The DemoNat project

Patrick Thévenon
patrick.thevenon@univ-savoie.fr

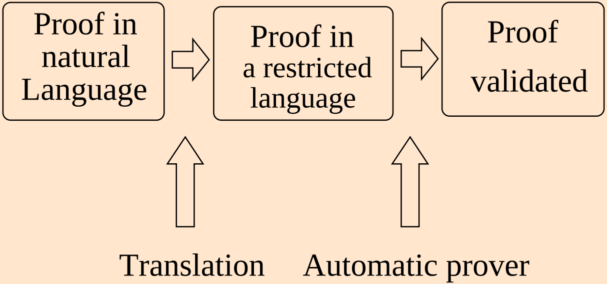
LAMA, Université de Savoie
Chambéry

18 April 2006
Types 2006
Nottingham

The DemoNat project

- Aim of the projet :
  - Develop a program able to
    Analyse and validate proofs in natural language
- What is it for :
  - Students can write proofs in a natural way
  - Faster to learn, because more intuitive
- Teams involved in the projet :
  - Lattice/Talana (Jussieu)
  - Calligramme (Nancy)
  - LAMA (Chambéry)

The system


Proof in natural Language
Proof in a restricted language
Proof validated
Translation
Automatic prover

My work in this project

- Practical :
  - Definition of a restricted language
  - Implementation of a prover
- Theoretical :
  - Principal type properties in a calculus with two arrows
  - Study of a logic system observed from the prover

The Restricted language

- Uses a small grammar (let, assume, prove, deduce,. . .)
- Allows to give hints to the prover (by, with)
- Describes a tree of logical (meta) rules
- To each rule is associated a formula that justifies it

The prover

- A resolution prover
- the prover is a functor (formulas are abstract type)
- To have a prover :
  - give a logic (definition of formulas, unification,. . .)
  - apply the functor to it
- Has been applied to
  - classical propositional and first order
  - will be used in PhoX, proof assistant
    developped by C. Raffalli

Lazy decomposition

- Problem : how to compute a set of clauses from a formula ?
- The justification of each step of a proof does not need to use the whole complexity of hypothesis
- We don't want to decompose everything while proving $F \longrightarrow F$
- The idea :
  - Decompose formulas during the proof search
  - clauses are sets of formulas
    (not necessarily atomic formulas)

A logical system from the prover

The logical rules (Propositional logic)

$$\frac{S = S'; \Gamma, A \vee B}{S; \Gamma, A, B} \qquad \frac{S = S'; \Gamma, \neg(A \vee B)}{S; \Gamma, \neg A} \qquad \frac{S = S'; \Gamma, \neg(A \vee B)}{S; \Gamma, \neg B}$$

Same with arrows ($A \rightarrow B = \neg A \vee B$)

$$\frac{S = S'; \Gamma, \neg(A \wedge B)}{S; \Gamma, \neg A, \neg B} \qquad \frac{S = S'; \Gamma, A \wedge B}{S; \Gamma, A} \qquad \frac{S = S'; \Gamma, A \wedge B}{S; \Gamma, B}$$

$$\frac{S = S'; \Gamma, A; \Gamma', \neg A}{S; \Gamma, \Gamma',}\,Res \qquad \frac{S = S'; \Gamma, A, A}{S; \Gamma, A}\,Contr$$

Logical rules

A Logical system

- dual to sequent calculus (elimination rules)
- similar to calculus of structures (non branching rules)

Aim

- Find a complete strategy of proof search :
  No resolution on formulas that are
  subformulas of unifiable formulas

The Last Slide

People interested can

- ask me for more information or a private demonstration
- Go see my web page : www.lama.univ-savoie.fr/~thevenon