Complementation of Non-Deterministic Büchi Automata

Daniel Weibel

October 9, 2014

Contents

1 Introduction

A Büchi complementation construction takes as input a Büchi automaton A and produces as output another Büchi automaton B which accepts the complement language of the input automaton A. Complement language denotes the "contrary" language, that is, B must *accept* (over a given alphabet) every word that A *does not* accept, and must in turn *not accept* every word that A *accepts*.

Büchi automata are finite automata (that is, having a finite number of states) which operate on infinite words (that is, words that "never end"). Operating on infinite words, they belong thus to the category ω-automata. An important application of Büchi automata is in model checking which is a formal system verification technique. There, they are used to represent both, the description of the system to be checked for the presence of a correctness property, and (the negation of) this correctness property itself.

In one approach to model checking, the correctness property is directly specified as a Büchi automaton One approach to model checking requires that the Büchi automaton representing the correctness property is complemented. It is here that the problem of Büchi complementation has one of its practical applications.

The complementation of non-deterministic Büchi automata is hard. It has been proven to be intrinsically exponential in the number of generated states [cite]. That is, the number of states of the output automaton is, in the worst case, an exponential function of the number of states of the input automaton.

2 The Wider Context: System Verification

3 The Büchi Complementation Problem

4 The Fribourg Construction

5 Discussion

6 Conclusions