Well-founded models in proofs of termination

Salvador Lucas

DSIC, Universitat Politècnica de València (UPV)
http://slucas.webs.upv.es/

16th International Workshop on Termination (WST 2018)
Oxford, United Kingdom

Main goal

Understanding, *analyzing*, and *automatically verifying* the termination behavior of declarative programs (e.g., Maude)

```
fmod PALINDROME is
  protecting QID .
  sorts List Pal .
  subsorts Qid < Pal < List .
  op nil : -> Pal .
  op __ : List List -> List .
  var I : Qid .
  var P : Pal .
  mb I P I : Pal .
endfm
```

```
fmod PROGRAM is
  sort S .
  ops a b : -> [S] .
  ceq a = b if a : S .
endfm
```

Challenging!

`PALINDROME` is (obviously?) *terminating*. `PROGRAM` is *not* (why?).

A theory $\mathcal{S}$ in a logic $\mathcal{L}$ is called *operationally terminating* if no infinite *well-formed* proof tree for $\mathcal{S}$ exists [LMM05].

Infinite branches are captured by using *proof jumps* $A \Uparrow B_1, \ldots, B_{n-1}, B_n$ associated to the *inference rules* $\frac{B_1 \ \cdots \ B_n \ \cdots \ B_p}{A}$ of $\mathcal{S}$ [LM14].

Use of logical models and well-founded relations

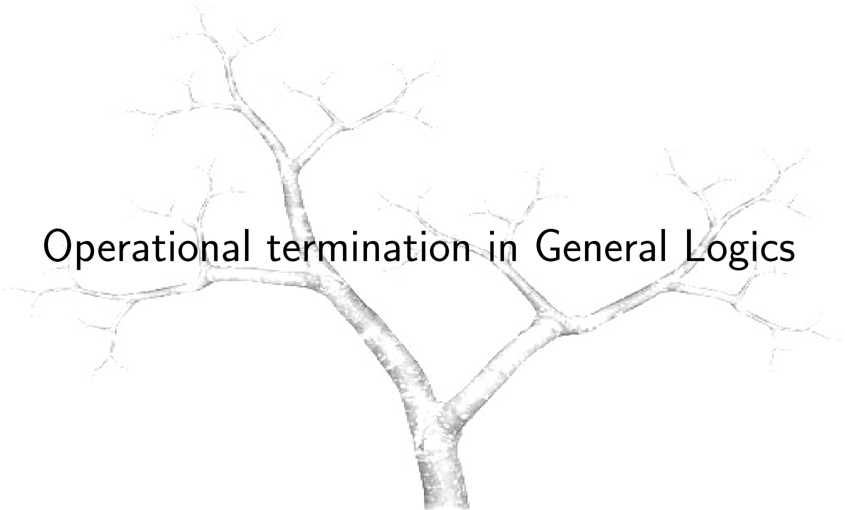
Provability of goals $\sigma(B_1), \ldots, \sigma(B_{n-1})$ captured by *logical models* when needed

The *head* A and the *hook* B_n of proof jumps indicate where *well-founded* relations $\sqsupset$ are required to prove termination: $\sigma(A) \sqsupset \sigma(B_n)$

Termination proofs = Logical models + Well-founded relations

Outline

1. Operational termination in General Logics
2. Operational termination and well-founded relations
3. Operational termination and well-founded models
4. Use in the Operational Termination Framework
5. Conclusions and future work



Operational termination in General Logics

A logical approach to declarative programming [Mes87]

1. *Declarative programs* $\mathcal{S}$ are *theories* of a given *logic* $\mathcal{L}$.
2. *Computations* with $\mathcal{S}$ are implemented as *deductions* in $\mathcal{L}$.
3. *Deductions* proceed according to the *Inference System* $\mathcal{I}(\mathcal{L})$ of $\mathcal{L}$.

Definition (Proof jumps of a theory $\mathcal{S}$ [LM14])

$$\mathcal{J}_{\mathcal{S}} = \{(A \Uparrow \vec{B}_i) \mid \frac{\vec{B}_n}{A} \in \mathcal{I}(\mathcal{S}), 1 \leq i \leq n\}$$

The specialization of the inference system for CS-MRTs [DLM+08] yields $\mathcal{I}(\texttt{PALINDROME})$ with *16 rules*.

Subject Reduction rules concern the dependency of sorts with *rewritings*

$$(SR)_Q\ \frac{x \to y \quad y : \texttt{Qid}}{x : \texttt{Qid}} \qquad (SR)_P\ \frac{x \to y \quad y : \texttt{Pal}}{x : \texttt{Pal}} \qquad (SR)_L\ \frac{x \to y \quad y : \texttt{List}}{x : \texttt{List}}$$

Membership rules describe the association of sorts to expressions by means of conditional memberships in the program.

$$(M1)_{\texttt{Q<P}}\ \frac{x :: \texttt{Qid}}{x :: \texttt{Pal}} \qquad (M1)_{\texttt{P<L}}\ \frac{x :: \texttt{Pal}}{x :: \texttt{List}} \qquad (M1)^c_Q\ \frac{}{c :: \texttt{Qid}} \text{ for } c \text{ of sort } \texttt{Qid}$$

$$(M1)_{\texttt{Nil}}\ \frac{}{\texttt{nil} :: \texttt{Pal}} \qquad (M1)_{\_\ \_}\ \frac{x :: \texttt{List} \quad y :: \texttt{List}}{x\ y :: \texttt{List}} \qquad (M1)_{\texttt{mbP}}\ \frac{I :: \texttt{Qid} \quad P :: \texttt{Pal}}{IPI :: \texttt{Pal}}$$

$$(M2)_Q\ \frac{x :: \texttt{Qid}}{x : \texttt{Qid}} \qquad (M2)_P\ \frac{x :: \texttt{Pal}}{x : \texttt{Pal}} \qquad (M2)_L\ \frac{x :: \texttt{List}}{x : \texttt{List}}$$

Reflexivity and *transitivity* rules for the many step rewrite relations.

$$(Rf)_L\ \frac{}{x \to^* x} \qquad (T)_L\ \frac{x \to y \quad y \to^* z}{x \to^* z}$$

The *congruence rules* propagate one-step reductions.

$$(C)_{\_\ \_,1}\ \frac{x \to y}{x\ z \to y\ z} \qquad (C)_{\_\ \_,2}\ \frac{x \to y}{z\ x \to z\ y}$$

Given $\frac{B_1 \cdots B_n}{A}$ with label ρ and $1 \leq i \leq n$, $[\rho]^i$ is the i-th proof jump $A \Uparrow B_1, \ldots, B_i$ obtained from ρ.

Example

The proof jumps obtained from the inference rule

$$(M1)_{\texttt{mbP}} \quad \frac{I :: \texttt{Qid} \qquad P :: \texttt{Pal}}{IPI :: \texttt{Pal}}$$

are the following:

$$\begin{array}{ll} [(M1)_{\texttt{mbP}}]^1 : & IPI :: \texttt{Pal} \Uparrow I :: \texttt{Qid} \\ [(M1)_{\texttt{mbP}}]^2 : & IPI :: \texttt{Pal} \Uparrow I :: \texttt{Qid},\ P :: \texttt{Pal} \end{array}$$

Operational termination and well-founded relations

Theorem (OT and well-founded relations)

A theory $\mathcal{S}$ is operationally terminating iff there is a well-founded relation $\sqsupset$ (on formulas) such that, for all $A \Uparrow \vec{B}_n \in \mathcal{J}_{\mathcal{S}}$,

$$\textit{for all } \sigma\textit{, if } \mathcal{S} \vdash \sigma(B_i) \textit{ for all } i,\ 1 \leq i < n\textit{, then } \sigma(A) \sqsupset \sigma(B_n) \tag{1}$$

Proof keys:

- *if part* (by contradiction): an infinite well-formed proof tree is represented by an infinite sequence of (instances of) proof jumps which (by hypothesis) lead to an infinite sequence of comparisons with $\sqsupset$ and then to a contradiction.
- *only if part*: by operational termination, a well-founded *proof progress* relation on formulas $\upuparrows$ is defined by considering all possible well-formed proof trees and the forks we can follow to go up on it. By construction, $\upuparrows$ can be taken as $\sqsupset$ to satisfy (1).

Theorem (OT and well-founded relations)

A theory $\mathcal{S}$ is operationally terminating iff there is a *well-founded relation* $\sqsupset$ *(on* *formulas**) such that, for all* $A \Uparrow \vec{B}_n \in \mathcal{J}_{\mathcal{S}}$*,*

$$\text{for all } \sigma, \text{ if } \mathcal{S} \vdash \sigma(B_i) \text{ for all } i, 1 \leq i < n, \text{ then } \sigma(A) \sqsupset \sigma(B_n)$$

How to use this result to *prove* OT? Problems:

1. infinitely many substitutions σ
2. provability statements $\mathcal{S} \vdash \sigma(B_i)$
3. symbol $\sqsupset$ does *not* belong to the language of $\mathcal{S}$

Idea

Encode as a *first-order satisfiability* problem

Operational termination and well-founded models

In the following result, $\mathcal{S}$ is a first-order theory, and

- $\overline{\mathcal{S}}$ is obtained from $\mathcal{I}(\mathcal{S})$ by interpreting inference rules $\frac{B_1 \cdots B_n}{A}$ as first-order sentences $(\forall\vec{x})\, B_1 \wedge \cdots \wedge B_n \Rightarrow A$.
- $\_^{\downarrow}$ is a transformation $P(t_1,\ldots,t_n)^{\downarrow} = f_P(t_1,\ldots,t_n)$ from atoms $P(t_1,\ldots,t_n)$ into terms $f_P(t_1,\ldots,t_n)$ of a *new sort* s_τ, where $f_P : w \to s_\tau$ are new function symbols for each *predicate* $P : w$.
- $\pi_{\sqsupset} : s_\tau s_\tau$ is a new binary (infix) predicate accepting terms of sort s_τ.

Theorem (OT and well-founded models)

A theory $\mathcal{S}$ is operationally terminating if and only if there is an interpretation $\mathcal{A}$ with no empty domain such that

$$\mathcal{A} \models \overline{\mathcal{S}} \cup \{B_1 \wedge \cdots \wedge B_{n-1} \Rightarrow A^{\downarrow}\, \pi_{\sqsupset}\, B_n^{\downarrow} \mid A \Uparrow \vec{B}_n \in \mathcal{J}_{\mathcal{S}}\}$$

and $\pi_{\sqsupset}^{\mathcal{A}}$ is well-founded on $\mathcal{A}_{s_\tau}$.

Example (Operational termination of PALINDROME)

We use S instead of Qid to make some symbols (a and b) *available* for sequencing. With AGES [GLR16], a model $\mathcal{A}$ is obtained [LG18]:

$$\mathcal{A}_{\mathtt{S}} = \mathcal{A}_{\mathtt{Pal}} = \mathbb{N} - \{0\} \quad \mathcal{A}_{\mathtt{List}} = \mathbb{N} \quad \mathcal{A}_{s_\tau} = \mathbb{N} \cup \{-1\}$$

Function symbols are interpreted as follows:

$$\begin{array}{ccc}
\mathtt{a}^{\mathcal{A}} = \mathtt{b}^{\mathcal{A}} = 1 & \mathtt{nil}^{\mathcal{A}} = 1 & x_\,_^{\mathcal{A}} y = x + y + 1 \\
f^{\mathcal{A}}_{_:\mathtt{S}}(x) = 3x & f^{\mathcal{A}}_{_:\mathtt{Pal}}(x) = 2x + 1 & f^{\mathcal{A}}_{_:\mathtt{List}}(x) = 2x + 2 \\
f^{\mathcal{A}}_{_::\mathtt{S}}(x) = -1 & f^{\mathcal{A}}_{_::\mathtt{Pal}}(x) = 2x & f^{\mathcal{A}}_{_::\mathtt{List}}(x) = 2x + 1 \\
f^{\mathcal{A}}_{\to}(x,y) = 2x - 1 & f^{\mathcal{A}}_{\to^*}(x,y) = 4x + y + 1 &
\end{array}$$

Finally, predicates are interpreted as follows:

$$\begin{array}{ccc}
_ : \mathtt{S}^{\mathcal{A}}(x) \Leftrightarrow \mathit{true} & _ : \mathtt{Pal}^{\mathcal{A}}(x) \Leftrightarrow \mathit{true} & _ : \mathtt{List}^{\mathcal{A}}(x) \Leftrightarrow \mathit{true} \\
_ :: \mathtt{S}^{\mathcal{A}}(x) \Leftrightarrow x \geq 1 & _ :: \mathtt{Pal}^{\mathcal{A}}(x) \Leftrightarrow x \geq 1 & _ :: \mathtt{List}^{\mathcal{A}}(x) \Leftrightarrow x \geq 1 \\
x \to^{\mathcal{A}} y \Leftrightarrow x > y & x(\to^*)^{\mathcal{A}} y \Leftrightarrow \mathit{true} & x\, \pi^{\mathcal{A}}_{\sqsupset}\, y \Leftrightarrow x > y
\end{array}$$

Use in the Operational Termination Framework

A *removal pair* $(\gtrsim, \sqsupset)$, consists of binary relations $\gtrsim$ and $\sqsupset$ on formulas such that $\sqsupset$ is well-founded and $\gtrsim \circ \sqsupset \subseteq \sqsupset$ (or $\sqsupset \circ \gtrsim \subseteq \sqsupset$).

Definition (Removal pair processor [LM14])

$\mathsf{P}_{RP}(\mathcal{S}, \mathcal{J}) = \{(\mathcal{S}, \mathcal{J} - \{\psi\})\}$ iff

1. for all $C \Uparrow \vec{D}_m \in \mathcal{J} - \{\psi\}$ and substitutions σ, if $\mathcal{S} \vdash \sigma(D_i)$ for all $1 \leq i < m$, then $\sigma(C) \gtrsim \sigma(D_m)$ or $\sigma(C) \sqsupset \sigma(D_m)$, and
2. for all substitutions σ, if $\mathcal{S} \vdash \sigma(B_i)$ *for all* $1 \leq i < n$, then $\sigma(A) \sqsupset \sigma(B_n)$.

Definition (Semantic version of P_{RP} [Luc16])

$\mathsf{P}_{RP}(\mathcal{S},\mathcal{J}) = \{(\mathcal{S},\mathcal{J}-\{\psi\})\}$ if

1. $\mathcal{A} \models \overline{\mathcal{S}}$,
2. if $\mathcal{J}-\{\psi\} \neq \varnothing$, then $\mathcal{A} \models x\,\pi_{\gtrsim}\,y \wedge y\,\pi_{\sqsupset}\,z \Rightarrow x\,\pi_{\sqsupset}\,z$,
3. for each $C \Uparrow \vec{D}_m \in \mathcal{J}-\{\psi\}$, there is $\pi_{\bowtie} \in \{\pi_{\gtrsim},\pi_{\sqsupset}\}$ such that $\mathcal{A} \models \bigwedge_{i=1}^{m-1} D_i \Rightarrow C^{\downarrow}\,\pi_{\bowtie}\,D_m^{\downarrow}$, and
4. $\pi_{\sqsupset}^{\mathcal{A}}$ is well-founded and $\mathcal{A} \models \bigwedge_{i=1}^{n-1} B_i \Rightarrow A^{\downarrow}\,\pi_{\sqsupset}\,B_n^{\downarrow}$

Example (Use of the semantic version of P_{RP})

Consider the CTRS $\mathcal{R} = \{\mathsf{a} \to \mathsf{b}, \mathsf{f}(\mathsf{a}) \to \mathsf{b}, \mathsf{g}(x) \to \mathsf{g}(\mathsf{a}) \Leftarrow \mathsf{f}(x) \to x\}$ in [GA01, page 46], where $\mathcal{I}(\mathcal{R})$ is:

$$(Rf)\ \frac{}{x \to^* x} \quad (T)\ \frac{x \to y \qquad y \to^* z}{x \to^* z} \quad (C)_{\mathsf{f},1}\ \frac{x \to y}{\mathsf{f}(x) \to \mathsf{f}(y)} \quad (C)_{\mathsf{g},1}\ \frac{x \to y}{\mathsf{g}(x) \to \mathsf{g}(y)}$$

$$(Rl)_1\ \frac{}{\mathsf{a} \to \mathsf{b}} \quad (Rl)_2\ \frac{}{\mathsf{f}(\mathsf{a}) \to \mathsf{b}} \quad (Rl)_3\ \frac{\mathsf{f}(x) \to^* x}{\mathsf{g}(x) \to \mathsf{g}(\mathsf{a})}$$

We can remove $[(T)]^2$ from the initial OT problem $\tau_I = (\mathcal{R}, \mathcal{J}_{\mathcal{R}})$. A model $\mathcal{A}$ with finite domain $\mathcal{A} = \{0, 1, 2, 3\}$ is found with Mace4 [McC10]:

$$\begin{array}{ccc}
\mathsf{a}^{\mathcal{A}} = 0 & \mathsf{b}^{\mathcal{A}} = 1 & \mathsf{f}^{\mathcal{A}}(x) = (x+2) \mod 4 \\
\mathsf{g}^{\mathcal{A}}(x) = x \mod 2 & f_{\to}^{\mathcal{A}}(x,y) = x \mod 2 & f_{\to^*}^{\mathcal{A}}(x,y) = x \mod 2
\end{array}$$

$$\begin{array}{ll}
\to^{\mathcal{A}} = \{(0,1),(0,3),(2,1),(2,3)\} & (\to^*)^{\mathcal{A}} = \{(x,x) \mid x \in \mathcal{A}\} \cup \to^{\mathcal{A}} \\
\pi_{\sqsupset}^{\mathcal{A}} = \{(0,1)\} & (\pi_{\gtrsim})^{\mathcal{A}} = \{(0,0),(1,1)\} \\
(\pi_{\sqsupset}^{+})^{\mathcal{A}} = \{(0,1)\} &
\end{array}$$

Successive applications of P_{RP} prove operational termination of $\mathcal{R}$.

Conclusions and future work

Our approach [LM14]:

1. *Declarative programs* P are *theories* of a given *logic* $\mathcal{L}$
2. *Operational termination* characterizes the *termination behavior* of P as the absence of *infinite proof trees* for P (using $\mathcal{I}(\mathcal{L})$)
3. *Proof jumps* capture infinite proof trees in computations.
4. The *OT Framework* provides a practical approach to prove or disprove finiteness of OT problems

Termination proofs via logical models and well-founded relations

Logic provides the expressive framework to describe the *language semantics* which is captured by *logical models* when needed

Proof jumps indicate where *well-founded* relations are required to prove termination.

Termination proofs = Logical models + Well-founded relations

In the early literature on termination of (variants of) *Term Rewriting Systems*, well-founded orderings have been pervasive...

...often as a *particular* requirement *together with* those having to do with the specific *variant* at stake:

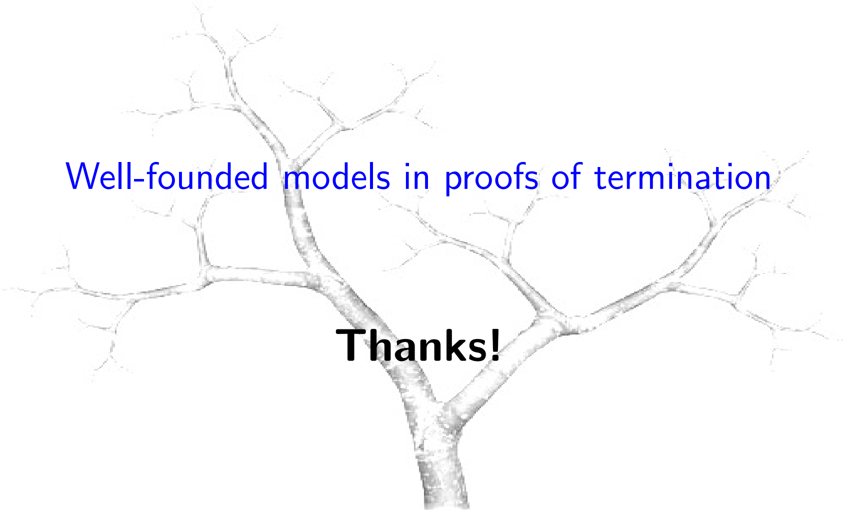
Variant	Name	Specific requirements
TRS	reduction ordering	monotonicity, stability
AC-TRS	AC-reduction ordering	compatibility with AC axioms
CS-TRS	μ-reduction ordering	μ-monotonicity, stability
CTRS	quasi-decreasingness	subterm property
CRMS	reductivity	monotonicity, stability, use of $\rhd$
$\vdots$	$\vdots$	$\vdots$

A unified treatment of rewriting-based systems

All these specific requirements are *naturally* obtained from the *logical structure* of the programming language semantics.

Challenges

1. More specific techniques for proving operational termination (e.g., dependency pairs) could be formalized/integrated to be used in the OT Framework by using *transformations* between different logics
2. The *modular structure of programs* regarding proofs of operational termination is also worth to be considered in this setting
3. Definition of *proof strategies* to improve the efficiency of the proof methodology
4. *Implementation* of the OT framework through a combined effort to improve the tools MTT [DLM08] and MU-TERM [AGLN10]



Well-founded models in proofs of termination

Thanks!

B. Alarcón, R. Gutiérrez, S. Lucas, R. Navarro-Marset.
Proving Termination Properties with MU-TERM. In *Proc. of AMAST'10*, LNCS 6486:201-208, 2011.

F. Durán, S. Lucas, C. Marché, J. Meseguer, X. Urbain,
Proving Operational Termination of Membership Equational Programs, Higher-Order and Symbolic Computation 21(1-2):59–88, 2008.

F. Durán, S. Lucas, and J. Meseguer.
MTT: The Maude Termination Tool (System Description). In *Proc. of IJCAR'08*, LNAI 5195:313–319, 2008.

J. Giesl and T. Arts.
Verification of Erlang Processes by Dependency Pairs.
Applicable Algebra in Engineering, Communication and Computing 12:39–72, 2001.

R. Gutiérrez, S. Lucas, and P. Reinoso.
A tool for the automatic generation of logical models of order-sorted first-order theories. In *Proc. of PROLE'16*, pages 215-230, 2016. Tool available at `http://zenon.dsic.upv.es/ages/`.

S. Lucas.
Use Of Logical Models For Proving Operational Termination In General Logics. In *Selected papers from WRLA'16*, LNCS 9942:1-21, 2016.

S. Lucas and R. Gutiérrez.
Automatic Synthesis of Logical Models for Order-Sorted First-Order Theories.
Journal of Automated Reasoning 60(4):465–501, 2018.

S. Lucas, C. Marché, and J. Meseguer.
Operational termination of conditional term rewriting systems. *Information Processing Letters*, 95:446–453, 2005.

S. Lucas and J. Meseguer.
Proving Operational Termination Of Declarative Programs In General Logics. In *Proc. of PPDP'14*, pages 111-122, ACM Press, 2014.

W. McCune
Prover9 & Mace4.
`http://www.cs.unm.edu/~mccune/prover9/`, 2005–2010.

J. Meseguer.
General Logics. In *Logic Colloquium'87*, pages 275-329, North-Holland, 1989.