Finding loop invariants using tree grammars

Gabriel Ebner

January 28, 2015

Problem

$$\{\top\}$$

for $\mathtt{i} < \mathtt{n}$ **do** $\mathtt{x} := \mathrm{set}(\mathtt{x}, \mathtt{i} + 1, \mathrm{get}(\mathtt{x}, \mathtt{i}))$ **od**

$$\{\forall y \leq \mathtt{n}\ \mathrm{get}(\mathtt{x}, y) = \mathrm{get}(\mathtt{x}, 0)\}$$

Problem

$$\{\top\}$$

for $\mathtt{i} < \mathtt{n}$ **do** $\mathtt{x} := \mathrm{set}(\mathtt{x}, s(\mathtt{i}), \mathrm{get}(\mathtt{x}, \mathtt{i}))$ **od**

$$\{k \leq \mathtt{n} \rightarrow \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)\}$$

Problem

$$\{\top\}$$
$$\textbf{for } \mathtt{i} < \mathtt{n} \textbf{ do } \mathtt{x} := \mathrm{set}(\mathtt{x}, s(\mathtt{i}), \mathrm{get}(\mathtt{x}, \mathtt{i})) \textbf{ od}$$
$$\{k \leq \mathtt{n} \rightarrow \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)\}$$

$$\Gamma = \{\forall x\, x \leq 0 \rightarrow x = 0, \qquad \text{(I0)}$$
$$\ldots \text{ other axioms for arithmetic } \ldots$$
$$\forall x \forall y \forall z\, \mathrm{get}(\mathrm{set}(x, y, z), y) = z, \qquad \text{(ge)}$$
$$\ldots\}$$

Problem

$$\{\top\}$$
$$\textbf{for } \mathtt{i} < \mathtt{n} \textbf{ do } \mathtt{x} := \mathrm{set}(\mathtt{x}, s(\mathtt{i}), \mathrm{get}(\mathtt{x}, \mathtt{i})) \textbf{ od}$$
$$\{k \leq \mathtt{n} \rightarrow \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)\}$$

$$\Gamma = \{\forall x\, x \leq 0 \rightarrow x = 0, \qquad \text{(I0)}$$
$$\ldots \text{ other axioms for arithmetic } \ldots$$
$$\forall x \forall y \forall z\, \mathrm{get}(\mathrm{set}(x, y, z), y) = z, \qquad \text{(ge)}$$
$$\ldots\}$$

Generate a proof π with a loop invariant:

$$\mathrm{get}(\mathtt{x}, \mathtt{i}) = \mathrm{get}(\mathtt{x}, 0) \wedge k \leq i \rightarrow \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)$$

Problem

$$\{\top\}$$
$$\textbf{for } \mathtt{i} < \mathtt{n} \textbf{ do } \mathtt{x} := \mathrm{set}(\mathtt{x}, s(\mathtt{i}), \mathrm{get}(\mathtt{x}, \mathtt{i})) \textbf{ od}$$
$$\{k \leq \mathtt{n} \to \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)\}$$

$$\Gamma = \{\forall x\, x \leq 0 \to x = 0, \qquad \text{(I0)}$$
$$\ldots \text{ other axioms for arithmetic } \ldots$$
$$\forall x \forall y \forall z\, \mathrm{get}(\mathrm{set}(x, y, z), y) = z, \qquad \text{(ge)}$$
$$\ldots\}$$

Generate ~~a proof π with~~ a loop invariant:

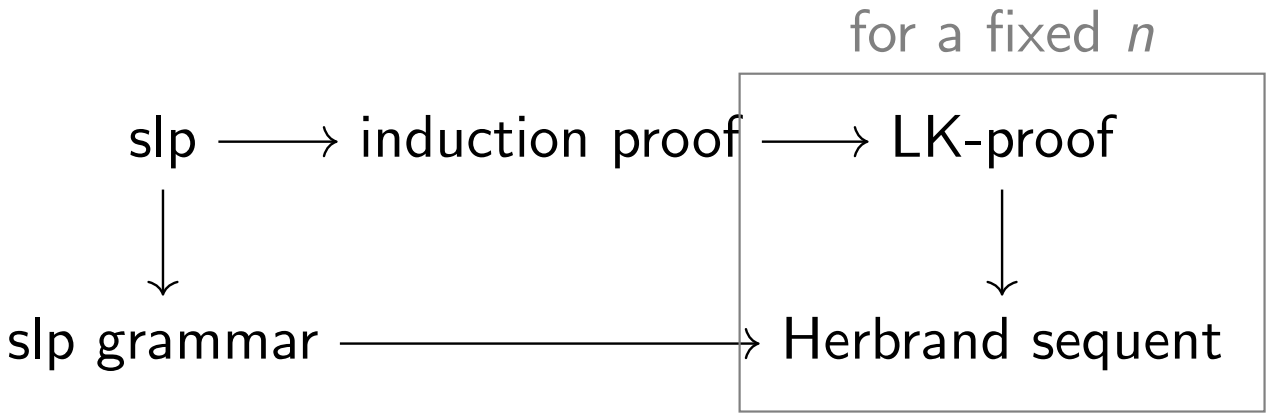
$$\mathrm{get}(\mathtt{x}, \mathtt{i}) = \mathrm{get}(\mathtt{x}, 0) \wedge k \leq i \to \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)$$

Verification conditions

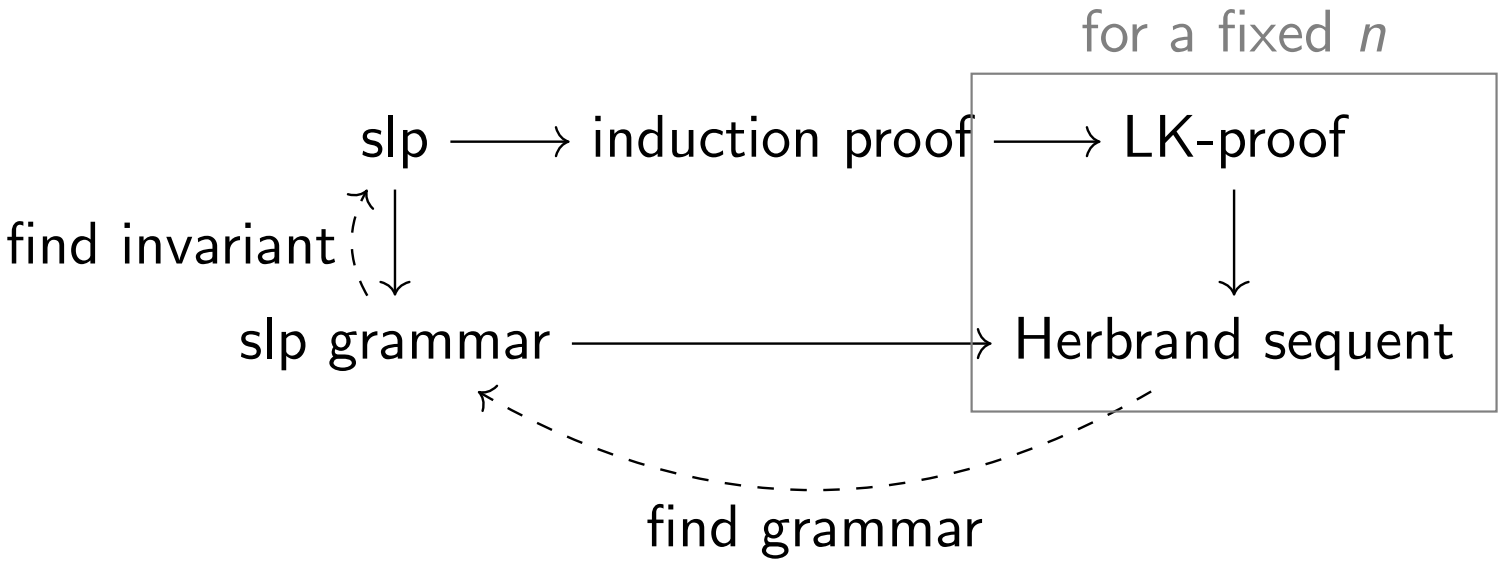
$X(\mathtt{i})$ is a loop invariant if

$$\{\top\}\ \textbf{skip}\ \{X(0)\}$$
$$\{X(\mathtt{i})\}\ \mathtt{x} := \mathrm{set}(\mathtt{x}, s(\mathtt{i}), \mathrm{get}(\mathtt{x}, \mathtt{i}))\ \{X(s(\mathtt{i}))\}$$
$$\{X(\mathtt{n})\}\ \textbf{skip}\ \{k \leq \mathtt{n} \rightarrow \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)\}$$

Big picture


for a fixed n
slp
induction proof
LK-proof
slp grammar
Herbrand sequent

Big picture


for a fixed n
slp
induction proof
LK-proof
find invariant
slp grammar
Herbrand sequent
find grammar

Induction problem

$$\{A(\mathtt{x})\}$$
$$\textbf{for } \mathtt{i} < \mathtt{n} \textbf{ do } \mathtt{x} := f(\mathtt{x}) \textbf{ od}$$
$$\{B(\mathtt{x})\}$$

$$\Downarrow (\mathtt{x} \mapsto \sigma(i))$$

$$\Gamma, A(\sigma(0)),$$
$$\quad \forall i\ \sigma(s(i)) = f(\sigma(i))$$
$$\vdash B(\sigma(n))$$

Induction problem

$$\{\top\}$$
$$\textbf{for } \mathtt{i} < \mathtt{n} \textbf{ do } \mathtt{x} := \mathrm{set}(\mathtt{x}, s(\mathtt{i}), \mathrm{get}(\mathtt{x}, \mathtt{i})) \textbf{ od}$$
$$\{k \leq \mathtt{n} \rightarrow \mathrm{get}(\mathtt{x}, k) = \mathrm{get}(\mathtt{x}, 0)\}$$

$$\Downarrow (\mathtt{x} \mapsto \sigma(i))$$

$$\Gamma, \top,$$
$$\quad \forall i\ \sigma(s(i)) = \mathrm{set}(\sigma(i), s(i), \mathrm{get}(\sigma(i), i))$$
$$\vdash k \leq n \rightarrow \mathrm{get}(\sigma(n), k) = \mathrm{get}(\sigma(n), 0)$$

LK-proof ($n = 2$)

$$\Gamma, \top,$$
$$\quad \forall i\ \sigma(s(i)) = \mathrm{set}(\sigma(i), s(i), \mathrm{get}(\sigma(i), i))$$
$$\vdash k \leq n \to \mathrm{get}(\sigma(n), k) = \mathrm{get}(\sigma(n), 0)$$

$$\Downarrow (n \mapsto 2)$$

$$\Gamma, \top,$$
$$\quad \forall i\ \sigma(s(i)) = \mathrm{set}(\sigma(i), s(i), \mathrm{get}(\sigma(i), i))$$
$$\vdash k \leq 2 \to \mathrm{get}(\sigma(2), k) = \mathrm{get}(\sigma(2), 0)$$

Herbrand sequent ($n = 2$)

We only really care about instances of Γ.

$$
\begin{aligned}
& k \leq 0 \to k = 0, \\
& \ldots \text{other instances of arithmetical axioms} \ldots \\
& \mathrm{get}(\mathrm{set}(\sigma(0), 1, \mathrm{get}(\sigma(0), 0)), 1) = \mathrm{get}(\sigma(0), 0), \\
& \mathrm{get}(\mathrm{set}(\sigma(1), 2, \mathrm{get}(\sigma(1), 0)), 2) = \mathrm{get}(\sigma(1), 0), \\
& \ldots \\
& \vdash
\end{aligned}
$$

Herbrand sequent ($n = 2$, term language version)

We only really care about instances of Γ.

$$
\begin{aligned}
L = \{ & r_{10}(k), \\
& \ldots \text{other instances of arithmetical axioms} \ldots \\
& r_{\mathrm{ge}}(\sigma(0), 1, \mathrm{get}(\sigma(0), 0)), \\
& r_{\mathrm{ge}}(\sigma(1), 2, \mathrm{get}(\sigma(1), 0)), \\
& \ldots \}
\end{aligned}
$$

slp grammar

We only really care about instances of Γ.

$$\tau \to r_{10}(k)$$

. . . other instances of arithmetical axioms . . .

$$\tau \to r_{\mathrm{ge}}(\sigma(\nu), s(\nu), \mathrm{get}(\sigma(\nu), 0))$$

. . .

Finding loop invariants

Boolean unification problem:

$$\Gamma_0, A \vdash X(0)$$
$$\Gamma_1, X(\nu) \vdash X(s(\nu))$$
$$\Gamma_2, X(n) \vdash B$$

$\Gamma_0, \Gamma_1, \Gamma_2, A, X(\nu), B$ are all quantifier-free!

Necessary conditions for X

$$\Gamma_0, A \vdash X(0)$$
$$\Gamma_1(\nu), X(\nu) \vdash X(s(\nu))$$
$$\Gamma_2, X(n) \vdash B$$

Necessary conditions for X

$$\Gamma_0, A \vdash X(0)$$
$$\Gamma_1(\nu), X(\nu) \vdash X(s(\nu))$$
$$\Gamma_2, X(n) \vdash B$$

$$\underbrace{\Gamma_0 \wedge A}_{C_0} \vdash X(0)$$

Necessary conditions for X

$$\Gamma_0, A \vdash X(0)$$
$$\Gamma_1(\nu), X(\nu) \vdash X(s(\nu))$$
$$\Gamma_2, X(n) \vdash B$$

$$\underbrace{\Gamma_0 \wedge A \wedge \Gamma_1(0)}_{C_1} \vdash X(1)$$

Necessary conditions for X

$$\Gamma_0, A \vdash X(0)$$
$$\Gamma_1(\nu), X(\nu) \vdash X(s(\nu))$$
$$\Gamma_2, X(n) \vdash B$$

$$\underbrace{\Gamma_0 \wedge A \wedge \Gamma_1(0) \wedge \Gamma_1(1)}_{C_2} \vdash X(2)$$

Necessary conditions for X

$$\Gamma_0, A \vdash X(0)$$
$$\Gamma_1(\nu), X(\nu) \vdash X(s(\nu))$$
$$\Gamma_2, X(n) \vdash B$$

$$\underbrace{\Gamma_0 \wedge A \wedge \Gamma_1(0) \wedge \Gamma_1(1) \wedge \Gamma_1(2)}_{C_3} \vdash X(3)$$

Algorithm for invariant finding

$$\underbrace{\Gamma_0 \wedge A \wedge \Gamma_1(0) \wedge \Gamma_1(1)}_{C_2} \vdash X(2)$$

- Compute C_2
- Find a consequence of C_2
- Replace some occurrences of 2 by ν
- Check if it is a solution

Algorithm for invariant finding

$$\underbrace{\Gamma_0 \wedge A \wedge \Gamma_1(0) \wedge \Gamma_1(1)}_{C_2} \vdash X(2)$$

- Compute C_2
- Find a consequence of C_2
- Replace some occurrences of 2 by ν
- Check if it is a solution

Theoretically complete!

Summary

General method for loop verification:

- For-loops with loop-free body
- Δ_0 pre- and post-condition
- Π_1 loop invariant (single quantifier only)

Future work:

- Implementation
- Nested loops
- Quantifier blocks