Constructive Formalization of Regular Languages

Jan-Oliver Kaiser

Advisors: Christian Doczkal, Gert Smolka
Supervisor: Gert Smolka

Contents

Motivation

- Strong interest in formalizations in this area.
- No complete and elegant formalization of regular languages in Coq.
- Recent formalizations avoid FA in favor of partial derivatives.

Previous work

- Constructively formalizing automata theory (2000)

  Robert L. Constable, Paul B. Jackson, Pavel Naumov, Juan C. Uribe

  PA: Nuprl

  The first constructive formalization of MH.

  Based on **FA**.

- Proof Pearl: Regular Expression Equivalence and Relation Algebra (2011)

  Alexander Krauss, Tobias Nipkow

  PA: Isabelle

  Based on **partial derivatives of RE**. No proof of termination.

- Deciding Kleene Algebras in Coq (2011)

  Thomas Braibant, Damien Pous

  PA: Coq

  Based on **FA**, matrices. Focus on performance.

- A Decision Procedure for Regular Expression Equivalence in Type Theory (2011)

  Thierry Coquand, Vincent Siles

  PA: Coq

  Based on **partial derivatives of RE**.

- A Formalisation of the Myhill-Nerode Theorem based on Regular Expressions (Proof Pearl) (2011)

  Chunhan Wu, Xingyuan Zhang, Christian Urban

  PA: Isabelle

  The first proof of MH based on **partial derivatives of RE**.

- Deciding Regular Expressions (In-)Equivalence in Coq (2011)

  Nelma Moreira, David Pereira, Simão Melo de Sousa

  PA: Coq

  Based on Krauss, Nipkow. Proof of termination.

Our Development

- We want to focus on elegance, not performance.
- Our main goals are MH and the decidability of RE equivalence.
- We use FA.
  They are not at all impractical. (Partly thanks to Ssreflect's finType)

Ssreflect

- Excellent support for all things boolean.
- Finite types with all necessary operations and closure properties.
  (very useful for alphabets, FA states, etc.)
- Lots and lots of useful lemmas and functions.

Finite automata

DFA and NFA without e-transitions.

- DFA to prove closure under $\cup$, $\cap$, and $\neg$.
- NFA to prove closure under $\bullet$ and $*$.

Also proven: NFA $\Leftrightarrow$ DFA.

This gives us: RE $\Rightarrow$ FA.

Roadmap

1. Emptiness test on FA ($\emptyset(A) := \mathcal{L}(A) = \emptyset$)
2. FA $\Rightarrow$ RE
3. Dedicedability of RE equivalence using RE $\Rightarrow$FA, (2) and (1):

$$\mathcal{L}(r) = \mathcal{L}(s)$$

$$\Leftrightarrow$$

$$\emptyset(\mathcal{A}(r) \cap \overline{\mathcal{A}(s)}) \wedge \emptyset(\overline{\mathcal{A}(r)} \cap \mathcal{A}(s))$$

4. Finally, we want to prove the MH theorem

With this we'll have a complete formalization of regular languages including RE, FA and MH and all corresponding equivalences.