HF Sets in Constructive Type Theory

Gert Smolka and Kathrin Stark

Interactive Theorem Proving, Nancy, August 24, 2016

SAARLAND UNIVERSITY
COMPUTER SCIENCE

A minimal computational axiomatization
of HF sets
with a unique model.

SAARLAND UNIVERSITY
COMPUTER SCIENCE

What are Hereditarily Finite sets?

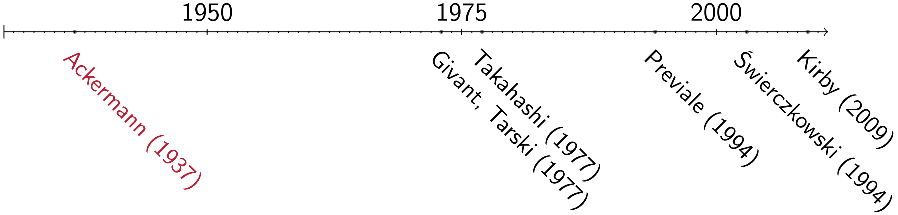
= all finite, well-founded sets whose elements are HF again

What are HF sets useful for?

Świerczkowski (1994), Paulson (2015)

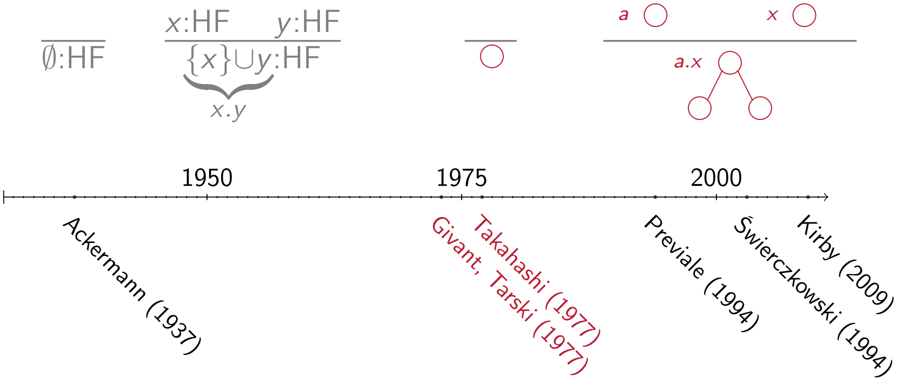
Previous Work

$hf \approx \mathbb{N}$

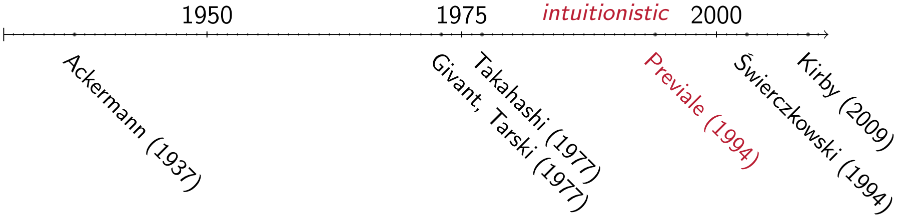


$$
\begin{array}{cccccccc}
 & & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\
21 & \approx & 1 & 0 & 1 & 0 & 1 \\
\lceil 21 \rceil & = & \{\lceil 4 \rceil, & & \lceil 2 \rceil, & & \lceil 0 \rceil\}
\end{array}
$$

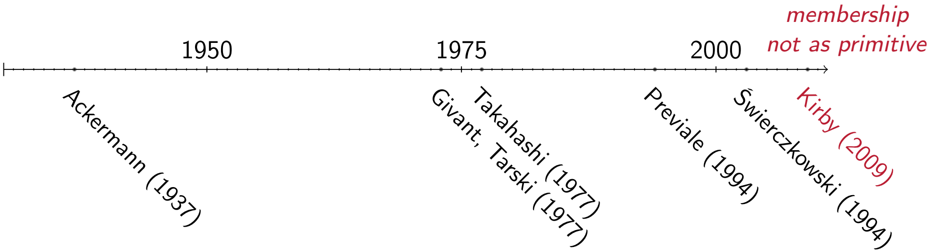
Previous Work



Previous Work



Previous Work



A minimal computational axiomatization
of HF sets
with a unique model.

What is needed for HF sets?

SAARLAND UNIVERSITY
COMPUTER SCIENCE

1. Constants: hf, $\emptyset$, $a.x$

   $x \in y := x.y = y$

2. A characterization of equality

$$\begin{aligned} x.x.y &= x.y && \textit{(cancellation)} \\ x.y.z &= y.x.z && \textit{(swap)} \\ x.y &\neq \emptyset && \textit{(discreteness)} \\ \underbrace{x.y.z = y.z \rightarrow x = y \vee x.z = z}_{x \in y.z \rightarrow x = y \,\vee\, x \in z} & && \textit{(membership)} \end{aligned}$$

3. A strong induction principle

$$\forall p : \text{hf} \rightarrow \text{Type}.\ p\ \emptyset \\ \rightarrow (\forall x\ y.\ p\ x \rightarrow p\ y \rightarrow p\ (x.y)) \rightarrow \forall x.\ p\ x$$

Working with the Induction Principle

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$R : \ p\,\emptyset \to (\forall x\, y.\ p\,x \to p\,y \to p\,(x.y)) \to \forall x.\ p\,x$$

$$R\ p_0\ p_S\ \emptyset \stackrel{?}{=} p_0$$
$$R\ p_0\ p_S\ (a.x) \stackrel{?}{=} p_S\ (R\ p_0\ p_S\ a)\ (R\ p_0\ p_S\ x)$$

$$\pi_1\ \emptyset = \text{None}$$
$$\pi_1\ (a.x) = \text{Some}\ a$$

⚡

Working with the Induction Principle

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$$R:\ p\,\emptyset \to (\forall x\, y.\ p\,x \to p\,y \to p\,(x.y)) \to \forall x.\ p\,x$$

1. Recursive Specification

$$\begin{aligned} \emptyset \quad &\cup \quad y &= \quad& y \\ a.x \quad &\cup \quad y &= \quad& a.(x \cup y) \end{aligned}$$

2. Membership Specification

$$z \in x \cup y \leftrightarrow z \in x \lor z \in y$$

1. Membership Specification

$$\Sigma u.\ \forall z.\ z \in u$$
$$\leftrightarrow z \in x \lor z \in y$$

2. Recursive Specification
   Needed: extensionality

What is not needed as primitives?

1. Membership
   $x \in y := x.y = y$
2. Recursion equations
3. Decidability of equality: dep. on extensionality
4. Extensionality: dep. on decidability of equality

Extensionality and Decidability Results

SAARLAND UNIVERSITY
COMPUTER SCIENCE

$dec\ (x \in \underline{y})$

$dec\ (y \in \underline{x})$

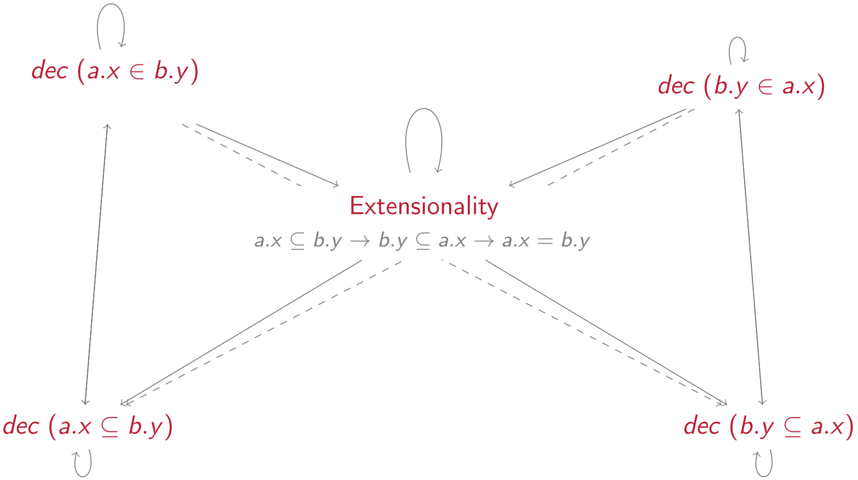
Extensionality

$$x \subseteq y \rightarrow y \subseteq x \rightarrow \underline{x} = \underline{y}$$

$dec\ (\underline{x} \subseteq y)$

$dec\ (\underline{y} \subseteq x)$

Extensionality and Decidability Results

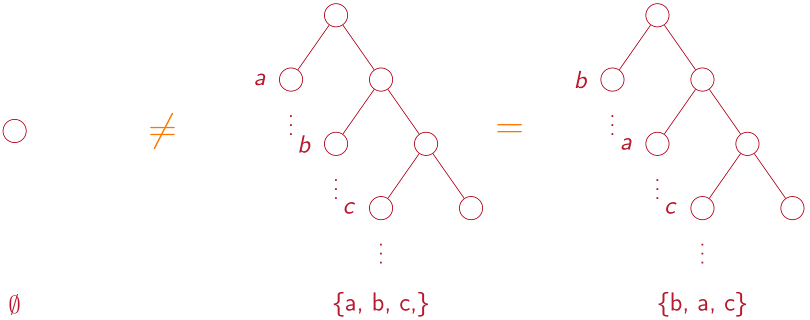


A minimal computational axiomatization
of HF sets
with a unique model.

existence uniqueness

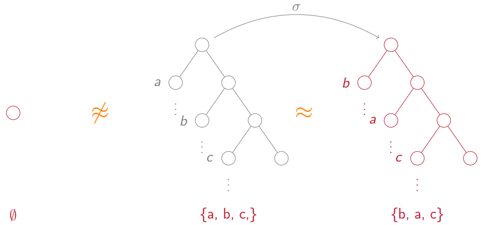
A Tree Model for HF Sets

SAARLAND UNIVERSITY
COMPUTER SCIENCE



HF sets = $\emptyset$ + $a.x$ + equality + induction principle

A Tree Model for HF Sets



1 An inductive type representing the tree structure:

$$T := 0 \mid T.T$$

2 An equivalence relation $\approx$: $T \to T \to \text{Prop}$

3 An idempotent normalizer $\sigma : T \to T$ s.t.

$$s \approx t \leftrightarrow \sigma s = \sigma t$$

4 Construct a subtype X of T only containing normalized trees.

Definition of $\approx$

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Equivalence

$$\frac{}{s.s.t \approx s.t} \qquad \frac{}{s.t.u \approx t.s.u}$$

$$\frac{}{s \approx s} \qquad \frac{s \approx t}{t \approx s} \qquad \frac{s \approx t \quad t \approx u}{s \approx u} \qquad \frac{s \approx s' \quad t \approx t'}{s.t \approx s'.t'}$$

To show: $\approx$ satisfies the equality axioms of HFs, for example

1. $s.s.t \approx s.t$
2. $s.t.u \approx t.u \rightarrow s \approx t \vee s.u \approx u$

A Normalization Function

Idea: Use sorted trees as normal form.

Lexical Tree Order

$$\frac{}{0 < s.t} \qquad \frac{s < s'}{s.t < s'.t'} \qquad \frac{t < t'}{s.t < s.t'}$$

Define a sort function $\sigma : T \to T$ according to the above order satisfying

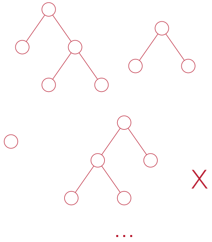
1. $\sigma\ (\sigma\ s) = \sigma\ s$
2. $s\ \approx\ t \leftrightarrow \sigma s = \sigma t$

$\Rightarrow$ There exists a type $\{t \mid \sigma t = t\}$.

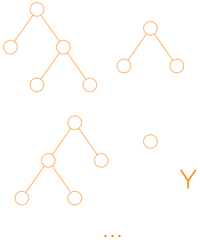
A minimal computational axiomatization
of HF sets
with a unique model.

existence uniqueness

Are all HF structures the same?



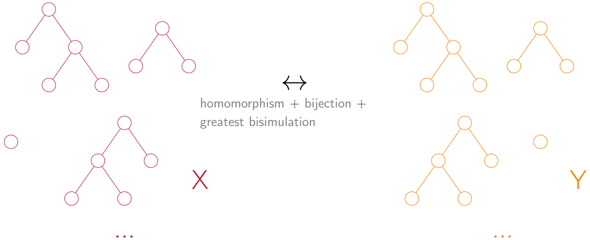
$\overset{?}{\leftrightarrow}$



$f : X \to Y$
homomorphism:

$$f\ \emptyset = \emptyset$$
$$f\ (a.x) = (f\ a).(f\ x)$$

Are all HF structures the same?



$$\frac{}{R\ \emptyset\ \emptyset} \qquad \frac{R\ a\ b \quad R\ x\ y}{R\ a.x\ b.y}$$

1. **Totality** $\forall x.\ \Sigma y.\ R\ x\ y.$
2. **Functionality** $R\ x\ y \to R\ x\ y' \to y = y'$
   - **Simulation** $R\ x\ y \to a \in x \to \exists\ b.b \in y \land R\ a\ b$
3. f homomorphism $\Rightarrow\ R\ x\ (f\ x)$
4. All homomorphisms between HF structures are equivalent.
5. All HF structures are isomorphic.

A minimal computational axiomatization
of HF sets
with a unique model.

Axiomatization + Discreteness +
Operations + Ordinals + Categoricity +
Model Construction

Everything is formalized in Coq.

$\sim$ 2000 lines

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Everything is formalized in Coq.

similar to proofs in paper

special-purpose tactic based on intro-elim rules

Everything is formalized in Coq.

no inductive types except for the model construction

Everything is formalized in Coq.

Where? - www.ps.uni-saarland.de/extras/hfs

Contribution

SAARLAND UNIVERSITY
COMPUTER SCIENCE

- First minimal, computationally complete axiomatization of HF sets
- Operationally complete axiomatization
- First proof of categoricity

Further Work

- A recursor with equations
- Axiomatization of non-wellfounded sets

Thank you for your attention!

Where? - www.ps.uni-saarland.de/extras/hfs