The ACL2 Theorem Prover: Round 2

By Gabe Shelley and

Steve Forrest

IMPS vs. ACL2

IMPS

UI ENVIRONMENT

- Emacs

LOGIC

- Simple Type Theory

THEOREM PROVING

- interactive environment
- use of tactics possible

PROOF-SCRIPTS

- somewhat legible
- generated by deduction graph

THEORIES

- theory is set of axioms
- Support for theory interpretations

ACL2

UI ENVIRONMENT

- Text-based + Emacs for Proof Trees
- DrACuLa: ACL2 in DrScheme
- ACL2(s) – Eclipse plugin

LOGIC

- FOL + Recursive functions

THEOREM PROVING

- highly automated
- can provide hints to prover

PROOF-SCRIPTS

- very legible (write your own!)

THEORIES

- theory is set of “runes” (rule names)
- No support for theory interpretations; set operations on theories

Example of Proof Script

```
(defthm integer-implies-square-is-integer
   (implies (integerp u) (integerp (* u u)) )
   :rule-classes nil
)

(defthm even-square-implies-even-square-divisible-by-4
  (implies (and (integerp p) (evenp (* p p)))
           (integerp (* 1/4 p p)))
  :hints (("Goal"
           :use ((:instance even-square-implies-even)
                 (:instance integer-implies-square-is-integer (u (* 1/2 p)) )
           )
           :in-theory (disable even-square-implies-even)
  ))
)
```

Applications

AMD5k86

- Verification of floating point division micro-code

Motorola CAP (complex arithmetic processor) digital signal processor

Java Virtual Mchine

Proving theorems about JVM model behaviour when interpreting bytecodes

Questions

Any questions or comments?

References

Moore, J. S. – ‘An ACL proof of Write Invalidate Cache Coherence’, http://citeseer.ist.psu.edu/cache/papers/cs/1068/http

Kaufmann, M., Moore, J. S., An Industrial Strength Theorem Prover for a Logic Based on Common Lisp, http://www.cs.utexas.edu/users/moore/publications/k

Moore, J. S., ACL2 Proof Demonstration, http://www.cs.utexas.edu/users/moore/publications/demos.html