Lot 5 : Technologie de Vérification

PCRI (LRI,INRIA Futurs, LIX)

LORIA, LSV-ENS Cachan

Averroes, T0+18, Juin 2004

Objectifs

– Améliorer la performance des outils de vérification
  - ▷ Coq (logique d'ordre supérieur)
  - ▷ Elan (réécriture)
  - ▷ HCMC (model-checker)
– Favoriser des techniques de preuves sûres
– Collaboration entre outils

Sous-projets

5.1 Combiner raisonnement équationnel et déductif dans Coq

5.2 Mise en œuvre dans HCMC de la vérification compositionnelle (sous-lot 4.1)

5.3 Automatisation de la déduction

Sous-lot 5.1 : Noyau de **Coq** avec réécriture

Un projet à long terme

- 3 thèses (*théoriques*) dans le cadre de CALIFE
- Développement d'un prototype CAC (branche de Coq)
- Mécanisation des preuves de terminaison

Coq + réécriture : déduction modulo

$$\frac{t : T \qquad T \equiv U}{t : U}$$

Décidabilité de $T \equiv U$, cohérence...

Approches basées sur la réécriture

$$\text{Coq} \begin{cases} 0 + x & \longrightarrow x \\ S(x) + y & \longrightarrow S(x+y) \end{cases} \qquad \text{CAC} \begin{cases} 0 + x & \longrightarrow x \\ x + 0 & \longrightarrow x \\ \ldots & \longrightarrow \ldots \\ x + (y+z) & \longrightarrow (x+y)+z \end{cases}$$

Terminaison, modèles, machines de réduction ...

Architecture

Prototype permettant de tester la conversion étendue avec réécriture

- Ajout des symboles définis par réécritures au-dessus de CIC
- Communication avec CiME pour réduire les termes

Branche de l'archive Coq

Exemple

```
Symbol plus AC : nat → nat → nat.
Rules [x,y:nat] {
  (plus O x)      ⇒ x;
  (plus (S x) y) ⇒ S (plus x y) }
```

Terminaison

Deux DEA en cours (LORIA,LRI)

Rendre explicite des preuves sophistiquées de terminaison

- S. Hinderer au LORIA : preuve Coq explicite pour la terminaison par interprétation polynomiale, interprétation d'une trace de CiME.
- Th. Hubert au LRI : preuve Coq explicite pour déduire la terminaison de la réécriture à partir de la terminaison des paires de dépendances (méthode modulaire de preuve de terminaison).

Sous-lot 5.1 : Fournitures

5.1.1 Définition de la classe de réécriture à intégrer

5.1.2 Architecture du moteur de test de conversion

5.1.3 Prototype d'extension du système Coq.

Sous-lot 5.1 : Perspectives

- Test du prototype Coq+réécriture
- Améliorations du prototype
- Bibliothèque Coq pour les preuves explicites de terminaison
- Combinaison déduction et procédures de décision à la Shostak (Dan Hernest, PCRI-LIX)

Sous-lot 5.2 : Vérification compositionnelle

Mise en œuvre dans HCMC des techniques développées dans le sous-lot 4.1.

- Version à base de DBM (Difference Bounded Matrix) calculant une sur-approximation implantée.
- Intégration d'une bibliothèque de polyhèdres PPL (Parma Polyhedra Library) : calculs exacts pour la dernière vérification sur le processus `nil` vue comme un problème de résolution de contraintes.

Sous-lot 5.2 : Fournitures

Prototype d'extension aux p-automates de HCMC.

- Version approchée implantée.
- Version exacte avec PPL en cours de mise au point

Perspectives

Validation sur des exemples :

- PGM ou train d'atterrissage Airbus

Sous-lot 5.3 : Mécanisation de la déduction

– Coq-Elan : trace de réécriture en Elan réinterprétée en Coq

  – Trace de preuve produites par Elan
    exprimée dans le $\rho\sigma$-calcul (λ-termes + réécriture)
  – Traduction des $\rho\sigma$-termes en termes de preuve
  – Prise en compte de la récurrence : utiliser les égalités dans les hypothèses de récurrence (chargement dynamique)
  – Symboles AC, extension à la réécriture conditionnelle

Récurrence et déduction modulo

Lien entre preuve automatique par récurrence (Spike) et déduction modulo

- Principe de récurrence noetherienne représenté dans $HOL_{\lambda,\sigma}$ (présentation au premier ordre, modulo de HOL)
- Possibilité d'intégrer dynamiquement de nouvelles réécritures
- Relation noetherienne définie par réécriture
- Recherche de preuve par narrowing

Preuves déductives dans Coq

Contexte

- Termes de preuves explicites
- Logique intuitioniste
- Connecteurs inductifs : $\bot, \wedge, \vee, \exists, = \ldots$

Principe

Étendre les algorithmes de preuve automatique aux inductifs

Résultat

Tactique pour le fragment du premier ordre, implantée en Coq.

Sous-lot 5.3 : Fournitures

- Preuves par récurrence et réécriture dans Coq
  - ▷ Fourniture 5.3.1 + article CADE 2003
- Mécanisation du raisonnement du premier ordre dans Coq
  - ▷ Fourniture 5.3.2 + article TYPES 2003

Sous-lot 5.3 : Perspectives

– Coq-Elan : génération par Elan des ordres et schémas de récurrence
– Récurrence-modulo : Implantation de la recherche de preuve
– Coq : prise en compte de l'égalité (congruence-closure) et de la combinaison de théories.