Formalizing Kruskal's Tree Theorem in Isabelle/HOL

Christian Sternagel

JAIST

July 5, 2012

Kickoff Meeting of Austria-Japan Joint Project
Gamagori, Japan

Kruskal's Tree Theorem

If the set A is well-quasi-ordered then the set of finite trees over A is well-quasi-ordered by homeomorphic embedding.

Comments

- proof structure as Nash-Williams 1963
- which claims

A new and simple proof is given . . .

Overview

- Motivation
- Preliminaries
- Kruskal's Tree Theorem - A Proof Sketch
- Formalization Challenges
- Conclusion

Bibliography

G. Higman.
Ordering by divisibility in abstract algebras. *Proc. London Math. Soc.*, 1952.
doi:10.1112/plms/s3-2.1.326.

J. B. Kruskal.
Well-quasi-ordering, the tree theorem, and vazsonyi's conjecture. *Trans. Amer. Math. Soc.*, 1960.
doi:10.2307/1993287.

C. S. J. A. Nash-Williams.
On well-quasi-ordering finite trees. *Math. Proc. Cambridge*, 1963.
doi:10.1017/S0305004100003844.

C. Sternagel.
Well-Quasi-Orders. In *The Archive of Formal Proofs*. 2012.
http://afp.sf.net/devel-entries/Well_Quasi_Orders.shtml.

Why?

- long-standing open problem in formalized mathematics
- Kruskal's Tree Theorem is main ingredient to prove well-foundedness of simplification orders for first-order rewriting
- ultimately, we want to strengthen termination library of IsaFoR (Isabelle Formalization of Rewriting)

> First, for constrained rewriting with forbidden patterns, we want to be able to certify the loop detection algorithm of [78] which encompasses the algorithms for loops under the innermost and outermost strategy [77, 76]. To date no formal certification techniques for these highly interesting techniques are known. This is partly due to the fact that the correctness proof uses a very powerful and complex theorem: Kruskal's tree theorem [40], whose formal verification is open.

Homeomorphic Embedding on Lists

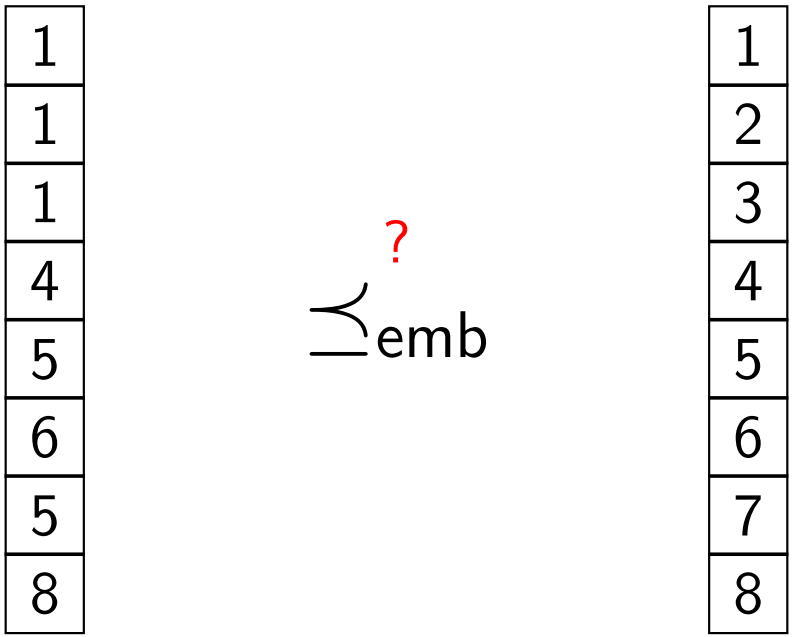
- empty list, $[]$
- adding element x to finite list xs, $x \cdot xs$
- append lists xs and ys, $xs \mathbin{@} ys$
- set of finite lists over A, A^*:

$$\frac{}{[] \in A^*} \qquad \frac{x \in A \quad xs \in A^*}{x \cdot xs \in A^*}$$

- embedding relation w.r.t. $\preceq$:

$$\frac{}{[] \preceq_{\mathsf{emb}} ys} \qquad \frac{xs \preceq_{\mathsf{emb}} ys}{xs \preceq_{\mathsf{emb}} y \cdot ys} \qquad \frac{x \preceq y \quad xs \preceq_{\mathsf{emb}} ys}{x \cdot xs \preceq_{\mathsf{emb}} y \cdot ys}$$

Example - List Embedding



We can . . .

- drop elements $\left(\dfrac{xs \trianglelefteq_{\text{emb}} ys}{xs \trianglelefteq_{\text{emb}} y \cdot ys}\right)$
- replace elements by smaller elements $\left(\dfrac{x \trianglelefteq y \quad xs \trianglelefteq_{\text{emb}} ys}{x \cdot xs \trianglelefteq_{\text{emb}} y \cdot ys}\right)$
- note that empty list is embedded in any list $([] \trianglelefteq_{\text{emb}} ys)$

Homeomorphic Embedding on Trees

- tree with node f and list of direct subtrees ts, $f(ts)$
- root of tree $\mathrm{root}(f(ts)) = f$, direct subtrees $\mathrm{args}(f(ts)) = ts$
- set of finite trees over A, $\mathcal{T}(A)$:

$$\frac{f \in A \quad \forall t \in ts.\, t \in \mathcal{T}(A)}{f(ts) \in \mathcal{T}(A)}$$

- homeomorphic embedding relation w.r.t. $\preceq$:

$$\frac{t \in ts}{t \preceq_{\mathsf{emb}} f(ts)} \qquad \frac{f \preceq g \quad ss\ (\preceq^{=}_{\mathsf{emb}})_{\mathsf{emb}}\ ts}{f(ss) \preceq_{\mathsf{emb}} g(ts)}$$

$$\frac{s \preceq_{\mathsf{emb}} t \quad t \preceq_{\mathsf{emb}} u}{s \preceq_{\mathsf{emb}} u} \qquad \frac{s \preceq_{\mathsf{emb}} t}{f(ss_1 \mathbin{@} s \cdot ss_2) \preceq_{\mathsf{emb}} f(ss_1 \mathbin{@} t \cdot ss_2)}$$

Homeomorphic Embedding on Trees (cont'd)

Embedding TRS

let $\mathcal{E}\mathsf{mb}(\preceq)$ be the infinite TRS

$$f(ts) \to t \qquad \text{if } t \in ts$$
$$f(ts) \to g(ss) \qquad \text{if } g \preceq f \text{ and } ss =_{\mathsf{emb}} ts$$

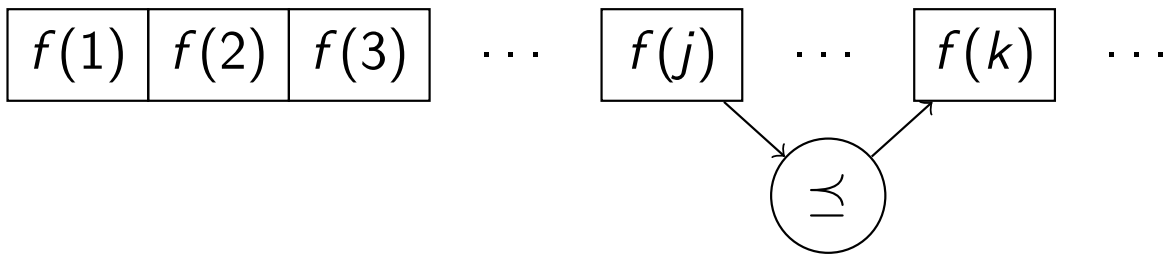
Result

$s \preceq_{\mathsf{emb}} t$ iff $t \to^{+}_{\mathcal{E}\mathsf{mb}(\preceq)} s$

Well-Quasi-Orders - Definitions

- let A be a set and $\preceq$ a binary relation
- A is wqo by $\preceq$ ($\preceq_A$ is a wqo, or wqo($\preceq_A$)):
  (1) transitive: $\forall x \in A.\, \forall y \in A.\, \forall z \in A.\, x \preceq y \wedge y \preceq z \longrightarrow x \preceq z$
  (2) *all infinite sequences over A are good*:

$$\forall f.\, (\forall i.\, f(i) \in A) \longrightarrow (\exists j\ k.\, j < k \wedge f(j) \preceq f(k))$$



- a sequence that is not good, is called bad

Property

- strict part of $\preceq$ is $x \prec y = x \preceq y \wedge y \npreceq x$
- let wqo($\preceq_A$), then $\prec_A$ is well-founded on A

Kruskal's Tree Theorem - A Proof Sketch

Proof Structure of $\mathrm{wqo}(\preceq_{\mathcal{F}}) \implies \mathrm{wqo}(\preceq_{\mathcal{T}(\mathcal{F})})$

assume $\preceq_{\mathcal{F}}$ is wqo

assume $\preceq_{\mathcal{T}(\mathcal{F})}$ is not wqo

$\implies$ exists minimal bad sequence $t_1, t_2, t_3, \ldots$ with $t_i \in \mathcal{T}(\mathcal{F})$

$\implies$ exists sequence $f_1, f_2, f_3, \ldots$ with $\mathrm{root}(t_i) = f_i$, $\mathrm{args}(t_i) = ts_i$

$\implies$ let $T = \bigcup_i ($

$\implies$ $\preceq_{\{f_i\}}$ and $\preceq_{T^*}$ are wqo

$\implies$ $\preceq_{\{f_i\} \times T^*}$ is wqo

$\implies$ exist j, k with $j < k$ and $(f_j, ts_j) \preceq_{\{f_i\} \times T^*} (f_k, ts_k)$

$\implies$ $t_j \preceq_{\mathcal{T}(\mathcal{F})} t_k$

$\implies$ $t_1, t_2, t_3, \ldots$ is good

contradiction!

prove it!

ense?

Higman's lemma

Dickson's lemma

Formalization Challenges

Existence of Minimal Bad Sequence - Nash-Williams 1963

Select an $t_1 \in \mathcal{T}(\mathcal{F})$ such that t_1 is the first term of a bad sequence of members of $\mathcal{T}(\mathcal{F})$ and t_1 is as small as possible. Then select an t_2 such that t_1, t_2 are the first two terms of a bad sequence of members of $\mathcal{T}(\mathcal{F})$ and t_2 is as small as possible [...]. Assuming the Axiom of Choice, this process yields a bad sequence t_1, t_2, t_3, ...

The Axiom of Choice in Isabelle

- $\forall x.\, \exists y.\, P\ x\ y \implies \exists f.\, \forall x.\, P\ x\ (f\ x)$

Minimal in What Sense?

- subtree relation, t is (proper) subtree of s, written $t \trianglelefteq s$ ($t \triangleleft s$), iff: $\dfrac{t \in ts}{t \triangleleft f(ts)} \quad \dfrac{s \triangleleft t \quad t \in ts}{s \triangleleft f(ts)}$
- proper subtree relation is well-founded (allowing for induction)

Auxiliary Definitions

- infinite sequence f is minimal at position n ($\min_n(f)$), iff:

$$\forall g.\, (\forall i < n.\, g(i) = f(i)) \wedge g(n) \triangleleft f(n) \wedge (\forall i \geq n.\, \exists j \geq n.\, g(i) \trianglelefteq f(j))$$

$$\implies \text{good}_{\preceq_{\text{emb}}}(g)$$

- replace elements of sequence f by those of sequence g, starting at n: $(f\langle n \rangle g)(i) =$ if $i \geq n$ then $g(i)$ else $f(i)$

Key Lemma

(1) $\min_n(f)$

(2) $\mathsf{bad}_{\preceq_{\mathsf{emb}}}(f)$

$\Longrightarrow$ $\exists g.\, \forall i \leq n.\, g(i) = f(i)$
$\wedge\ g(n+1) \trianglelefteq f(n+1)$
$\wedge\ \forall i \geq n+1.\, \exists j \geq n+1.\, g(i) \trianglelefteq f(j)$
$\wedge\ \mathsf{bad}_{\preceq_{\mathsf{emb}}}(f\langle n+1\rangle g)$
$\wedge\ \min_{n+1}(f\langle n+1\rangle g)$

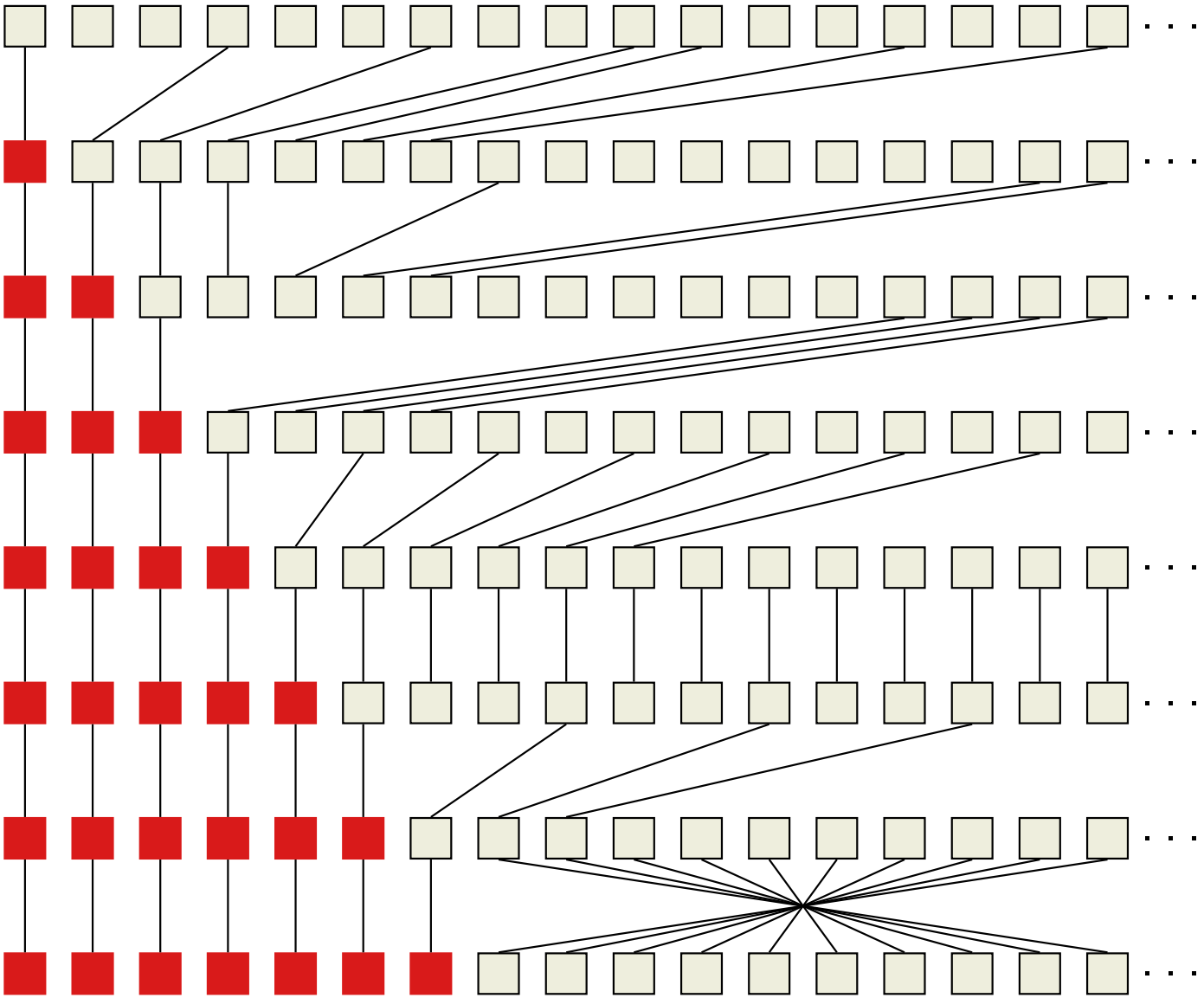
Construct Minimal Bad Sequence

- from AC and key lemma obtain function ν, s.t., given sequence satisfying (1) and (2) and index n, returns sequence satisfying conclusion
- auxiliary sequence (of sequences)

$$m'(n) = \begin{cases} \nu(f, n) & \text{if } n = 0 \\ m'(n-1)\langle n\rangle\nu(m'(n-1), n-1) & \text{otherwise} \end{cases}$$

- minimal bad sequence $m(i) = m'(i)(i)$

Idea



Conclusion

Related Work

- *Murthy, Extracting Constructive Content from Classical Proofs, PhD, 1990* (Nuprl)
- *Fridlender, Ramsey's Theorem in Type Theory, Tech. Report, 1993* (ALF)
- *Herbelin, A Program from an A-Translated Impredicative Proof of Higman's Lemma, 1994* (2-letter alphabet, Coq)
- *Fridlender, Higman's Lemma in Type Theory, PhD, 1997* (ALF)
- *Seisenberger, On the Constructive Content of Proofs, PhD, 2003* (Minlog)
- *Berghofer, A Constructive Proof of Higman's Lemma in Isabelle, TYPES 2003* (2-letter alphabet, Isabelle)
- *Martín-Mateos et al., A Formal Proof of Higman's Lemma in ACL2, JAR 2011* (ACL2)

Future Work

- investigate how Zorn's Lemma could be of help
- reformulate proof using open induction

The End