Master 2 internship proposal

Title: Deciding rational functions among regular ones using streaming string transducers

Location: Laboratoire d'Informatique Fondamentale de Marseille
`http://www.lif.univ-mrs.fr`

Supervisor:
Pierre-Alain Reynier
Tel: 04 91 82 93 13
Web: `http://www.lif.univ-mrs.fr/~preynier`
Email: `pierre-alain.reynier@lif.univ-mrs.fr`

This internship may lead to a PhD thesis.

Context.

Verification of software is a major challenge of computer science. Model-checking techniques, which heavily rely on automata-logic connections, have proven to be successful.

In the context of data processing, transducers are an appealing model for describing transformations. Automata-logic connections yield efficient tools, based on automata, and rich specification formalisms, based on logic. While in the context of words, rational languages (recognised by finite-state automata) and regular languages (defined by MSO formula) coincide, this is no longer the case in the setting of transformations. Indeed, *regular word-to-word transformations* (i.e. MSO-definable transformations) correspond to those definable using two-way finite state transducers [5]. Transformations defined by one-way finite state automata, called *rational transformations*, form a strict subclass.

An interesting particular case is when the relation defined by the transducer is actually a function (we say it is functional). This setting is indeed common in practice. Again, rational functions form a strict subclass of regular functions. While, in the setting of languages of words, every finite-state automaton can be determinized, this does not hold anymore for one-way functional transducers: the class of functions realized by deterministic transducers, called *sequential functions*, forms a strict subclass of that of rational functions.

Recently, a new model has been proposed to describe word-to-word (functional) transformations, called *Streaming String Transducers* [2] (SST for short). It has been shown in [1] that its expressive power coincides with the class of regular functions (i.e. regular functional transformations). In addition, restrictions exist which allow to recover the two classes of rational and of sequential functions [3].

In terms of applications, this model has given rise to a programming language allowing to describe any regular word-to-word transformations and to evaluate them efficiently [4].

In the context of stream processing, it is important to be able to evaluate a transformation "online", i.e. keeping in memory only a small (i.e. bounded) part

of the input word. This is actually not always possible (consider for instance the reverse transformation). An obvious necessary condition is to be realizable by a *one-way* finite-state transducer. This motivates the study of the following problem: given a regular function, is it rational?

Objectives

The previous problem has recently been shown to be decidable [6]. The proof considers as input regular functions described by means of deterministic two-way transducers, and determines whether the two-way-ness can be removed. The drawback of this approach is that it yields a non-elementary complexity.

The main objective of this internship (which is ambitious) is to come up with an elementary decision procedure for this problem. The non-elementary complexity comes from the fact that the procedure is iterative. The idea is to use the model of SST (both as input and output) in order to come up with a more direct proof. As preliminary objectives, one can consider subclasses of SST, by restricting the number of variables and/or the update functions. Another direction is to study the decision of the class of sequential functions among regular ones.

References

[1] R. Alur and P. Černý. Expressiveness of streaming string transducers. In *FSTTCS*, volume 8, pages 1–12, 2010.

[2] R. Alur and P. Cerný. Streaming transducers for algorithmic verification of single-pass list-processing programs. In *POPL 2011*, pages 599–610, 2011.

[3] R. Alur, L. D'Antoni, J. V. Deshmukh, M. Raghothaman, and Y. Yuan. Regular functions and cost register automata. In *LICS 2013*, pages 13–22. IEEE Computer Society, 2013.

[4] R. Alur, L. D'antoni, and M. Raghothaman. DReX: A Declarative Language for Efficiently Computable Regular String Transformations. In *POPL*, 2015.

[5] J. Engelfriet and H. J. Hoogeboom. MSO definable string transductions and two-way finite-state transducers. *ACM Transactions on Computational Logic (TOCL)*, 2(2):216–254, 2001.

[6] E. Filiot, O. Gauwin, P. Reynier, and F. Servais. From two-way to one-way finite state transducers. In *LICS 2013*, pages 468–477, 2013.