Rippling in PVS

A. A. Adams[1] and L. A. Dennis[2]

[1] School of Systems Engineering, The University of Reading, P.O. Box 225, Whiteknights, Reading, RG6 6AY
A.A.Adams@Rdg.ac.uk
http://www.rdg.ac.uk/~sis00aaa/

[2] School of Computer Science and iT, Jubilee Campus The University of Nottingham, Wollaton Road, Nottingham, NG8 1BB
lad@cs.nott.ac.uk
http://cs.nott.ac.uk/~lad/

Rippling is a method of controlling rewriting of the terms in an induction step of an inductive proof, to ensure that a position is reached whereby the induction hypothesis can be applied (see [1] for a full discussion). Rippling was developed primarily by the Mathematical Reasoning Group at the University of Edinburgh. The primary implementations are in the two proof planning systems Clam and λClam. An implementation is also available in HOL. We plan to implement rippling as a tactic for automatic generation of proofs requiring induction in PVS. Rippling has mostly been used as part of a larger project for developing high-level proof strategies, but has rarely been applied to "real-world" examples. Once we have this implementation we intend to assess the utility of this as a tactic by running rippling on the large number of inductive proofs developed by Gottliebsen as part of the PVS Real Analysis library [2]. By comparing the performance of the automation offered by rippling on these proofs with the original proof, which were proved by a combination of hand-generation of proofs and by existing PVS strategies, we hope to assess the utility of rippling as a technique for real-world applications.

References

1. Alan Bundy. The automation of proof by mathematical induction. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning, Volume 1*. Elsevier, 2001.
2. H. Gottliebsen. Transcendental Functions and Continuity Checking in PVS. pages 198–215. Springer-Verlag LNAI 1869, 2000.