2018-00621 - Post-Doc Position/ Induction reasoning in SMT [S]

Niveau de diplôme exigé : Thèse ou équivalent
Fonction : Post-Doctorant

Contexte et atouts du poste

Many applications, notably in the context of verification (for critical systems in transportation, energy, etc.), rely on checking the satisfiability of logic formulas. Satisfiability-module-theories (SMT) solvers handle large formulas in expressive languages with built-in and custom operators (e.g. arithmetic and data structure operators). These tools are built using a cooperation of a SAT (propositional satisfiability) solver to handle the Boolean structure of the formula and theory reasoners to tackle the atomic formulas (e.g. x > y + z for the theory of arithmetic).

Currently, SMT solvers essentially handle first-order logic, and have very limited capabilities to reason by induction. In [1,8], the SMT solvers are used as black-boxes to discharge proof obligations of induction-based provers. Later on in [4], Reynolds and Kuncak proposed as solution the integration of explicit induction reasoning inside SMT solvers. Explicit induction schemas are used to perform the inductive strengthening of (the negation and skolemisation of) some universally quantified formula given as input. Their approach is able to automatically discover subgoals that can be then proved by induction. A disadvantage of the method is that it requires the definition of induction schemas before the proof starts, so it may happen that crucial induction hypotheses are missing or the provided induction hypotheses are unusable.

On the other hand, proofs by induction can also be automatically performed using implicit induction by the means of reductive inference systems [2, 3, 5, 6, 7], as shown when proving properties about conditional specifications. The lazy use of induction hypotheses, i.e., only when they are needed, and the easy development of mutual induction proofs are some of the advantages of this approach.

On the other hand, proofs by induction can also be automatically performed using implicit induction by the means of reductive inference systems [2, 3, 5, 6, 7], as shown when proving properties about conditional specifications. The lazy use of induction hypotheses, i.e., only when they are needed, and the easy development of mutual induction proofs are some of the advantages of this approach.

Currently, SMT solvers essentially handle first-order logic, and have very limited capabilities to reason by induction. In [1,8], the SMT solvers are used as black-boxes to discharge proof obligations of induction-based provers. Later on in [4], Reynolds and Kuncak proposed as solution the integration of explicit induction reasoning inside SMT solvers. Explicit induction schemas are used to perform the inductive strengthening of (the negation and skolemisation of) some universally quantified formula given as input. Their approach is able to automatically discover subgoals that can be then proved by induction. A disadvantage of the method is that it requires the definition of induction schemas before the proof starts, so it may happen that crucial induction hypotheses are missing or the provided induction hypotheses are unusable.

On the other hand, proofs by induction can also be automatically performed using implicit induction by the means of reductive inference systems [2, 3, 5, 6, 7], as shown when proving properties about conditional specifications. The lazy use of induction hypotheses, i.e., only when they are needed, and the easy development of mutual induction proofs are some of the advantages of this approach.

On the other hand, proofs by induction can also be automatically performed using implicit induction by the means of reductive inference systems [2, 3, 5, 6, 7], as shown when proving properties about conditional specifications. The lazy use of induction hypotheses, i.e., only when they are needed, and the easy development of mutual induction proofs are some of the advantages of this approach.

On the other hand, proofs by induction can also be automatically performed using implicit induction by the means of reductive inference systems [2, 3, 5, 6, 7], as shown when proving properties about conditional specifications. The lazy use of induction hypotheses, i.e., only when they are needed, and the easy development of mutual induction proofs are some of the advantages of this approach.

How to apply

Upload your file on jobs.inria.fr in a single pdf or zip file, and send it as well by email to sorin.stratulat@loria.fr and pascal.fontaine@loria.fr . Your file should contain the following documents:

- Your CV
- A cover/motivation letter describing your interest in this topic
- A short (max one page) description of your Master thesis (or equivalent) or of the work in progress if not yet completed
- Your degree certificates and transcripts for Bachelor and Master (or the last 5 years)
- Master thesis (or equivalent) if it is already completed and publications if any (it is not expected that you have any). Only the web links to these documents are preferable, if possible.

In addition, one recommendation letter from the person who supervises(d) your Master thesis (or research project or internship) should be sent directly by his/her author to sorin.stratulat@loria.fr and pascal.fontaine@loria.fr.

Applications are to be sent as soon as possible.

Conditions pour postuler

Sécurité défense :
Ce poste est susceptible d'être affecté dans une zone à régime restrictif (ZRR), telle que définie.

Informations générales

- Thème/Domaine : Preuves et vérification
- Ville : Villers-lès-Nancy
- Centre Inria : CRI Nancy - Grand Est
- Date de prise de fonction souhaitée : 01-10-2018
- Durée de contrat : 1 an, 4 mois
- Date limite pour postuler : 06-06-2018

Contacts

- Equipe Inria : VERIDIS
- Recruteur : Stratulat Sorin / sorin.stratulat@loria.fr

L'essentiel pour réussir

Application deadline

June 6th, 2018 (Midnight Paris time)
The proposed methods will be implemented as a prototype within the veriT SMT solver (http://www.verit-solver.org/).

Principales activités
The work can be organized as follows
- examine libraries of formal proofs to identify proof obligations requiring reasoning by induction, and build a library of benchmarks;
- develop an integration schema between induction and SMT:
  - as a first step, the ideas from reference [4] above will be implemented in the veriT solver;
  - then, the goal is to replace explicit by implicit induction reasoning;
- implement this integration schema;
- test the integration schema against the set of benchmarks.

Compétences
It is necessary to be acquainted with satisfiability modulo theories techniques. Knowledge of interactive theorem proving, reasoners by induction, or formal reasoning is a plus.

The development of prototypes will be done using the C language. Being familiar with a C-like language is thus required.

Avantages sociaux
- Subsidised catering service
- Partially-reimbursed public transport
- Social security
- Paid leave
- French courses

Rémunération
Salary: 2653€ gross/month