Algorithmes est structures de données pour la vérification formelle

Plan général du cours 2016

Gérard Berry

Collège de France

Chaire Algorithmes, machines et langages

gerard.berry@college-de-france.fr

COLLÈGE DE FRANCE
1530

Rappel : premier cours à Rennes !

Inria Rennes, 4 novembre 2015

Cours 1 :

L'importance des langages en informatique

Séminaire par Thomas Jensen (Inria Rennes) :

Intégration de la vérification formelle dans les langages de programmation

Réflexion sur la difficulté intrinsèque de la création et de la réalisation de langages pour la conception, la spécification, la réalisation et la vérification des applications informatiques

Vérification automatique : quelle est la question ?

- Dans beaucoup de domaines, il faut résoudre de nombreux problèmes combinatoires difficiles
  - conception assistée de circuits électroniques
  - optimisation et vérification de programmes ou de circuits
  - équivalence de programmes ou de circuits
  - programmation par contraintes, optimisation générale, jeux, etc.
- Ces problèmes prennent souvent la forme de formules mathématiques, avec deux grandes questions :
  - satisfiabilité : la formule peut-elle être rendue vraie en choisissant bien les valeurs des variables ?
  - validité : la formule est-elle vraie pour toutes les valeurs des variables?

L'expression des problèmes repose sur des théories variées:
logique, arithmétique, graphes, algèbre, etc.
qu'il faut aussi mélanger

Vérification automatique : quel est l'objectif ?

- Fournir des algorithmes automatiques pour résoudre les questions posées
  - algorithmes énumératifs : explorer explicitement l'espace pour trouver les solutions ou montrer leur absence (exhaustivement, aléatoirement, etc.)
  - algorithmes déductifs : s'appuyer sur la logique et les théories mathématiques décidables (ou semi-décidables), de complexité raisonnable, surtout dans les « cas pratiques ».
- Ne pas demander de compétence particulière à l'utilisateur
  - sinon celle de pouvoir exprimer son problème dans les formalismes fournis → ergonomie non triviale

Un objectif hélas intrinsèquement difficile : prévoir le coût !
Client : Quelle ressources dois-je allouer pour ce problème?
Fournisseur : Ahem... On ne le saura qu'après....

Vérification automatique : quels sont les moyens ?

- Une variété de théories décidables, avec des progrès majeurs
  - le calcul booléen, avec BDDs, ZDDs, et algorithmes SAT
  - la satisfaction modulo théories (SMT), mélange de logique booléenne et de théories décidables variées, rendues compatibles.
  - les automates temporisés, pour les problèmes temps-réels
  - les explorateurs explicites (SPIN, CADP, etc.), bien adaptés aux algorithmes disribués et aux protocoles de communication
- Des communautés très actives
  - améliorations constantes des algorithmes
  - constitutions de grandes librairies de benchmarks : difficiles, aléatoires, industriels,...
  - compétitions SAT, SMT, etc., majeures pour les progrès

Un merveilleux domaine de recherche,
où il faut avoir l'algorithme dans la peau !

Algorithmes est structures de données pour la vérification formelle

Collège de France, 9 mars 2016

Cours 2 :

Présentation générale du cours

Les BDDs (Binary Decision Diagrams)

Séminaire par Jean-Christophe Madre (Mentor Graphics) et Patrick Vuillod (Synopsys) :

Application des BDDs dans la conception de circuits intégrés

Suite du cours 6 du 1er avril 2015, *Vérification et optimisation booléennes d'automates et de circuits*

Les BDDs sont la première structure permettant le calcul booléen efficace, avec des applications dans de nombreux domaines

Algorithmes est structures de données pour la vérification formelle

Paris, 19 mars 2016

Cours 3 :

SAT : la satisfaction booléenne

Séminaire par Laurent Simon (Labri Bordeaux)

SAT : des victoires contre des problèmes difficiles

Les cours 2 et 3 concernent le calcul propositionnel (booléen) qui est au cœur de beaucoup de problèmes informatiques. Il a fait des progrès étonnants depuis environ 2000, permettant de traiter des centaines de milliers de variables sur des problèmes industriels chose absolument impensable au 20e siècle !

Algorithmes est structures de données pour la vérification formelle

Collège de France, 23 mars 2016

Cours 4 :

SMT : la satisfaction modulo théories

Séminaire par Sylvain Conchon (LRI Orsay)

SMT en pratique : le démonstrateur Alt-Ergo

SMT traite de formules booléennes donc les atomes ne sont plus des lettres mais des formules écrites dans d'autres théories, généralement décidables (mais pas toujours) : algèbres, arithmétique restreintes, inéquations linéaires, bitvecteurs, accès aux tableaux, gestion mémoire, etc. Les solveurs SMT savent combiner ces théories pour la vérification, la programmation par contraintes, etc.

Algorithmes est structures de données pour la vérification formelle

Collège de France, 30 mars 2016

Cours 5 :

La vérification formelle des programmes temporisés

Séminaire par Kim Larsen (LRI Orsay)

Real-Time Model Checking of Embedded Systems

Les transitions des programmes temporisés se déclenchent soit par l'arrivée d'événements, soit par le passage du temps. Sous certaines conditions restrictives, leur théorie et décidables Le système UPPAAL de Kim Larsen et. al. implémente les algorithmes correspondants. Il s'applique en particulier aux systèmes embarqués industriels.

Algorithmes est structures de données pour la vérification formelle

Collège de France, 6 avril 2016

Cours 6 :

La vérification par énumération explicite

Les systèmes précédents travaillaient de façon implicite, c'est à dire à l'aide de formules décrivant les objets et états. A l'inverse, la vérification explicite construit explicitement les états et transitions d'un système, soit exhaustivement, soit de façon aléatoire. Elle est très utilisée pour vérifier les protocoles de communication et les algorithmes distribués

Algorithmes est structures de données pour la vérification formelle

Collège de France, 6 avril 2016

Séminaire par Stéphanie Delaune (LSV, ENS Cachan) :

La vérification formelle appliquée aux protocoles cryptographiques

Les protocoles cryptographiques sont les programmes qui servent à établir une communication sécurisée.
Ils sont fragiles et constituent le principal point d'entrée pour les attaques de sécurité, comme le montreront quelques exemples. Ils sont aussi très difficiles à vérifier.

La preuve formelle est de plus en plus vue comme le meilleur moyen (voire le seul) d'assurer leur bon fonctionnement.
Elle demande la mise en œuvre de techniques très élaborées.

Algorithmes est structures de données pour la vérification formelle

Collège de France, 13 avril 2016

Cours 7 :

Réponses aux questions de l'année

Assistants et internautes, envoyez vos questions par courriel à gerard.berry@college-de-france.fr, même si vous étiez dans la salle.

Et, s'il vous plaît, n'attendez pas les derniers jours !

Algorithmes est structures de données pour la vérification formelle

Collège de France, 13 avril 2016

Séminaire par Chantal Keller (LRI Orsay)

Vers une automatisation sûre et expressive :
combiner preuves automatiques et interactives

Les algorithmes de vérification étudiés cette année ont l'avantage d'être automatiques et de demander peu d'effort à l'utilisateur. Mais leur portée est limitée.

A l'inverse, les assistants de preuves permettent de réaliser des preuves très générales, mais demandent une forte compétence.

L'exposé étudiera comment mélanger harmonieusement ces deux types d'outils afin de cumuler leurs avantages.