Enhancing Proof Assistant Systems

Saarbrücken Exchange Visit, February 4-8, 2002

1. The Saarbrücken Cooperation
2. Proof planning for NUPRL
3. ActiveMath
4. Miscellaneous Contacts

Enhancing Proof Assistant Systems

- **Enhancing proof automation through AI-techniques**
- **Knowledge-acquisition for proof planning**
- **Developing standards for mathematics libraries**
- **Exchange of formalized mathematical knowledge**
- **Cross fertilization in development of user interfaces**

Joint NSF-DAAD project

Planned Visits of CU researchers

- Alexey Nogin

  Implementation of logic interface between MetaPRL *and* ΩMEGA

- Christoph Kreitz

  Automating Nuprl *proofs using* ΩMEGA*'s proof planners*

- Stuart Allen

  Semantical basis for an interface between Nuprl *and* ΩMEGA

- Christoph Kreitz

  Investigation of meta-reasoning in proof planning.

- Amanda Holland-Minkley

  Integrating Nuprl*'s and* ΩMEGA*'s mechanisms for proof presentation*

- NN

  Exploring the application domain of IO-automata

Saarbrücken People

- **Erica Melis:** Proof Planning, Analogy, Mathematics education systems, Machine Learning
- **Christoph Benzmüller:** Higher-Order Theorem Proving, Agent-based reasoning, GUI, natural language interface
- **Paul Libbrecht:** ActiveMath
- **Martin Pollet:** (Interactive) proof planning
- **Andreas Meier:** Proof Planning, integration of proof systems into ΩMEGA
- **Andreas Fiedler:** Proof presentation, natural language, cognitive models
- **Heiko Mantel:** Linear Logic Theorem Proving, Security, Refinement (DFKI)
- **Axel Schairer:** Theorem Proving, Security, Verification, Proof Reuse (DFKI)
- **Serge Autexier:** Theorem Proving, Software Verification (DFKI)
- **Dieter Hutter:** Software Verification, Induction, Proof Planning & Reuse (DFKI)
- **Michael Kohlhase :** MathWeb, OMDoc, Higher-Order Theorem Proving (CMU)

⋮

Use proof planning to guide proof development in NUPRL

- **JPROVER-like methodology for external proof guidance**
  – External tool (ΩMEGA) develops proof "plan"
  – NUPRL executes (and checks) proof

- **Cooperative proof development**
  – ΩMEGA plans proof while interacting with NUPRL instead of a-priori

- **Involve variety of proof search mechanisms**
  – First-order provers and decision procedures
  + Extension of Pientka's rippling-based proof strategy
  + Constraint Solvers and Computer Algebra Systems
  + Knowledge-based proof planning (reasoning about proof methods)
  ... make suggestions for meaningful proof steps

- **Testbed: automate specification proofs in NUPRL**

Knowledge-based Proof Planning in Ωmega

- **Methods:**
  - Abstract descriptions of common proof patterns or strategies
    e.g. premise + conclusion + application condition + proof schema
  - May be generic or application/domain-specific

- **Proof plan:**
  - Sketch of a real proof: partially ordered set of instantiated methods
  - Must be checked by expanding/executing corresponding tactics
  - Complete if expanded methods prove a given goal sequent

- **Planning $\equiv$ high-level proof search**
  - Planning search space smaller than at calculus level
  - Control rules express meta-knowledge (heuristics, preferences, intuition,. . .)
  - Asynchronous external reasoners may assist by providing suggestions
    - Theorem provers (EQP, SPASS, TPS, . . .) fill gaps in plans
    - CAS check conditions that involve symbolic computations
    - Constraint solvers instantiate meta-variables through satisfiability checks
    - PVS connected through MathWeb

Proposed Cooperation Nuprl $\leftrightarrow$ Ωmega

Nuprl : submits actual proof sequent to planner

receives proof tactic and parameters

executes tactics and sends results back to planner

Ωmega : receives sequent from Nuprl

consults its control mechanisms to develop (partial) plan

converts plan into sequence of tactics to be sent to Nuprl

evaluates results returned by Nuprl wrt. plan

Nuprl may always execute `Auto*` first

Ωmega may have to revise a plan

Nuprl may have to change prove nodes above the current sequent (transformation tactics)

Status

- **Generic strategy for planning specification proofs**
  - involves basic tactics and methods for dynamically introducing program control structures (case split, induction, while loop)
  - "tested" with $\sqrt{x}$, $|x-y|$, $gcd(x,y)$, $log_2 x$, ... (Details??)
- **Agenda**
  - Abstract description of tactics and methods used (application conditions?)
  - Precise description / implementation of planning strategy
  - Define requirements for communication NUPRL↔ΩMEGA (Stuart?)
  - Handish experiments until NUPRL and ΩMEGA can communicate
  - Build NUPRL↔ΩMEGA through MATHWEB/OMDOC (M. Kohlhase?)

  ... *write paper to keep NSF/DAAD happy*

ActiveMath

Mathematics education system using AI-techniques

- **Web-based learning environment**
- **Generates interactive mathematical courseware**
  - Content extracted from knowledge base
  - Material selection follows known pedagocical rules
  - Sources based on the OMdoc XML syntax
  - Tracks dependencies between concepts, lemmata, etc.
- **Students can practice interactive problem solving**
  - Exercises and proofs reveal more insights/details
  - Exercises may invoke reasoning systems (CAS, Constraint Solver, Planner)
- **A "popular" outlet for material created with Nuprl?**
  - Use Nuprl's TR generation feature and Nuprl - XML translator

ADDITIONAL DISCUSSIONS

- **Matrix-characterization for $\mathcal{MELL}$** (H. Mantel)
  - Foundation for extending JPROVER to Linear Logic
  - Supports reasoning about resource use
  - *Status*: paper about to be finished
- JPROVER **enhancements** (S. Autexier)
  - Dynamically use multiple instances of quantified formula
  - *Status*: PhD thesis soon to be finished, uses much of our work
- **Proof Reuse** (E. Melis, A. Schairer, D. Hutter)
  - Links D. Kozen's calculus for publishing/citation to automated deduction
  - *Status*: Discussed papers on automatic lemma extraction, details to be integrated into Dexter's approach, then into NUPRL

Insights

- **The Saarbrücken group offers many valuable contacts**
- Nuprl **can profit from proof planning**
- **The ActiveMath experiences show how to make formal mathematics available to the "public"**
- **We need an interface between the systems**
  - MathWeb/OMDoc (Kohlhase, Stuart, Lori, ...)
- **We should join forces in creating a user interface**
  - Send "gifted" PhD student around? Saarbrücken may have one