Sugar 2.0 in HOL

a deep embedding

Mike Gordon

(with help from Cindy Eisner and Dana Fisman of IBM)

UNIVERSITY OF CAMBRIDGE
Computer Laboratory
Automated Reasoning Group
Hardware Verification Group

IBM's *Sugar 2.0* selected by *Accellera* as the industry standard property language

- LTL based, but incorporates features from ITL (regular expressions) and CTL
  - $\{r_1\} \mapsto \{r_2\}$, $\{r_1 \&\& r_2\}$ (ITL), $X!f$, $[f_1 U f_2]$ (LTL), $EX f$, $E[f_1\ U\ f_2]$ (CTL)
- supports infinite paths (for model checking) and finite paths (for simulation run checking)
- has both clocked and unclocked semantics (equivalent for trivial clocks)
  - $\pi \overset{c}{\models} f$ — formula f holds for (finite or infinite) path π when weakly clocked on c
  - $\pi \overset{c!}{\models} f$ — formula f holds for (finite or infinite) path π when strongly clocked on c
- most constructs are defined from a small kernel (name 'sugar' from 'syntactic sugar')
  - $b[=i] = \{\neg b[*]; b\}[*i]; \neg b[*]$, $within(r_1, b)\{r_2\} = \{r_1\} \mapsto \{r_2 \&\& b[=0]; b\}$
  - $[f_1\ W\ f_2] = [f_1\ U\ f_2] \vee G f_1$, $next\_event(b)(f) = [\neg b\ W\ b \wedge f]$

Semantics in HOL's classical higher order logic is a straightforward deep embedding

Sugar syntax

$[f_1\ U\ f_2]$

HOL representation

$\texttt{F\_UNTIL}(f_1, f_2)$

Sugar semantics

$\pi \overset{c!}{\models} [f_1\ U\ f_2] \iff$ there exist i and $k \geq i$ s.t. $\hat{L}(\pi^{0,i}) \overset{T}{\models} \{\neg c[*]; c\}$, $\pi^k \overset{T}{\models} c$, $\pi^k \overset{c!}{\models} f_2$, and for every j s.t. $i \leq j < k$ and $\pi^j \overset{T}{\models} c$: $\pi^j \overset{c!}{\models} f_1$

HOL representation

```
F_SEM M p (STRONG_CLOCK c) (F_UNTIL(f1,f2)) =
 ∃i k ∈ PL p. k ≥ i ∧
   FIRST_RISE M p c i ∧
   F_SEM M (RESTN p k) (WEAK_CLOCK T) (F_BOOL c) ∧
   F_SEM M (RESTN p k) (STRONG_CLOCK c) f2 ∧
   ∀j ∈ PL p. i ≤ j ∧ j < k ∧
      F_SEM M (RESTN p j) (WEAK_CLOCK T) (F_BOOL c)
      ⇒
      F_SEM M (RESTN p j) (STRONG_CLOCK c) f1)
```

Typical examples of minor errors found by attempting to prove 'sanity checking' properties

Original semantics	Corrected semantics
$\pi \overset{c}{\models} b \iff$ if there exists i: $\hat{L}(\pi^{0,i}) \overset{T}{\models} \{\neg c[*]; c\}$ then $L(p_i) \models b$	$\pi \overset{c}{\models} b \iff$ for every i s.t. $\hat{L}(\pi^{0,i}) \overset{T}{\models} \{\neg c[*]; c\}$, $L(p_i) \models b$
$\pi \models \{r_1\} \mapsto \{r_2\} \iff$ either for every j such that $\hat{L}(\pi^{0,j}) \models r_1$ there exists k such that $\hat{L}(\pi^{j,k}) \models r_2$, or for every j such that $\hat{L}(\pi^{0,j}) \models r_1$ and for every k there exists a finite word w such that $\hat{L}(\pi^{j,k})w \models r_2$	$\pi \models \{r_1\} \mapsto \{r_2\} \iff$ for every j such that $\hat{L}(\pi^{0,j}) \models r_1$, either there exists k such that $\hat{L}(\pi^{j,k}) \models r_2$, or for every k there exists a finite word w such that $\hat{L}(\pi^{j,k})w \models r_2$

HOL deduction can be used to derive and verify proof rules

McMillan's 'circular inference rule'

'$\neg[f_2 U \neg f_1]$' means 'f_2 up to $t-1$ implies f_1 at t',
'$\neg[f_1 U \neg f_2]$' means 'f_1 up to $t-1$ implies f_2 at t',
so:

$$\frac{\neg[f_2 U \neg f_1], \qquad \neg[f_1 U \neg f_2]}{G(f_1 \wedge f_2)}$$

An iteration rule (*c.f.* Hoare Logic while-rule)

assume functions f, b and g satisfy:
$\forall x.\ f(x) = \texttt{if}\ b(x)\ \texttt{then}\ f(g(x))\ \texttt{else}\ x$
and '$<x>$' means 'the current state is x', then:

$$\frac{\forall x.\ G(<x> \rightarrow X!<g(x)>)}{\forall x.\ <x> \rightarrow next\_event(\neg b)<f(x)>}$$