Avis de Soutenance

Monsieur Dara LY

Informatique

Soutiendra publiquement ses travaux de thèse intitulés

Formalisation d'un vérificateur dynamique de propriétés mémoire pour programmes C

dirigés par Monsieur JEAN-MICHEL COUVREUR et Monsieur Frederic LOULERGUE
Ecole doctorale : Mathématiques, Informatique, Physique Théorique et Ingénierie des Systèmes - MIPTIS
Unité de recherche : LIFO - Laboratoire d'Informatique Fondamentale d'Orléans

Soutenance prévue le ***lundi 05 décembre 2022*** à 14h00
Lieu : 1 rue Raimond Castaing et rue René Thom - 91190 GIF SUR YVETTE
Salle : LISN - Bâtiment DIGITEO

Composition du jury proposé

M. Frédéric LOULERGUE	Université d'Orléans	Co-directeur de thèse
M. Julien SIGNOLES	CEA LIST	Co-encadrant de thèse
M. Nikolai KOSMATOV	Thales Research and Technology	Co-encadrant de thèse
M. Jean-Christophe FILLIATRE	CNRS	Examinateur
M. François PESSAUX	ENSTA Paris	Examinateur
M. Alan SCHMITT	INRIA	Rapporteur
M. Claude MARCHE	INRIA	Rapporteur
M. Mickaël DELAHAYE		Invité

Mots-clés : propriétés mémoire, vérification dynamique, preuve formelle,,

Résumé :

La vérification d'assertions à l'exécution est une technique permettant de contrôler, lors de l'exécution d'un programme, la conformité de ce programme vis-à-vis d'une spécification donnée sous forme d'annotations formelles : les assertions. Un procédé appelé instrumentation transforme les assertions en code exécutable, de manière à mettre en œuvre un moniteur en ligne pour le programme à vérifier. Le long de l'exécution du programme instrumenté, le moniteur contrôle la conformité du programme vis-à-vis des assertions, et, en cas de non-respect d'une assertion, met fin à l'exécution. Autrement, il laisse le comportement fonctionnel du programme inchangé. La complexité de mise en œuvre de l'instrumentation dépend largement des propriétés exprimables dans le langage d'annotation. Dans le cas de programmes en langage C, l'outil E-ACSL (greffon de Frama-C, une plateforme open-source d'analyse de code C), permet la vérification de propriétés relatives à l'état mémoire du programme, mais requiert pour cela une instrumentation complexe. La présente thèse est consacrée à la formalisation de cette instrumentation : il s'agit d'en donner une définition précise et d'étudier ses propriétés sémantiques. Nous proposons une modélisation du problème comme une traduction de programmes depuis un langage source, muni d'assertions logiques, vers un langage cible. Ce dernier est dépourvu d'assertions logiques, mais intègre une structure de données, appelée mémoire d'observation, dédiée au suivi des propriétés mémoire. Nous donnons une caractérisation axiomatique de la mémoire d'observation, et utilisons celle-ci pour définir la sémantique du langage cible de la transformation, dont nous montrons qu'elle est correcte vis-à-vis de la sémantique des langages concernés. Additionnellement, nous étudions l'optimisation de l'instrumentation par analyse de flot de données, technique mise en œuvre dans l'outil E-ACSL afin de réduire le surcoût en performance induit par l'instrumentation. L'analyse vise à déterminer un sous-ensemble d'emplacements mémoire minimal dont l'instrumentation permette au moniteur d'évaluer correctement les assertions du programme. Nous définissons une telle analyse, et prouvons qu'elle est sûre, au sens où limiter l'instrumentation aux seuls emplacements désignés par l'analyse ne compromet pas la validité des verdicts du moniteur.