Some Work in Progress

Higher-Order Horn Clauses
and
Higher-Order Model Checking

Martin Lester Luke Ong Steven Ramsay

University of Oxford

I. Motivation

```
sum(n:int) =
  c := 0
  while (n > 0) do
    c := c + n
    n := n - 1
  assert (c >= n)
```

$$P \subseteq \mathbb{Z} \times \mathbb{Z}$$

$$\forall nc. \quad c = 0 \Rightarrow P\, n\, c$$

$$\forall nc. \quad n > 0 \land P\, n\, c \Rightarrow P\, (n-1)\, (c+n)$$

$$\forall nc. \quad n \leq 0 \land P\, n\, c \Rightarrow c \geq n$$

$$c = 0 \Rightarrow P\, n\, c$$

$$n > 0 \wedge P\, n\, c \Rightarrow P\,(n-1)\,(c+n)$$

$$n \leq 0 \wedge P\, n\, c \Rightarrow c \geq n$$

$$P \mapsto \lambda xy.\,(y \;=\; 0 \;\vee y \;\geq x)$$

$$Th \;\models\; \forall nc.\; n > 0 \wedge (c = 0 \vee c \geq n) \Rightarrow (c + n = 0 \vee c + n \geq n - 1)$$

```
repeat (f: int -> int) (s: int) (n: int) : int =
  if n <= 0 then s else f (repeat f s (n-1))

succ (u:int) = u + 1

assert (repeat succ 0 n >= n)
```

$$Repeat : (int \to int \to bool) \to int \to int \to int \to bool$$

$$n \leq 0 \wedge r = s \Rightarrow Repeat\ f\ s\ n\ r$$

$$n > 0 \wedge f\ z\ r \wedge Repeat\ f\ s\ (n-1)\ z \Rightarrow Repeat\ f\ s\ n\ r$$

$$v = u + 1 \Rightarrow Succ\, u\, v$$

$$Repeat\ Succ\ 0\ n\ r \ \Rightarrow r\ \geq n$$

II. Higher-Order Horn Clauses

Fix a (first-order), sorted assertion/constraint language:

$$\langle Sig,\ Th,\ Fm\rangle$$

Consider higher-sorts:

$$\iota ::= int \mid \cdots$$

$$\rho ::= bool \mid \iota \to \rho \mid \rho \to \rho$$

Assume countably many variables *Vars* *x,y,z*… of each sort

Terms *s,t,u*… are built from *Sig* and *Vars* using application.

Atomic formulas:

$$\phi \in Fm \qquad\qquad x\, t_1 \ \cdots\ t_k : bool$$

General formulas built in the usual way, with quantification at all relational sorts.

Fix a family of sets A_ι in which to interpret each Sig sort ι.

Interpret general formulas *s* inside in the standard model over $(A_\iota)_\iota$:

$$\begin{aligned} D_\iota &:= A_\iota \\ D_{bool} &:= \mathbb{B} \\ D_{\sigma_1 \to \sigma_2} &:= D_{\sigma_1} \Rightarrow D_{\sigma_2} \end{aligned}$$

e.g.

$$\langle \mathbb{Z}, \ldots \rangle, \alpha \models \forall x : (int \to bool) \to bool.\ s$$

iff

$$\text{for all } d \in (\mathbb{Z} \Rightarrow \mathbb{B}) \Rightarrow \mathbb{B}\text{:} \quad \langle \mathbb{Z}, \ldots \rangle, \alpha[x \mapsto d] \models s$$

Higher-Order Constrained Horn Clause Problem:

Fix a distinguished subset of relational variables, *RelVars*:

$$R_1 : \rho_1, \; \ldots, \; R_k : \rho_k$$

Fix a set of horn clauses *H* over *RelVars* (the free variables of H are in *RelVars*):

$$B ::= true \mid x\, t_1 \;\cdots\; t_k \mid \phi \mid B \wedge B$$

$$H \; ::= \forall x{:}\rho.\; H \mid B \Rightarrow R_i\, x_1 \;\cdots x_k \mid B \Rightarrow \phi$$

Determine if, for all models *A* of *Th*, there is some valuation α of *RelVars* such that:

$$A, \, \alpha \models H \qquad \left(P \wedge \forall \vec{x}.\, s \Rightarrow \phi \right)$$

In quantifier free linear arithmetic interpreted by $Th(\mathbb{Z}, 0, 1, +, -, \leq)$...

$$Succ : int \to int \to bool$$
$$Repeat : (int \to int \to bool) \to int \to int \to int \to bool$$

$$n \leq 0 \wedge r = s \Rightarrow Repeat\ f\ s\ n\ r$$

$$n > 0 \wedge f\ z\ r \wedge Repeat\ f\ s\ (n-1)\ z \Rightarrow Repeat\ f\ s\ n\ r$$

$$v = u + 1 \Rightarrow Succ\ u\ v$$

P

$$Repeat\ Succ\ 0\ n\ r \Rightarrow r \geq n$$

The (safety) model checking problem for higher-order recursion schemes: $[\![\mathcal{G}]\!] \in L(\mathcal{A})$

$$S = F\,c$$
$$F x = a\,x\,(F\,(b\,x))$$

$$\begin{aligned} \delta(q_0, a) &= q_1,\ q_0 \\ \delta(q_1, b) &= q_1 \\ \delta(q_1, c) &= \epsilon \end{aligned}$$

In the quantifier free language of an *2*-state automaton, interpreted by the theory of $\mathcal{A}$:

$$Th(\mathsf{Trees},\ a,\ b,\ c,\ Q_0,\ Q_1) \text{ where } Q_i = \{t \in \mathsf{Trees} \mid t \in L(\mathcal{A}, q_i)\}$$

$$R_F\,c\,x \Rightarrow R_S\,x$$
$$y = a\,x\,z \wedge R_F\,(b\,x)\,z \Rightarrow R_F\,x\,y$$

P

$$R_S\,x \Rightarrow Q_0\,x$$

III. Symbolic Models of Higher-Type

$$n \leq 0 \land r = s \Rightarrow Repeat\ f\ s\ n\ r$$

$$n > 0 \land f\ z\ r \land Repeat\ f\ s\ (n-1)\ z \Rightarrow Repeat\ f\ s\ n\ r$$

$$v = u + 1 \Rightarrow Succ\ u\ v$$

$$Repeat\ Succ\ 0\ n\ r \Rightarrow r \geq n$$

$$\boldsymbol{Succ \mapsto \lambda uv.\,(v = u + 1)}$$

$$\boldsymbol{Repeat \mapsto \lambda f s n r.\,((\forall xy.\ f\ x\ y \Rightarrow y \geq x + 1) \land s \geq 0 \Rightarrow r \geq n)}$$

$$Repeat \mapsto \lambda f s n r.\, (\forall x y.\ f\ x\ y \Rightarrow y = x + 1) \wedge s \geq 0 \Rightarrow r \geq n$$

$$\boldsymbol{Repeat : (x{:}int \to y{:}int \to y \geq x{+}1) \to s{:}int \to n{:}int \to r{:}int \to (v \geq 0 \Rightarrow r \geq n)}$$

Syntax:

$$\boxed{\Delta \vdash T :: \sigma}$$

iff

$$\frac{\phi \in Fm(\Delta)}{\Delta \vdash \phi :: bool}$$

$$\frac{\Delta \vdash S :: \sigma_1 \qquad \Delta, x : \sigma_1 \vdash T :: \sigma_2}{\Delta \vdash x{:}S \to T :: \sigma_1 \to \sigma_2}$$

$$\frac{\Delta \vdash T_1 :: \sigma \quad \cdots \quad \Delta \vdash T_n :: \sigma}{\Delta \vdash \bigwedge_{i=1}^{n} T_i :: \sigma}$$

Semantics:

$$[\![x{:}int \to y{:}int \to y \geq x + 1]\!]^{Th(\mathbb{Z})}$$
$$= \{\rho \in \mathbb{Z} \Rightarrow \mathbb{Z} \Rightarrow \mathbb{B} \mid \forall m n.\, \rho(n)(m) \text{ implies } n \geq m + 1\}$$

Syntax:

$$\Gamma \vdash s : T$$

iff

$$\frac{}{\Gamma, x : T \vdash x : T}$$

$$\frac{s \in Fm(\Gamma)}{\Gamma \vdash s : s}$$

$$\frac{\Gamma \vdash s : x{:}S \rightarrow T \qquad \Gamma \vdash t : S}{\Gamma \vdash s\, t : T[t/x]}$$

$$\frac{\Gamma \vdash s : \phi \qquad \Gamma \vdash t : \psi}{\Gamma \vdash s \wedge t : \phi \wedge \psi}$$

$$\frac{\Gamma \vdash s : T_1 \qquad \Gamma \vdash s : T_2}{\Gamma \vdash s : T_1 \wedge T_2}$$

$$\frac{\Gamma \vdash s : T_1 \qquad T_1 \leq T_2}{\Gamma \vdash s : T_2}$$

Semantics:

$$\Gamma \models s : T$$

iff

$$A, \alpha \models Th, \Gamma$$

implies

$$A, \alpha \models s : T$$

$$A \models Th$$

and

$$A, \alpha \models \Gamma$$

$$[\![s]\!]^A_\alpha \in [\![T]\!]^A_\alpha$$

$$\forall x.\ \alpha(x) \in [\![\Gamma]\!]^A(x)$$

Syntax:

$$\vdash P : \Gamma$$

iff

For all *R,* if:

$$R\,x_1\ \cdots x_k \Leftarrow t \in P$$

and

$$R : x_1{:}S_1 \rightarrow \cdots x_k{:}S_k \rightarrow \phi \in \Gamma$$

then the following is provable:

$$\Gamma, x_1 : S_1, \ldots, x_k : S_k \vdash t : \phi$$

Semantics:

$$\models P : \Gamma$$

iff

For all models *A* of *Th*:

$$\forall x.\ [\![P]\!]^A(x) \in [\![\Gamma]\!]^A(x)$$

(iff

$$A,\ [\![P]\!]^A \models \Gamma$$

)

Soundness:

$\vdash P : \Gamma$ implies $\models P : \Gamma$

$\Gamma \vdash s : T$ implies $\Gamma \models s : T$

Writing $[\![P, s]\!]^A_\theta$ for $[\![s]\!]^A_{([\![P]\!]^A \cup \theta)}$:

$\vdash P : \Gamma$ and $\Gamma \cup \Delta \vdash s : \phi$

implies, for all *A*:Th and θ:Δ :

$$[\![P, s]\!]^A_\theta \in [\![\phi]\!]^A_\theta$$

($\vdash \mathcal{G} : \Gamma$ and $\Gamma \vdash s : q$

implies

$[\![\mathcal{G}, s]\!]^{\text{Trees}} \in [\![q]\!]^{\text{Trees}}$)

$\vdash P : \Gamma$ and $\Gamma \cup \Delta \vdash s : \phi$

implies for all *A*:Th, there is a valuation α:

$$A, \alpha \models P \wedge \forall \Delta .\, s \Rightarrow \phi$$

$$n \leq 0 \land r = s \Rightarrow Repeat\ f\ s\ n\ r$$

$$n > 0 \land f\ z\ r \land Repeat\ f\ s\ (n-1)\ z \Rightarrow Repeat\ f\ s\ n\ r$$

$$v = u + 1 \Rightarrow Succ\ u\ v$$

P

$$Repeat\ Succ\ 0\ n\ r \Rightarrow r \geq n$$

Find

$$\Gamma$$

such that

$\vdash P : \Gamma$ and

$$\Gamma, n : int, r : int \vdash Repeat\ Succ\ 0\ n\ r : r \geq n$$

IV. Algorithms

- Dependent (refinement) type inference for functional programs
  - Bakst, Jhala, Kawaguchi, Rondon, Seidel, Vazou…
  - Hashimoto, Kobayashi, Sato, Terauchi, Unno…
- Reduction to first-order horn clauses via a dependent type system
  - Jhala, Majumdar and Rybalchencko CAV’11
- Extension of technology from HORS model checking
  - TRecS/Lazy Annotation (Revisited) crossover

$$n \leq 0 \wedge r = s \Rightarrow Repeat_2\ f\ s\ n\ r$$

$$n > 0 \wedge f\ z\ r \wedge Repeat_1\ f\ s\ (n-1)\ z \Rightarrow Repeat_2\ f\ s\ n\ r$$

$$n \leq 0 \wedge r = s \Rightarrow Repeat_1\ f\ s\ n\ r$$

$$n > 0 \wedge f\ z\ r \wedge Repeat_0\ f\ s\ (n-1)\ z \Rightarrow Repeat_1\ f\ s\ n\ r$$

$$false \Rightarrow Repeat_0\ f\ s\ n\ r$$

$$v = u + 1 \Rightarrow Succ\ u\ v$$

$$Repeat_2\ Succ\ 0\ n\ r \wedge r < n$$

$$n > 0 \wedge Succ\ z\ r \wedge Repeat_1\ Succ\ 0\ (n-1)\ z \wedge r < n$$

$$n = 0 \wedge r = 0 \wedge r < n$$

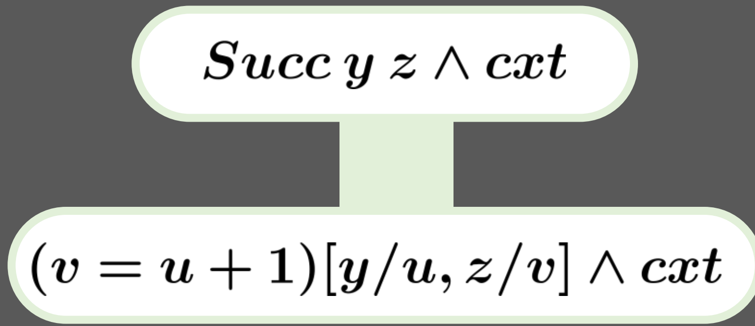
$$n > 0 \wedge r = z + 1 \wedge Repeat_1\ Succ\ 0\ (n-1)\ z \wedge r < n$$

$$n > 0 \wedge r = z + 1 \wedge n - 1 = 0 \wedge z = 0 \wedge r < n$$

$$n > 0 \wedge r = z + 1 \wedge Succ\ y\ z \wedge Repeat_0\ Succ\ 0\ (n-2)\ y \wedge r < n$$

$$n > 0 \wedge r = z + 1 \wedge Succ\ y\ z \wedge false \wedge r < n$$

$$n > 0 \wedge r = z + 1 \wedge z = y + 1 \wedge false \wedge r < n$$



$$\llparenthesis (v = u + 1)[y/u, z/v] \rrparenthesis_{\Gamma} \;\wedge\; \llparenthesis cxt \rrparenthesis_{\Gamma}$$ unsat

$$\llparenthesis (v = u + 1) \rrparenthesis_{\Gamma}[y/u, z/v] \;\wedge\; \llparenthesis cxt \rrparenthesis_{\Gamma}$$ unsat

$$\llparenthesis (v = u + 1) \rrparenthesis_{\Gamma} \;\wedge\; \llparenthesis cxt \rrparenthesis_{\Gamma} \wedge y = u \wedge z = v$$ unsat

Interpolant($\llparenthesis (v = u + 1) \rrparenthesis_{\Gamma}$ / $\llparenthesis cxt \rrparenthesis_{\Gamma} \wedge y = u \wedge z = v$)

$$v \geq u + 1$$

$$\boldsymbol{Succ\; y\; x : (v \geq u + 1)[x/v, y/u]}$$

$$\boldsymbol{Succ\; y : v{:}int \rightarrow (v \geq u + 1)[y/u]}$$

$$\boldsymbol{Succ : u{:}int \rightarrow v{:}int \rightarrow v \geq u + 1}$$

$$\Gamma$$

χ = $s \geq 0 \Rightarrow r \geq n$ = Interpolant (... / ...)

$$Repeat_1\ Succ\ 0\ (n-1)\ z : \chi[z/r, (n-1)/n, 0/s]$$

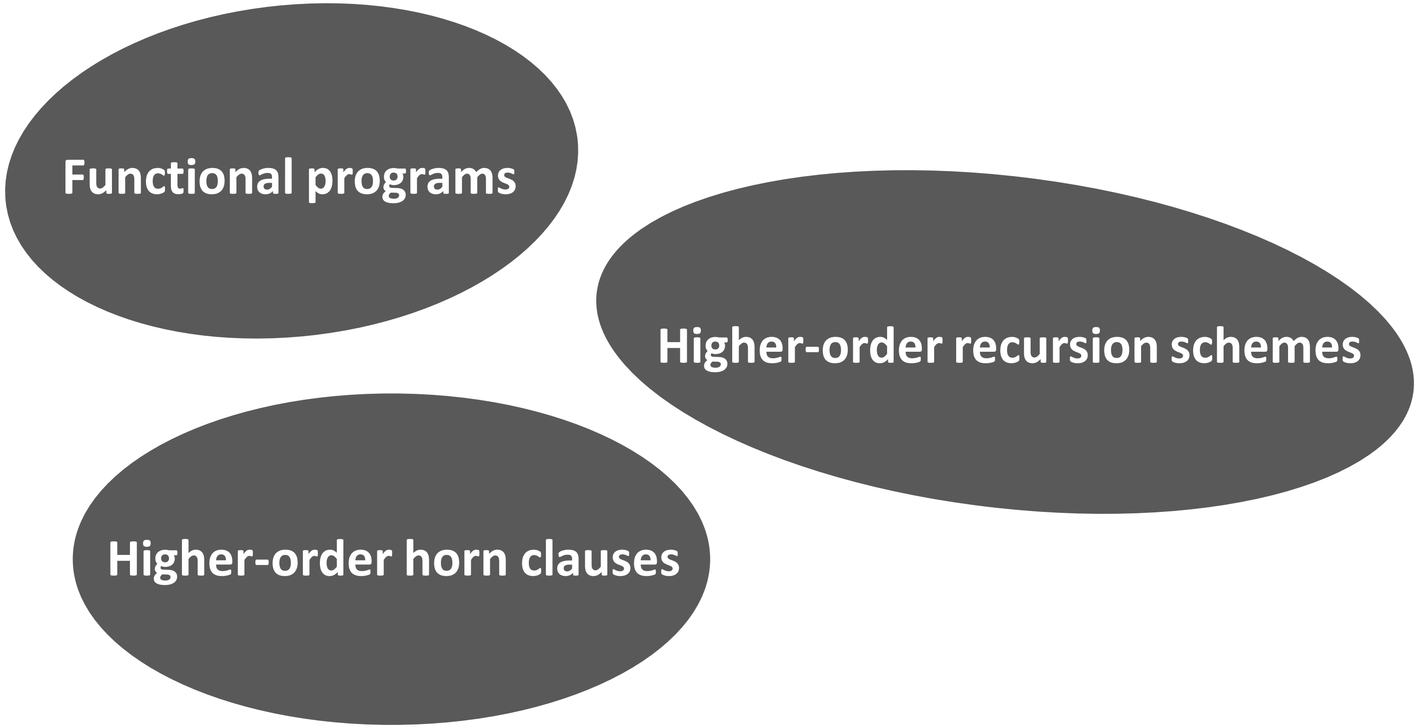
$$Repeat_1\ Succ\ 0\ (n-1) : r{:}\iota \rightarrow \chi[(n-1)/n, 0/s]$$

$$Repeat_1\ Succ\ 0 : n{:}\iota \rightarrow r{:}\iota \rightarrow \chi[0/s]$$

$$Repeat_1\ Succ : s{:}\iota \rightarrow n{:}\iota \rightarrow r{:}\iota \rightarrow \chi$$

$$Repeat_1 : (u{:}\iota \rightarrow v{:}\iota \rightarrow v \geq u+1) \rightarrow s{:}\iota \rightarrow n{:}\iota \rightarrow r{:}\iota \rightarrow \chi$$

$$\Gamma$$



- Higher-order constraint problems
- Technology transfer from HORS model checking to HOHC satisfiability
- Higher-order co-horn clauses (type inference)

End