IsaPlanner: A Proof Planner for Isabelle

Lucas Dixon

December 17, 2004

Motivation

- Improve automation in Isabelle by using tools from proof planning
- A generic approach?
- Take advantage of the proof tools available in Isabelle
- Expressing the reasoning process and the resulting proof plans in a *readable* form (Surveyable proofs)
- Proof planning to aid the interactive user

IsaPlanner Approach to Proof Planning

- Encode Techniques: capture common patterns of reasoning
- Applying a technique generates high level descriptions of the proofs called *proof plans* (a representation of proof scripts)
- Our approach derives Isar style proof scripts.
- Rippling: a difference removal technique, principally used to guide rewriting of the step case during inductive proofs

IsaPlanner

- A framework for encoding and applying reasoning techniques that derives readable proof scripts.
- Mechanisms for sharing non-logical information between proof techniques.
- Applying proof techniques produces Isar proof scripts
  - Allow human inspection of the proof attempt
  - Support debugging of proof
  - Support the exploration of proof
- Provide well-behaved techniques

Hierarchy of Behaviour

1. Bug
2. Non termination
3. Uninformative result (big goals, exceptions, "no")
4. Informative result (meaningful progress)
5. Proof/Refutation

Example Generated Proof Script

theorem `sum_of_odds:` $(\sum_{i<n}\ 2\ *\ i\ +\ 1)\ =\ n^2$
  proof `(induct n)`
    show $(\sum_{i<0}\ 2\ *\ i\ +\ 1)\ =\ 0^2$ **by** `simp`
  next
    fix n
    assume `IH:` $(\sum_{i<n}\ 2\ *\ i\ +\ 1)\ =\ n^2$
    have $(\sum_{i<n}\ 2\ *\ i\ +\ 1)\ +\ ((2\ *\ n)\ +\ 1)\ =\ (n\ +\ 1)^2$
      gap `(conjecture_lemma` $n^2\ +\ ((2\ *\ n)\ +\ 1\ =\ (n\ +\ 1)^2)$
    thus $(\sum_{i<(n\ +\ 1)}\ 2\ *\ i\ +\ 1)\ =\ (n\ +\ 1)^2$ **by** `rippling`
  qed

Example Generated Proof Script

theorem `sum_of_odds:` $(\sum_{i<n}\ 2\ *\ i\ +\ 1)\ =\ n^2$

  proof `(induct n)`

    show $(\sum_{i<0}\ 2\ *\ i\ +\ 1)\ =\ 0^2$ **by** `simp`

  next

    fix n

    assume `IH:` $(\sum_{i<n}\ 2\ *\ i\ +\ 1)\ =\ n^2$

    have $(\sum_{i<n}\ 2\ *\ i\ +\ 1)\ +\ ((2\ *\ n)\ +\ 1)\ =\ (n\ +\ 1)^2$

      gap `(conjecture_lemma` $n^2\ +\ ((2\ *\ n)\ +\ 1\ =\ (n\ +\ 1)^2)$

    thus $(\sum_{i<(\ n\ +\ 1\ )}\ 2\ *\ i\ +\ 1)\ =\ (n\ +\ 1)^2$ **by** `rippling`

  qed

Proof Planning?

- Declarative description (datatype) of Isar proof scripts
- Compared with tactics:
  - More information - history of the proof
  - Surveyable proofs
- Mix human/computer manipulation of 'readable' formal proof

Encoded Techniques

- Induction, Case splitting, Equational Reasoning
- Higher order rippling (provides rewriting)
- Lemma speculation and generalisation critics
- This combination forms a powerful inductive theorem prover

Some Observations

- Supports *exploration* of Isar proofs
- Can prove some apparently complicated problems, for example: $(x^y)^z = x^{(y \cdot z)}$ in ordinal arithmetic.
- Conjecturing lemmas is a directed form of theory formation
- Unsuccessful branches/failed proof attempts can still result in proving useful lemmas
- Caching conjectures (proved and unprovable) prunes the search space significantly

Some Issues

- Where to draw the line between work done by a tactic and work done by proof planning?
- What information to store in source files for proof re-execution?
- Interface issues: how to use support interactive proof planning? (Proof by contextual menu!)