formelle [Ber99a]. Un programme peut être compilé en un système d'équations booléennes avec mémoires, c'est-à-dire en un circuit synchrone. Cette traduction permet la synthèse directe de circuits électroniques ou la synthèse de programmes par tri des équations et traduction directe en C. Elle permet également d'établir une interface avec de nombreux systèmes de vérification formelle comme ceux développés dans le projet.

Le langage Esterel est désormais commercialisé par la société Esterel Technologies, avec laquelle nous entretenons des relations suivies de coopération.

Les recherches nouvelles autour du langage et de son modèle comportemental concernent :

- la caractérisation de méthodes efficaces d'implantation logicielle, pour se limiter à l'exécution de parties réellement actives dans une réaction ;
- l'étude de l'optimisation des circuits et programmes engendrés, et en particulier de leur optimisation modulaire ;
- le lien entre les langages synchrones dominés par le contrôle comme Esterel et les langages synchrones dominés par les données comme Lustre et Signal ;
- le lien entre les langages synchrones et les nouvelles méthodes de synthèse de systèmes mixtes matériel / logiciel ;
- la redécouverte de structure dans le cadre de la réingénierie de programmes à partir de systèmes d'équations booléennes « plats ».

Ces recherches sont de plus en plus fondées sur la définition d'une approche d'analyse par sémantique statique dédiée à la programmation synchrone.

Les développements théoriques conduisent à des algorithmes implantés dans des prototypes, pouvant aboutir à des extensions du compilateur Esterel v5. Des retours d'utilisateurs industriels ou universitaires viennent fréquemment susciter de nouvelles questions théoriques et pratiques concernant les méthodologies de conception et leurs besoins algorithmiques.

3.2. Méthodes d'analyse et de vérification automatique de systèmes réactifs synchrones

Participants : Yannis Bres, Robert de Simone, Fabrice Peix, Annie Ressouche, Xavier Thirioux, Éric Vecchié.

Mots clés : *verification automatique, fiabilité, diagrammes de décision binaires, model-checking, partitionnement, abstraction.*

Les modèles synchrones de systèmes réactifs réclament et permettent à la fois des méthodes puissantes d'analyse et de vérification automatique de propriétés fonctionnelles. Ces méthodes sont basées sur une exploration exhaustive des configurations dynamiquement atteignables du système. Elles utilisent des techniques de représentation symbolique comme les BDD (diagrammes de décision binaires). L'efficacité de certains algorithmes est cruciale pour le passage à l'échelle de ces méthodes sur des exemples de taille industrielle. Les techniques de « model-checking » symbolique sont désormais bien établies dans le domaine de la vérification automatique de systèmes réactifs synchrones. Néanmoins, les problèmes de performance dus à des phénomènes d'explosion combinatoire nous conduisent à rechercher de nouvelles techniques et à améliorer les techniques existantes pour réussir à pratiquer l'analyse de systèmes de taille toujours plus grande.

En général nos méthodes prennent appui sur la structuration hiérarchique de la modélisation (par exemple en Esterel) pour raffiner les calculs d'espace d'états. Nous étudions des partitionnement de la relation de transition, des méthodes de réduction compositionnelle des modèles, des approximations conservatives, des abstractions de données et des représentations symboliques de contextes séquentiels, pour former une panoplie d'outils permettant d'attaquer de diverses façons ce problème de complexité de représentation.

4. Domaines d'application

4.1. Télécommunications

Mots clés : *télécommunications, protocoles, mobilité.*

C'est naturellement un domaine important pour les systèmes réactifs et la communication par messages. Nos contributions y portent à la fois sur la spécification et sur la programmation fiable à l'aide de formalismes adaptés de haut niveau.

4.1.1. Programmation réactive synchrones de protocoles

Les téléphones ou postes de radio portables du futur vont devenir de véritables systèmes multimédia capables d'allier son, images, navigation Internet, consultation de base de données et relais de communications simultanées sur différentes gammes d'ondes. Ils auront leurs protocoles dédiés, avec normalisations UMTS, et des algorithmes de cryptage et d'évasion de fréquence. Les protocoles devront être téléchargeables. Nous collaborons avec Thomson sur l'utilisation d'Esterel pour programmer des piles protocolaires radio. Le projet PLANETE de l'INRIA conduit aussi des expériences sur l'utilisation d'Esterel pour des protocoles de nature similaire.

4.1.2. Spécification de microcontrôleurs dédiés

Le développement de microprocesseurs dédiés aux systèmes mobiles du futur (téléphones cellulaires notamment) passe par la spécification complète de haut niveau de tels systèmes, dans des buts de tests intensifs et de validation avant la mise en production effective des circuits. Nous collaborons avec la société Texas Instruments sur ces thèmes, afin de concevoir un modèle logiciel fiable (dénommé « Golden Model » dans leur processus de conception) qui puisse être utilisé en aval par les industriels développeurs d'application pour valider leurs concepts avant même la fourniture effective des microprocesseurs.

4.2. Systèmes embarqués en avionique/automobile

Mots clés : *systèmes embarqués, transports, protocoles, programmation réactive, contrôleur, avionique, automobile.*

Les systèmes embarqués étant souvent critiques pour leur fiabilité, il est essentiel de les développer et de les valider avec des méthodes formelles de programmation et de vérification. Par leur parallélisme inhérent de programmation, les langages synchrones permettent de s'affranchir de la gestion dynamique de tâches parallèles telle qu'on la trouve dans les systèmes classiques comme les OS temps-réel et qui est difficile à maîtriser. Le déterminisme des programmes parallèles Esterel permet une mise au point et une vérification beaucoup plus simple. Nous collaborons activement avec Dassault Aviation sur ces thèmes, ce qui a constitué le moteur principal des améliorations du langage Esterel par retour d'expériences au cours des dix dernières années.

4.3. Synthèse de circuits

Mots clés : *circuit, contrôleur, matériel.*

Les circuits matériels deviennent de plus en plus complexes, surtout en ce qui concerne le contrôle des chemins de données (pipeline, cohérence de caches, interfaces bus, etc.). Esterel est adapté à la description et à la synthèse efficace de contrôleurs matériels. Nous travaillons sur ce thème avec le laboratoire « Intel Strategic CAD lab » de Portland, US. Nous avons acquis, pour valider nos résultats de synthèse, la plate-forme de CAO électronique Synopsys par le biais de l'opération européenne **EuroPractice**.

4.4. Conception conjointe matériel / logiciel

Mots clés : *ingénierie, circuits digitaux.*

Les systèmes embarqués sont souvent faits de composants mixtes matériel / logiciel dont la conception doit être conjointe. Les langages synchrones sont bien adaptés à ce problème, car ils peuvent être compilés indifféremment sur des cibles matérielles ou logicielles. Nous travaillons dans ce domaine avec la société Cadence Design Systems. Cette société développe des langages fortement inspirés d'Esterel, mais à la syntaxe influencée par les langages C (comme ECL, « Esterel C Language ») ou Java (comme Jester, « Java-Esterel »). Nous étudions la possibilité d'intégrer ces formalismes dans le produit VCC (Virtual Components Compiler).

5. Logiciels

5.1. Esterel

Participants : Yannis Bres [correspondant], Dumitru Potop.

Mots clés : *Esterel, compilateur, synthèse, optimisation, causalité.*

Le compilateur Esterel v5 traduit les programmes Esterel vers C ou des formats de description de circuits digitaux. Le compilateur Esterel v5 consiste en plusieurs processeurs permettant de produire des codes objets pour des cibles matérielles ou logicielles. L'environnement de programmation contient également un simulateur-débogueur graphique XES permettant de mettre au point les programmes, des optimiseurs spécialisés fondés sur des techniques de calcul booléen, et des interfaces vers le système de vérification automatique de propriétés XEVE.

Le compilateur a été conçu de façon préindustrielle pour offrir de bonnes performances et une grande robustesse. Il est diffusé sur le Web en version binaire d'évaluation à l'URL http://www.esterel.org/. Les sociétés Dassault Aviation et Synopsys en ont acquis des licences sources. Il est désormais industrialisé et diffusé par la société Esterel Technologies, qui développe également autour de cette base un formalisme de représentation graphique, issu à l'origine des travaux de Charles André de l'I3S, et nommé SyncCharts.

A la suite de Gérard Berry plusieurs membres de notre équipe ont rejoint la société Esterel Tcehnologies, qui assure désormais l'essentiel des évolutions du compilateur industriel. Nous réalisons des contributions prototypes à partir de la souche universitaire pour valider nos travaux de recherche avancée.

5.2. Xeve

Participants : Amar Bouali [Esterel Technologies], Robert de Simone, Xavier Thirioux.

Mots clés : *vérification automatique, BDD, abstraction, minimisation, partitionnement, réduction, équivalence comportementale, bisimulation, observateur, sûreté, vivacité, logiciel fiable, interface graphique.*

Ce logiciel traduit en algorithmes nos travaux sur le « model-checking » efficace de systèmes réactifs synchrones. Les spécifications réactives synchrones modélisent fréquemment des systèmes embarqués, pour lesquels la fiabilité fonctionnelle est un élément critique. Les méthodes formelles, et en particulier les activités de vérification automatique connues sous l'appellation de « model-checking », ont donc rencontré un essor important dans les années récentes. Les problèmes pratiques posés sont ceux du passage à l'échelle des algorithmes d'analyse et de construction de la représentation de l'espace d'états atteignables, dans des applications de taille industrielle.

Le logiciel Xeve contient l'essentiel de nos contributions dans ce domaine. Il s'appuie sur la bibliothèque TiGeR, pour la manipulation de représentations symbolique à base de Diagrammes de Décision Binaires (BDD). Xeve calcule des propriétés de sûreté et d'équité comportementales, ainsi que des quotients par bisimulation des espaces d'états atteignables. Il optimise ce calcul d'états de nombreuses façons, et nous poursuivons constamment des recherches sur de nouvelles techniques algorithmiques pour rendre plus efficace ce calcul, central dans l'analyse et la vérification automatique des systèmes réactifs synchrones.

Cette année nous avons expérimenté des méthodes simples et intuitives pour l'expression immédiate par un non-expert de propriétés de correction souhaitées. Les propriétés sont évaluées par des algorithmes dédiés, plus

efficaces que les méthodes générales d'interprétation par observateurs. Ce travail est effectué dans un nouveau logiciel νEVE, développé par Xavier Thirioux, ingénieur-expert dans le cadre du projet RNRT Syntel.

6. Résultats nouveaux

6.1. Exécution « efficace » logicielle de programmes Esterel

Participants : Dumitru Potop, Robert de Simone.

Mots clés : *Esterel, code efficace*.

La traduction d'Esterel en circuits mène à un schéma simple d'exécution des programmes, qui consiste à évaluer séquentiellement dans un ordre donné toutes les équations définissant les portes logiques et les regitres booléens du circuit, et ce lors de chaque réaction du programme. Néanmoins, dans un programme hiérarchique de larges parties peuvent se révèler structurellement inactives, ce qui est difficile à détecter après traduction en circuit. Le sujet de thèse de Dumitru Potop consiste dans l'utilisation de la structure du programme Esterel pour définir un mode de compilation produisant un code plus efficace car n'exécutant que certaines parties actives du système d'équations. Des travaux similaires ont été conduits dans la société Synopsys et au CNET Grenoble.

Le modèle intermédiaire de représentation de programmes défini actuellement devra également permettre d'effectuer des optimisations importantes. Il s'agit principalement d'identifier des équivalences synchrones entre points de contrôle parallèles, par des méthodes d'analyse statique appropriées.

6.2. Traduction d'Esterel vers SynDex

Participants : Fabrice Peix, Annie Ressouche, Robert de Simone.

Le logiciel SynDex, développé par le projet INRIA SOSSO à Rocquencourt, permet à l'origine de répartir un programme de type synchrone flôt de données sur une architecture matérielle, en optimisant les allocations de calcul. L'objectif est ici d'étudier l'implantation efficace de programmes Esterel dans ce cadre, ce qui pose de nouvelles questions du fait de l'aspect flôt de contrôle bien plus important en Esterel que dans les autres langages synchrones comme Signal et Lustre.

Une première traduction, par l'intermédiaire d'une transformation du flot de contrôle en flôt de données basée sur la traduction en circuits, a révélé les problèmes d'inefficacité de cette approche du fait de la trop faible granularité des calculs et de l'explosion du nombre de variables.

Nous étudions actuellement, dans le cadre de la thèse de Fabrice Peix, des méthodes pour intégrer des dépendances entre données dans le flôt de contrôle global, et ce à sémantique constante. Ce travail est mené dans la suite des études visant à connecter Esterel et SynDex, mais il pourrait avoir d'autres conséquences sur l'étude des relations enre formalismes synchrones.

Ce travail a donné lieu à plusieurs rencontres avec l'équipe SynDex.

6.3. Partitionnement structurel pour le calcul symbolique d'états atteignables

Participants : Robert de Simone, Eric Vecchié.

L'efficacité des méthodes de model-checking symbolique repose grandement sur le partitionnement des fonctions de transition du programme réactif. Dans le stage de DEA de Eric Vecchié, nous sommes intéressés à l'utilisation de partitionnements séquentiels (en fonction de l'enchaînement de modes dans le programme). Cette approche est maintenant généralisée pour considérer des sous-programmes partiellement indépendants car réagissant à des évènements disjoints, et donc pouvant évoluer de manière asynchrone. L'objectif est de saturer d'abord des sous-ensembles d'états atteints localement, afin d'en rendre l'expression symbolique BDD plus régulière, et par là de simplifier l'expression des configurations intermédiares globales. Ce point est important car c'est souvent à ce niveau qu'on constate une explosion combinatoire de la représentation.

6.4. Spécification simple de propriétés de correction

Participants : Robert de Simone, Xavier Thirioux.

Dans le cadre de notre participation au projet RNRT Syntel nous avons défini et réalisé une interface utilisateur permettant une meilleure intégration entre les logiciels Xes et Xeve, afin de permettre l'expression facile et naturelle de propriétés de correction de programmes au niveau du code source Esterel, pour une application algorithmique au niveau du code objet « circuit ». Nous travaillons actuellement à étendre cette interface pour permettre l'expression de propriétés plus variées, en fondant de nouvelles techniques algorithmiques optimisées sur cette forme d'expression des propriétés. L'objectif général reste toujours d'être capable d'optimiser le calcul de l'espace de configurations atteignables du système, avec ici des indications de l'utilisateur pour mieux partitionner et diviser ce calcul.

6.5. Verification par analyse de causalité

Participants : Charles André, Robert de Simone.

La vérification de propriétés temporelles des systèmes s'attache traditionnellement aux propriétés de correction entre instants. Du fait de la richesse des comportements lors d'une réaction unique en Esterel, les propriétés « dans linstant » sont également un sujet d'interêt pour établir la correction d'un programme. Nous avons étudié cet aspect en nous appuyant sur un exemple de file FIFO qui autorise le dêpôt et le retrait simultané de données. Ceci pose des problèmes causaux aux bornes (FIFO pleine ou vide). Une modélisation fine est nécessaire, et pour sa vérification nous introduisons des observateurs dans l'instant. Ces travaux ont fait l'objet d'une communication à MSR'01 [AdS01].

6.6. Esterel multiphases

Participant : Loïc Henry-Gréard.

Mots clés : *Esterel*, *HDL*.

Malgré sa proximité naturelle avec les circuits digitaux synchrones, Esterel manque parfois de constructions syntaxiques familières aux ingénieurs du domaine. Des extensions ont été proposées récemment par Gérard Berry et Mike Kishinevsky de la société Intel. Cette année Loïc Henry-Gréard a étudié dans le cadre de sa thèse une extension d'expressivité du langage autorisant la division de la réaction en plusieurs phases distinctes, et les techniques de synthèse matérielle correspondantes à base de « transparent latches ». Un exemple particulier est celui de programme pouvant réagir aux deux fronts, montants et descendants, de l'horloge de base.

Loïc Henry-Gréard a été récemment embauché par la société Intel à Portland.

6.7. Traduction vers des formalismes de description de circuits ou d'architectures de systèmes embarqués

Participants : Laurence Pierre, Annie Ressouche, Robert de Simone.

Nous avions précédemment réalisé un traducteur VHDL pour Esterel au niveau des sytèmes d'équations booléennes (les schémas de portes logiques des circuits synchrones). Nous envisageons maintenant d'étudier des traductions plus sophistiquées, respectant des caractéristiques de structuration modulaire et produisant du code VHDL de plus haut niveau comportemental. Ce travail doit se poursuivre en collaboration avec Laurence Pierre dans le cadre de sa délégation auprès de notre équipe.

6.8. Abstraction de registres dans l'analyse de programmes

Participant : Yannis Bres.

Mots clés : *Esterel*, *code efficace*.

La complexité du calcul symbolique de l'espace d'états atteignable est indirectement liée au nombre de registres composant les états locaux de cette structure. On peut simplifier ce calcul au prix d'une approximation

conservative en « oubliant » certains registres. Une technique originale et prometteuse d'interprétation abstraite a été définie dans ce sens. Elle est basée sur une interprétation trivaluée de certaines fonctions de transition. Son implantation et son évaluation en performance algorithmique sont en cours.

6.9. Conception « synchrone » d'un système de téléphonie par Internet

Participants : Fabrice Giocanti, Ludovic Segarra, Robert de Simone, Férial Virolleau.

Dans le cadre du stage ESIEE de Ludovic Segarra et Fabrice Giocanti nous avons entrepris de modéliser des parties du logiciel FreePhone, initialement développé dans le projet RODEO, en utilisant les méthodes synchrones de spécifications et de programmation.

Ce travail a été mené dans le cadre de l'Action Color MÛRE, et a eu pour effet d'initier une coopération avec l'ESIEE sur ce thème de la conception méthodologique d'applications de traitement du signal, qui devrait se poursuivre dans la perspective de la création d'une antenne de cette École à Sophia-Antipolis.

6.10. Synthèse de programmes structurés

Participants : Robert de Simone, Olivier Tardieu.

Les constructions syntaxiques du langage Esterel ont chacune une interprétation bien identifiée comme transformation de circuits. On peut s'interroger sur la possibilité de resynthétiser à partir d'un circuit non structurel un programme Esterel, en s'inspirant de l'expérience des formes produites par cette traduction. Il s'agit d'un travail de « reverse engineering », dont l'impact peut être potentiellement important s'il permet de restructurer tout circuit en un programme. Ceci devrait faire le sujet de la thèse d'Olivier Tardieu.

6.11. Vérification de moteurs pour les systèmes à base de connaissances

Participants : Annie Ressouche, Sabine Moisan [projet ORION], Jean-Paul Rigault [professeur UNSA-I3S].

Cette année nous avons étudié des modèles synchrones pour modéliser le comportement des moteurs et nous avons mené des expériences avec des outils existants. Le comportement des composants de BLOCKS a été tout d'abord spécifié en utilisant le langage Esterel (et l'outil graphique EsterelStudio commercialisé par Esterel Technologies). Cet outil comporte un simulateur qui nous a permis de visualiser le comportement des objets modélisés ainsi qu'un outil de preuves qui nous a permis de vérifier certaines propriétés relatives au comportement des méthodes des composants. Cette expérience a montré que les modèles synchrones sont bien adaptés pour décrire les comportements des composants d'un moteur, mais l'utilisation directe des outils existants s'avère difficile car l'application des techniques de vérification du domaine synchrone aux systèmes à base de connaissances est une approche nouvelle qui demande une adaptation tant au niveau de la définition du modèle qu'au niveau de la conception des outils.

Nous avons donc conçu un modèle synchrone mieux adapté à notre problématique et sur lequel les techniques de model checking sont valides. Pour ce faire, nous avons défini un langage de spécification comportemental pour décrire le fonctionnement des moteurs. Ce langage est inspiré du langage graphique Argos et il est bien adapté à la modélisation du comportement des composants de BLOCKS. Il manipule essentiellement des automates et il est muni d'un opérateur de parallélisme et d'une opération de hiérarchie qui permet de raffiner les états des automates. Il diffère d'Esterel dans la mesure où les opérateurs sont plus simples et n'entrainent pas de problème de causalité. Ensuite, nous avons donné une sémantique aux comportements décrits par ce langage en terme de machine d'états finis. Ce modèle est bien adapté à la vérification de propriétés de la logique temporelle et aux techniques de *model checking* qui lui sont associées. En particulier, notre langage supporte des techniques de vérification modulaires (par rapport à ses opérateurs). Quant à la preuve de comportement des moteurs ainsi construits, on peut envisager soit d'introduire les techniques de model checking directement dans l'environnement de développement, soit d'interfacer des outils existants.

Il reste encore à envisager la conception pratique d'outils réalistes. Dans un premier temps, nous pensons introduire notre langage dans Ptolemy II. C'est un système hétérogène et ouvert pour modéliser et simuler des systèmes qui offre la possibilité de définir son propre domaine de calcul. Ceci devrait nous permettre

d'avoir un simulateur du langage et d'interfacer les outils de model checking existants. A plus long terme, nous envisageons de réaliser un outil dédié. De plus, le domaine de la conception de systèmes de base de connaissances a de multiples facettes (comme la distribution des moteurs, l'utilisation des outils générés à travers le réseau,etc.) qui posent des problèmes nouveaux en matière de vérification.

7. Contrats industriels

7.1. Programmation synchrone pour les télécommunicatons

Participants : Robert de Simone, Xavier Thirioux.

Ce contrat RNRT a pour coordinateur la société Esterel Technologies, et rassemble comme partenaires, outre notre projet, les sociétés Thomson CSF et Cadence (US). Le but de cette action est d'étudier l'introduction des méthodes synchrones de modélisation dans VCC, un environnement général de simulation conjointe matériel/logiciel pour la conception de systèmes embarqués télécom commercialisé par la société Cadence et utilisé par Thomson CSF. Notre rôle a consisté en l'ajustement des fonctionnalités de nos outils dans ce contexte. Ce contrat a financé le poste d'ingénieur-expert de Xavier Thirioux. Il se termine à la fin de l'année 2001.

8. Actions régionales, nationales et internationales

8.1. Actions nationales

8.1.1. Action de recherche coopérative S-JAVA

Participant : Robert de Simone.

Notre participation actuelle dans cette action est minime. Il s'agit principalement de transférer nos compétences en model-checking symbolique, incarnées dans les logiciel de vérification FC2TOOLS, pour les voir appliquer dans le contexte des protocoles de sécurité et de la fiabilité de code embarqué (JAVA Card notamment). Ces compétences sont reprises par Eric Madelaine, un ancien membre de notre équipe maintenant chercheur au sein du projet OASIS.

8.1.2. Action Color-2001 Mûre

Participants : Fabrice Giocanti, Jérôme Millon, Ludovic Segarra, Robert de Simone, Férial Virolleau.

Cette action de coopération locale, commune avec les projets I3S Sports et Mozarts, vise à modéliser par des techniques synchrones des applications réactives issues de projets de recherche locaux. Elle a donné lieu à deux séries de stage étudiants.

Fabrice Giocanti et Ludovic Segarra, tous deux étudiants de l'ESIEE, ont réimplémenté les fonctionnalités de traitement du signal de FreePhone. Il s'agit d'un logiciel de transmision de voix sur IP, à l'origine développé dans le projet INRIA RODEO. Esterel a été ici utilisé pour le pilotage d'algorithmes, afin d'autoriser des codages et décodages différents du flux d'information en fonction de la charge du réseau.

Jérôme Millon, stagiaire EMERIR Perpignan, a modélisé une application de suivi de cible destinée au robot ANIS du projet ICARE. L'accent a été ici mis sur les aspects méthodologiques orientés-objet (UML) de la conceptioon d'application. La réalisation, initialement prévue sur la plate-forme distribuée du CyCab, a été différée du fait de l'indisponibilité de ce véhicule. Ce travail devrait pouvoir se continuer dans le cadre du DEA RSD UNSA de Jérôme Millon.

9. Diffusion des résultats

9.1. Animation de la Communauté scientifique

Nous avons créé le site web www.esterel.org pour rassembler la communauté d'utilisateurs du langage et promouvoir nos techniques et nos logiciels.

Robert de Simone est membre élu de la Commission d'Évaluation de l'Institut, et a participé à ce titre à plusieurs jurys de recrutement. Il a présidé pour l'année 2001 la Section d'Audition du recrutement chercheur de Sophia-Antipolis. Il est également membre de la Commission de Spécialistes en 27e section de l'université de Nice/Sophia-Antipolis. Enfin il a aussi fait partie du comité de programme du colloque MSR'2001.

Plusieurs membres de l'équipe ont participé à la conférence CAV'01, à Paris.

L'ensemble de notre équipe va participer au séminaire Synchron'2001, à Dagstuhl (Allemagne). Ces journées rassemblent pour des rencontres informelles les chercheurs du domaine « réactif synchrone », dont les trois équipes développant les formalismes Esterel, Lustre et Signal, et des spécialistes de conception orientée-objet synchrone.

9.2. Enseignement

Robert de Simone coordonne le cours « Méthodes formelles et fiabilité du logiciel » du DEA Informatique de l'université de Nice/Sophia-Antipolis, et il y enseigne (15h); il enseigne également à l'ISIA une semaine de cours sur les méthodes formelles et leurs applications (18h).

Dans le cadre du monitorat, Yannis Bres enseigne en DEUG Informatique à l'UNSA : TP d'algorithmique et programmation, cours et TP sur le Java Development Kit (en tout 92h annuelles).

Dans le cadre du monitorat, Fabrice Peix enseigne en DEUG Informatique à l'UNSA : TP Unix et systèmes informatiques (en tout 92h annuelles).

Dans le cadre du monitorat, Eric Vecchié enseigne en DEUG Informatique à l'UNSA (en tout 92h annuelles).

Annie Ressouche assure à l'ISIA les TP sur les outils d'analyse syntaxique (15h).

Dans le cadre d'une coopération STIC avec la Tunisie (ENSI de Tunis), Annie Ressouche a encadré avec Sabine Moisan un stage (Anissa Omrane) qui a permis un début de réalisation du serveur du projet Orion. Ce serveur dédié au pilotage donne accès à des bases de connaissances en pilotage de programmes afin de permettre l'accès collaboratif à des codes distants et le partage de codes entre équipes.

10. Bibliographie

Bibliographie de référence

[BdS91] F. BOUSSINOT, R. DESIMONE. *The Esterel Language.* in « Another Look at Real Time Programming, Proceedings of the IEEE », volume 79, 1991, pages 1293–1304.

[Ber99a] G. BERRY. *The Constructive Semantics of Pure Esterel.* version electronique, 1999, ftp://ftp.esterel.org/esterel/pub/papers/constructiveness3.ps.gz

[Ber99b] G. BERRY. *The Esterel Language Primer.* version electronique, 1999, ftp://ftp.esterel.org/esterel/pub/papers/primer.pdf

[Ber00] G. BERRY. *The Foundations of Esterel.* série Foundations of Computing Series, MIT Press, 2000, ftp://ftp.esterel.org/esterel/pub/papers/foundations.pdf

Articles et chapitres de livre

[BCE+] A. BENVENISTE, P. CASPI, S. EDWARDS, N. HALBWACHS, P. L. GUERNIC, R. DESIMONE. *Synchronous Languages Twelve Years Later.* in « Proceedings of the IEEE », note : à paraître.

[MRR02] S. MOISAN, A. RESSOUCHE, J.-P. RIGAULT. BLOCKS, *a Component Framework with Checking Facilities for Knowledge-Based Sys tems.* in « Informatica, Special Issue on Component Based Software Development », 2002, note : to appear.

Communications à des congrès, colloques, etc.

[AdS01] C. ANDRÉ, R. DESIMONE. *Programmation synchrone: Propriétés dans une réaction.* in « Actes de la conférence MSR'01 : Modélisation des Systèmes Réactifs », 2001, http://dmi.ensica.fr/msr2001/