ÉLÉMENT DE PORTFOLIO 05

Publication

1 DÉFINITION DE CET ÉLÉMENT

Titre de l'élément : Parameterized Synthesis for Fragments of First-Order Logic over Data Words. Publié dans les actes de FOSSACS'20.

URL de l'élément : `https://arxiv.org/pdf/1910.14294.pdf`

2 MOTIVATIONS DU CHOIX DE CET ÉLÉMENT

Cet article a été publié dans les actes de la conférence internationale FOSSACS'20, conférence de rang A qui rassemble des travaux de recherche fondamentale ayant des applications claires pour la science du logiciel. Il rassemble une partie du travail fait pendant la thèse de Mathieu Lehaut, doctorant de l'équipe co-dirigé par Béatrice Bérard et Nathalie Sznajder de l'équipe MoVe d'une part, et Benedikt Bollig du LMF d'autre part. Il illustre les travaux de l'équipe dans les aspects théoriques de la vérification.

3 PRÉSENTATION DE CET ÉLÉMENT

3.1 Contexte

Cet article traite du problème de synthèse, tel que défini par Alonzo Church en 1963 : il consiste à dériver automatiquement un système (sous la forme d'un modèle formel) à partir de sa spécification. Ainsi, si la spécification est irréalisable, on peut le savoir avant tout travail de réalisation et, dans le cas contraire, on obtient automatiquement un modèle correct par construction du système à réaliser. Il s'agit dans ce problème de systèmes réactifs ouverts, c'est-à-dire des systèmes qui doivent maintenir un comportement correct en interaction avec un environnement incontrôlable. Le problème est formulé de la façon suivante : étant donné un alphabet d'actions pour le système et un alphabet d'actions pour l'environnement, chacun des acteurs choisit une action alternativement, ce qui produit un mot infini. Etant donné une spécification sur les mots infinis, est-il possible pour le système d'avoir une stratégie lui permettant de toujours produire un mot autorisé par la spécification, quel que soit le comportement du système ? Dans le cas de systèmes simples, le problème est à présent bien compris et des outils arrivent à maturité. Lorsque le système à synthétiser doit être composé de plusieurs processus, ayant chacun une vision locale, le problème est plus difficile (et en général indécidable). Dans cet article nous nous sommes intéressés à la synthèse de systèmes distribués dans lesquels le nombre de processus est variable : les systèmes paramétrés. Ce type de systèmes est très répandu, que ce soit dans l'algorithmique distribuée, les réseaux ad-hoc, les systèmes biologiques... La question devient donc : existe-t-il un nombre de processus pour lesquels le système peut s'assurer de toujours satisfaire la spécification, quelles que soient les actions choisies par l'environnement ? On peut également considérer une version "universelle" : le système peut-il toujours garantir de satisfaire la spécification, quelles que soient les actions choisies par l'environnement, et quel que soit le nombre de processus ?

3.2 Résultats

Dans le cas des systèmes infinis, il n'est plus possible de décrire les exécutions de tels systèmes par des mots sur un alphabet *fini* : en effet, chaque action doit être localisée sur le processus qui l'a émise, ce qui revient à étiqueter chaque action par l'identifiant du processus sur lequel elle s'exécute. Dans le cas où le nombre de processus n'est pas connu à l'avance, cela revient à considérer un alphabet *infini*.

Décrire une spécification revient donc à définir un langage sur un alphabet infini. Plusieurs formalismes ont été proposés pour cela ces dernières années, que ce soit par des modèles d'automates ou des logiques. S'il n'existe pas (encore) de notion de langage régulier sur les alphabets infinis, la logique du premier ordre (FO) sur les mots avec données est largement acceptée comme standard pour décrire de tels modèles : les mots sont constitués de lettres ayant une composante finie, et une composante prise dans un domaine infini (les données), qui dans

notre cas, représente l'identifiant d'un processus. Son fragment FO^2, dans lequel au plus deux variables peuvent apparaître dans les formules, est décidable pour la satisfaisabilité et l'universalité.

Nous étudions donc dans cet article le problème de synthèse pour les systèmes paramétrés de la façon suivante : étant donnée une formule logique donnée dans la logique du premier ordre sur les mots avec données, existe-t-il un nombre de processus permettant au système de garantir une exécution correcte, quelle que soit la manière dont l'environnement se comporte ?

Notre modélisation comporte encore deux spécificités : dans le cas d'un système distribué paramétré, donc mettant potentiellement en jeu un grand nombre de processus, il n'est pas raisonnable d'envisager une communication synchrone entre le système et son environnement, avec une alternance d'actions entre les deux. Nous considérons donc le problème dans un cadre asynchrone, dans lequel le système n'a pas de contrôle sur le moment où les actions de l'environnement vont avoir lieu. Enfin, nous considérons que le système est composé de trois types de processus : les processus "système" qui n'ont aucune interaction avec l'environnement, les processus "environnement", qui ne comportent aucune action du système, et les processus "mixtes", sur lesquels ont lieu à la fois des actions du système et des actions de l'environnement.

Nos résultats sont les suivants :

- le problème de synthèse que nous considérons est indécidable pour FO^2,
- il est également indécidable pour le fragment de FO dans lequel on s'interdit d'imposer un ordre aux actions
- il devient décidable si l'on fixe le nombre de processus contrôlés par l'environnement, en laissant libre le nombre de processus "système".

3.3 Impact

La logique sur les mots avec données, en lien avec la théorie des automates, et les problèmes de vérification et de synthèse est intensivement étudiée ces dernières années, pour ses applications à la fois dans les systèmes paramétrés et dans les bases de données. Ce travail apporte donc un éclairage supplémentaire important dans ce domaine.