

Candidates and the Relational Action of Type Constructors

Robert Harper

January 13, 2022

1 Introduction

The termination proof given in Harper (2020) and the normalization proof given in Harper (2021) are similar in that they make use of the *action* of type constructors on *candidates (for computability)*. By defining these actions appropriately, the proof of the fundamental theorem “writes itself” in that the cases for the introduction and elimination rules for each type are automatic. This formulation and consolidation is the foundation for the method of *logical relations* used widely in semantics.

2 Hereditary Termination, Reformulated

Definition 1 (Candidate for Computability). *A closed candidate \mathcal{C} for a type A is a predicate on closed terms of type A that is closed under head expansion in that, for closed M and M' of type A , if $M \in \mathcal{C}$ and $M' \mapsto^* M$, then $M' \in \mathcal{C}$.*

Each of the type constructors has an action on closed candidates that suffices for the proof of the Fundamental Theorem.

Definition 2 (Action on Closed Candidates).

$$\begin{aligned} \mathbf{0} &\triangleq \emptyset \\ \mathbf{1} &\triangleq \{ M \mid M : A \text{ and } M \mapsto^* \star \} \\ \mathcal{C}_1 + \mathcal{C}_2 &\triangleq \{ M \mid M : A_1 + A_2 \text{ and } M \mapsto^* \mathbf{1} \cdot M_1 \text{ and } M_1 \in \mathcal{C}_1 \} \\ &\quad \cup \{ M \mid M : A_1 + A_2 \text{ and } M \mapsto^* \mathbf{2} \cdot M_2 \text{ and } M_2 \in \mathcal{C}_2 \} \\ \mathcal{C}_1 \times \mathcal{C}_2 &\triangleq \{ M \mid M : A_1 \times A_2 \text{ and } M \cdot \mathbf{1} \in \mathcal{C}_1 \text{ and } M \cdot \mathbf{2} \in \mathcal{C}_2 \} \\ \mathcal{C}_1 \rightarrow \mathcal{C}_2 &\triangleq \{ M \mid M : A_1 \rightarrow A_2 \text{ and if } M_1 \in \mathcal{C}_1 \text{ then } \mathbf{ap}(M, M_1) \in \mathcal{C}_2 \} \end{aligned}$$

Exercise 1. *Check that each of the actions results in a closed candidate, given that its arguments are closed candidates of appropriate type.*

Hereditary termination may be defined succinctly using these constructions:

$$\begin{aligned}
\text{HT}(\mathbf{0}) &\triangleq \mathbf{0} \\
\text{HT}(\mathbf{1}) &\triangleq \mathbf{1} \\
\text{HT}(A_1 + A_2) &\triangleq \text{HT}(A_1) + \text{HT}(A_2) \\
\text{HT}(A_1 \times A_2) &\triangleq \text{HT}(A_1) \times \text{HT}(A_2) \\
\text{HT}(A_1 \rightarrow A_2) &\triangleq \text{HT}(A_1) \rightarrow \text{HT}(A_2)
\end{aligned}$$

The content has not changed, only the formulation.

3 Hereditary Normalization, Reformulated

In the case of the normalization proof candidates are re-defined as *families* of sets indexed by the pre-order on contexts. Each context Δ determines a set (predicate) in such a way that if $\Delta' \leq \Delta$, then the set assigned to Δ must be contained in the set assigned to Δ' . This requirement is in line with weakening for the typing judgment: if $\Delta \vdash M : A$, then $\Delta' \vdash M : A$ for any $\Delta' \leq \Delta$. Put in other terms, the set assignment cannot depend on what variables are *not* present in the context, only on those that *are*.

Definition 3 (Open Candidate). *An open candidate, \mathcal{C} , for type A over context Δ is a set of terms $\Delta \vdash M : A$ that is closed under head expansion: if $M \in \mathcal{C}$, and $M' \mapsto_\beta M$, then $M' \in \mathcal{C}$.*

Definition 4 (Candidate Family). *A family of open candidates, \mathcal{F} , for a type A is an assignment of an open candidate $\mathcal{F}(\Delta)$ for A over Δ to each context Δ such that if $\Delta' \leq \Delta$, then $\mathcal{F}(\Delta) \subseteq \mathcal{F}(\Delta')$.*

As with closed candidates, each type constructor acts on candidate families in such a way as to ensure that the FTLR holds.

Definition 5 (Action on Candidate Families).

$$\begin{aligned}
\mathbf{0}(\Delta) &\triangleq \{ M \mid \Delta \vdash M : \mathbf{0} \text{ and } \text{norm}_\beta(M) \} \\
\mathbf{1}(\Delta) &\triangleq \{ M \mid \Delta \vdash M : \mathbf{1} \text{ and } \text{norm}_\beta(M) \} \\
(\mathcal{F}_1 + \mathcal{F}_2)(\Delta) &\triangleq \{ M \mid \Delta \vdash M : A_1 + A_2 \text{ and} \\
&\quad \text{if } M \rightarrow_\beta^* \mathbf{1} \cdot M_1 \text{ then } M_1 \in \mathcal{F}_1(\Delta) \text{ and} \\
&\quad \text{if } M \rightarrow_\beta^* \mathbf{2} \cdot M_2 \text{ then } M_2 \in \mathcal{F}_2(\Delta) \} \\
(\mathcal{F}_1 \times \mathcal{F}_2)(\Delta) &\triangleq \{ M \mid \Delta \vdash M : A_1 \times A_2 \text{ and } M \cdot \mathbf{1} \in \mathcal{F}_1(\Delta) \text{ and } M \cdot \mathbf{2} \in \mathcal{F}_2(\Delta) \} \\
(\mathcal{F}_1 \rightarrow \mathcal{F}_2)(\Delta) &\triangleq \{ M \mid \Delta \vdash M : A_1 \rightarrow A_2 \text{ and} \\
&\quad \text{if } \Delta' \leq \Delta \text{ and } M_1 \in \mathcal{F}_1(\Delta') \text{ then } \text{ap}(M, M_1) \in \mathcal{F}_2(\Delta') \}
\end{aligned}$$

Hereditary normalization may be succinctly defined using these operations in formally the same

way as before, albeit with the right-hand now being a family of open candidates in each case.

$$\begin{aligned}\text{HN}(\mathbf{0}) &\triangleq \mathbf{0} \\ \text{HN}(\mathbf{1}) &\triangleq \mathbf{1} \\ \text{HN}(A_1 + A_2) &\triangleq \text{HN}(A_1) + \text{HN}(A_2) \\ \text{HN}(A_1 \times A_2) &\triangleq \text{HN}(A_1) \times \text{HN}(A_2) \\ \text{HN}(A_1 \rightarrow A_2) &\triangleq \text{HN}(A_1) \rightarrow \text{HN}(A_2)\end{aligned}$$

4 Conclusion

These observations are the first step towards a more general theory of logical relations that accounts for more than just termination and preservation properties.

References

- Robert Harper. How to (re)invent Tait’s method. Unpublished lecture note, Spring 2020. URL <https://www.cs.cmu.edu/~rwh/courses/cht/pdfs/tait.pdf>.
- Robert Harper. Kripke-style logical relations for termination. Unpublished lecture note, Spring 2021. URL <https://www.cs.cmu.edu/~rwh/courses/cht/pdfs/kripke.pdf>.