Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 325 (2016) 253–276

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Coalgebraic Minimization of Automata by Initiality and Finality

Jurriaan Rot[1,2]

Université de Lyon, ENS de Lyon, CNRS, UCB Lyon 1, LIP

Abstract

Deterministic automata can be minimized by partition refinement (Moore's algorithm, Hopcroft's algorithm) or by reversal and determinization (Brzozowski's algorithm). In the coalgebraic perspective, the first approach can be phrased in terms of a minimization construction along the final sequence of a functor, whereas a crucial part of the second approach is based on a reachability construction along the initial sequence of another functor. We employ this coalgebraic perspective to establish a precise relationship between the two approaches to minimization, and show how they can be combined. Part of these results are extended to an approach for language equivalence of a general class of systems with branching, such as non-deterministic automata.

Keywords: minimization, automata, coalgebra

1 Introduction

The problem of minimizing deterministic automata has been studied since the early days of automata theory, and a number of different approaches have been proposed. Probably the most well-known family of algorithms, which includes Hopcroft's [11] and Moore's algorithm [19] as well as typical textbook constructions [12], is based on a stepwise refinement of a partition of states. Another approach, due to Brzozowski [7], is based on determinization and reversal. That approach appears (and is usually considered) to be fundamentally different than partition refinement [3,24]. To the best of our knowledge, a connection was only established in the work of Champarnaud et al [8] (and further extended in [9]), who explicitly showed how the partition of states that are language equivalent is obtained from the reversed determinized automaton that appears in Brzozowski's algorithm.

[1] This work was performed within the framework of the LABEX MILYON (ANR-10-LABX-0070) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR), and was supported by project ANR 12IS02001 PACE.

[2] Email: jurriaan.rot@ens-lyon.fr

http://dx.doi.org/10.1016/j.entcs.2016.09.042
1571-0661/

Partition refinement can be phrased abstractly as an inductive computation along the final sequence of a functor, generalizing from automata to coalgebras [1]. Starting with [6], Brzozowski's algorithm has also received significant attention from a coalgebraic perspective, as an elegant instance of duality between algebra and coalgebra [5], in several different formulations [5,4,15].

In this paper we employ the coalgebraic perspective on partition refinement and Brzozowski's algorithm to understand and establish their relationship. First, we dualize the construction of [1] and combine it with a variant of the Brzozowski construction from [15], to obtain a minimization construction based on a stepwise computation of reachability along an initial sequence. We then show how the i-th step of this reachability construction yields the i-th partition of states in partition refinement by a simple factorization, thus establishing a fundamental connection between the two minimization constructions. Based on this result, we define a minimization construction that combines partition refinement with the computation of reachability. In our motivating example of deterministic automata, we retrieve the combined minimization construction due to Champarnaud et al [8].

Our Brzozowski construction is based on [15], where it is formulated for systems with branching, such as non-deterministic, alternating and tree automata. In the last part of the paper, we consider such branching systems, and show how the reachability computation yields an abstract procedure for language equivalence.

Outline. In Section 2 we describe the ideas of partition refinement and Brzozowski's algorithm and their connection, for deterministic automata. Section 3 contains preliminaries, Section 4 recalls coalgebraic partition refinement, and Section 5 introduces the dual reachability construction. Section 6 introduces the abstract Brzozowski construction, and Section 7 establishes the connection with partition refinement. Section 8 concerns branching systems. Proofs can be found in the appendix.

Acknowledgments. The author is grateful to Filippo Bonchi, Matias Lee, Damien Pous and Alexandra Silva for comments, suggestions and discussions.

2 Minimization of deterministic automata

We fix an alphabet A, denote the set of words over A by A^* and the empty word by ε. A *deterministic automaton* is a triple (X, o, f) consisting of a set of states X, a transition function $f\colon X \to X^A$ and an output function $o\colon X \to 2$, where $2 = \{0, 1\}$ is a two-element set. Note that the state space X is not required to be finite, and there is no initial state. The semantics of an automaton is a function $l\colon X \to 2^{A^*}$ mapping each state to the language it accepts, inductively defined by $\varepsilon \in l(x)$ iff $o(x) = 1$ and $aw \in l(x)$ iff $w \in l(f(x)(a))$, for any letter $a \in A$ and word $w \in A^*$.

Our aim is to *minimize* deterministic automata: given an automaton (X, o, f) we search the automaton with the least number of states that accepts the same languages as those accepted by the states in X. Formulated slightly more abstractly, the aim is to find a factorization of the semantics $l\colon X \to 2^{A^*}$ as a surjective function $e\colon X \to E$ followed by an injective function $m\colon E \to 2^{A^*}$. Such a factorization uniquely turns the set E into a (minimal) automaton accepting all languages of