HOL4 Tactics Cheat Sheet

Heiko Becker

November 14, 2017

1 Rewriting tactics

1.1 \texttt{asm\_rewrite\_tac} : \texttt{thm list \rightarrow tactic}

simple rewriting tactic, taking assumptions into account, may diverge

1.2 \texttt{once\_rewrite\_tac} : \texttt{thm list \rightarrow tactic}

simple rewriting tactic, taking assumptions into account cannot diverge since each rewrite is done at most once

2 Simplification

2.1 \texttt{simp} : \texttt{thm list \rightarrow tactic}

stronger than rewriting, takes assumptions into account

2.2 \texttt{fs} : \texttt{thm list \rightarrow tactic}

strongest simplification tactic, does automatic case splits on disjunctions, performs arithmetic simplification as well as rewriting

2.3 \texttt{rveq} : \texttt{tactic}

equivalent to Coq’s subst tactic, removes simple equalities from assumptions

3 Logic

3.1 \texttt{conj\_tac} : \texttt{tactic}

conjunction introduction rule
3.2 DISJ_CASES_TAC : thm \rightarrow tactic
    case split on disjunction

3.3 match_mp_tac : thm \rightarrow tactic
    modus ponens with first order matching

3.4 ho_match_mp_tac : thm \rightarrow tactic
    modus ponens with higher order matching

3.5 drule ?
    similar to match_mp_tac but adds matched lemma as implication

3.6 Induct : tactic
    simple induction tactic that "guesses" induction variable

3.7 Induct_on : term quotation \rightarrow tactic
    induction tactic taking term as a parameter on which induction is to be performed

3.8 Cases_on : term quotation \rightarrow tactic
    case split tactic

4 Assumption manipulation

4.1 gen_tac : tactic
    \forall, \Rightarrow elimination tactic (same as intros in Coq)

4.2 strip_tac : tactic
    more general version of gen_tac, can also split conjunctions and disjunctions
4.3 qpat_x_assum : term quotation → (thm → tactic) → tactic
find according to term quotation a matching assumption and then apply
the given thm → tactic function to it useful for removing assumptions or
specializing them

4.4 qspecl_then : term quotation list → (thm → tactic) → thm → tactic
specialization with automatic type inference, takes continuation tactic that
is provided the specialized thm

4.5 ASSUME_TAC : thm → tactic
add given theorem to assumption list

5 Useful tactic "extensions"

5.1 rpt : tactic → tactic
repeat given tactic until it fails

5.2 kall_tac : thm → tactic
"kill" theorem which is given as parameter

5.3 CHANGED_TAC : tactic → tactic
fails if provided tactic does not change the goal

6 \(\alpha\) - renaming

6.1 rename1 : term quotation → tactic
rename assumptions and goal according to term quotation

7 Automation

7.1 metis_tac : thm list → tactic
automated theorem prover for First-Order logic with equality
7.2 prove_tac: thm list → tactic

automatic, semi-complete automated theorem prover for

8 Conversions / Rules

8.1 ONCE_REWRITE_CONV: thm list → term → thm

once rewrite given theorems in term, providing equality theorem

8.2 ONCE_REWRITE_RULE : thm list → thm → thm

once rewrite given theorems in given theorem, providing only resulting theorem after rewriting

8.3 GSYM: thm → thm

symmetry of equality for theorems (rewriting in HOL4 is always from left to right)

8.4 EVAL: term → thm

call-by-value evaluation of term