Available online at www.sciencedirect.com
ScienceDirect

Information Processing Letters 101 (2007) 107–111

Information Processing Letters

www.elsevier.com/locate/ipl

A power-set construction for reducing Büchi automata to non-determinism degree two ☆

Ulrich Ultes-Nitsche *

Department of Computer Science, University of Fribourg, Boulevard de Pérolles 90, CH-1700 Fribourg, Switzerland

Received 22 November 2005; received in revised form 3 August 2006; accepted 21 August 2006

Available online 24 October 2006

Communicated by K. Iwama

Abstract

Büchi automata are finite automata that accept languages of infinitely long strings, so-called ω-languages. It is well known that, unlike in the finite-string case, deterministic and non-deterministic Büchi automata accept different ω-language classes, i.e., that determination of a non-deterministic Büchi automaton using the classical power-set construction will yield in general a deterministic Büchi automaton which accepts a superset of the ω-language accepted by the given non-deterministic automaton.

In this paper, a power-set construction to a given Büchi automaton is presented, which reduces the degree of non-determinism of the automaton to at most two, meaning that to each state and input symbol, there exist at most two distinct successor states. The constructed Büchi automaton of non-determinism degree two and the given Büchi automaton of arbitrary non-determinism degree will accept the same ω-language.

modified subset-construction
Nondeterministic Büchi automaton ——————————> Nondeterministic Büchi automaton with nondeterminism degree 2

Keywords: Formal languages; Büchi automata; Non-determinism; Regular ω-languages

1. Introduction

? w-languages? w-regular languages

It is well known that deterministic and non-deterministic Büchi automata [1] accept different classes of ω-languages [10]. An example is $(a+b)^* \cdot b^\omega$, the ω-language of all ω-strings of a's and b's containing a finite number of a's, which can only be accepted by a non-deterministic Büchi automaton (see Fig. 1), but not by a deterministic one.

In particular the usual subset construction used to determine automata [4] will fail. For the discussed example, the determined automaton will accept all ω-strings of a's and b's containing infinitely many b's: $(a^* \cdot b)^\omega$ (see Fig. 2). In general, the determined version of a Büchi automaton will accept a superset of the given non-deterministic automaton.

In this paper, a power-set construction to a given Büchi automaton is presented, which reduces the degree of non-determinism to at most two, meaning that to each state and input symbol, there exist at most two distinct successor states. The constructed Büchi will accept the same language as the given automaton.

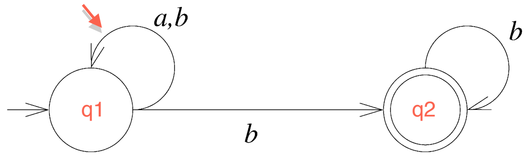
Karpínski showed already in [5] that all regular ω-languages can be accepted by Büchi automata with non-

☆ The work presented in this paper was supported by the Swiss National Science Foundation under grant #200021-103985/1 and by the Hasler Foundation under grant #1922.

* Tel.: +41 (0)26 300 91 49; fax: +41 (0)26 300 97 31.
E-mail address: uun@unifr.ch (U. Ultes-Nitsche).
URL: http://diuf.unifr.ch/tns.

0020-0190/$ – see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.ipl.2006.08.014

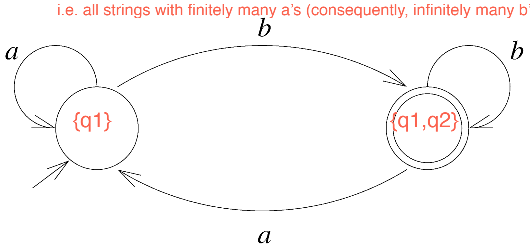
there cannot be infinitely many a's because then the automaton would never reach the accepting state



A

Fig. 1. Büchi automaton accepting $(a+b)^* \cdot b^\omega$.

finitely many a's (and b's) followed by infinitely many b's
i.e. all strings with finitely many a's (consequently, infinitely many b's)



A'

A' accepts L(A) plus something more (L(A) ⊆ L(A'))

Fig. 2. The determined version of Fig. 1 accepting $(a^* \cdot b)^\omega$.

finitely many a's (and b's) followed by infinitely many b's AND
infinitely many a's and infinitely many b's
i.e. all strings with infinitely many b's (consequently, either finitely or infinitely many a's)

determinism degree two.[1] However, Karpínski's construction takes the detour of constructing a Muller automaton [7] to a given regular ω-language, and from that Muller automaton, Karpínski constructs the Büchi automaton of non-determinism degree two.

deterministic

The construction presented in this paper is a much simpler power-set construction directly on a given Büchi automaton of arbitrary non-determinism degree. The construction works analogously to the determination of finite-string-accepting automata, but is performed "modulo accepting and non-accepting states": instead of considering sets of states in general, sets of purely accepting and sets of purely non-accepting states will be considered.

Hence the nondeterminism degree of two. If a successor state consists of non-accepting and accepting states it is split into two distinct states.

The paper is structured as follows: Section 2 introduces the basic notion of a Büchi automaton and presents a version of König's lemma that will be required to prove the main result in Section 3. After presenting a small example in Section 4, the paper will be concluded with a brief discussion of the presented result.

2. Preliminaries

Let Σ be an alphabet (set of finite cardinality). The set of all infinitely long strings (aka ω-strings) over Σ is represented by Σ^ω. Sets of ω-strings, i.e., subsets of Σ^ω, are called ω-languages over Σ.

[1] Non-determinism degree two is therefore the minimal degree of non-determinism which Büchi automata must have if the automaton class is supposed to accept the class of *all* regular ω-languages; automata of non-determinism degree one—i.e., deterministic Büchi automata—are too limited as discussed briefly above using the example $(a+b)^* \cdot b^\omega$.

A finite automaton $A = (Q, \Sigma, \delta, q_0, F)$ consists of a finite set Q of states, an input alphabet Σ, a transition relation $\delta : Q \times \Sigma \to 2^Q$, an initial state $q_0 \in Q$, and a set F of accepting states [4].

Let $x = x_0 x_1 x_2 \ldots \in \Sigma^\omega$ be an ω-string in Σ^ω. A run of A on x is a sequence $q_0 q_1 q_2 \ldots$ of states such that q_0 is A's initial state and $q_{i+1} \in \delta(q_i, x_i)$ for all $i \geqslant 0$. For a run r of A on x, $\omega(r)$ denotes the set of all states that repeat infinitely often in r. A run r is *successful* if and only if $\omega(r) \cap F \neq \emptyset$ (r contains at least one accepting state infinitely often). Automaton A *Büchi-accepts* x if and only if there exists an accepting run of A on x [1,10]. The ω-language represented by A is $L_A = \{x \in \Sigma^\omega \mid A \text{ Büchi-accepts } x\}$. ω-languages that are Büchi-acceptable by some finite automaton are called *regular ω-languages* [10]. The automaton is then called a *Büchi automaton*.

If, for all states q in Q and all input symbols a in Σ, the successor state of q with respect to a is uniquely determined (i.e., $\delta(q, a)$ is a singleton set or the empty set for all states q and symbols a), then A is called deterministic. Otherwise it is called non-deterministic. For ω-languages, the language classes accepted by deterministic and non-deterministic automata differ: There exist ω-languages that can only be represented by non-deterministic finite automata; an example is given in the introduction of this paper.

The *non-determinism degree* $\nu(q)$ of state q is the maximal number of successor states it can reach by reading an input symbol: $\nu(q) = \max_{a \in \Sigma}\{|\delta(q, a)|\}$. The degree of non-determinism $\nu(A)$ of automaton A is the maximal degree of non-determinism of one of its states: $\nu(A) = \max_{q \in Q}\{\nu(q)\}$.

In the next section, the central lemma will be proved using König's lemma [6] in the version of Hoogeboom and Rozenberg [3]:

König's Lemma: if G is a connected graph with infinitely man vertices with finite degree each, then G contains an infinitely long simple path (path with no repeated vertices).

Lemma. *Let $R \subseteq E \times E$ be a relation over an arbitrary set E. For all $n \geqslant 0$, let E_n be a finite nonempty subset of E such that $\bigcup_{n \geqslant 0} E_n$ is infinite and to each $e \in E_{n+1}$ there exists an $f \in E_n$ such that $(f, e) \in R$. Then there exists an infinite sequence $(e_n)_{n \geqslant 0}$ such that $e_n \in E_n$ and $(e_n, e_{n+1}) \in R$ for all $n \geqslant 0$.*

This version: (more or less) a tree with infinitely many vertices and a finite branching has an infinite path.

3. Reducing the degree of non-determinism to two

Let $A = (Q, \Sigma, \delta : Q \times \Sigma \to 2^Q, q_0, F)$ be a (non-deterministic) Büchi automaton. Let its *reduced version with non-determinism degree two* be

$$R = \left(2^F \cup 2^{Q-F}, \Sigma, \bar{\delta} : 2^Q \times \Sigma \to 2^{2^Q}, \{q_0\}, 2^F\right)$$

All subsets of the accepting states

All subsets of the non-accepting states

such that returns two states

the set of the non-accepting ones that would usually be returned

$$\bar{\delta}(q,a) = \{\delta^*(q,a) \cap F, \delta^*(q,a) \cap (Q - F)\}$$

the set of the accepting ones that would usually be returned

with $\delta^*: 2^Q \times \Sigma \to 2^Q$ being the usual extension of δ to sets of states (q is a subset of Q):

$$\delta^*(q,a) = \{r \in Q \mid \exists p \in q\colon r \in \delta(p,a)\}.$$

Note that in this subset construction only sets of states which are entirely accepting or which are entirely non-accepting are considered (i.e., the set of states of R is $2^F \cup 2^{Q-F}$). The initial state is $\{q_0\}$ (singleton sets trivially contain only either accepting or non-accepting states). The accepting states of R are sets of accepting states of A (i.e., the set of accepting states of R is 2^F). $\bar{\delta}$ relates a set of states and a symbol to all reachable accepting and all reachable non-accepting states, respectively ($\delta^*(q,a) \cap F$ and $\delta^*(q,a) \cap (Q - F)$). As usual, R will be reduced by removing all states which are not reachable or which are not co-reachable.[2]

This construction is "nearly a determination step", but by distinguishing accepting from non-accepting states, the definition of $\bar{\delta}$ restricts the degree of non-determinism $\nu(R)$ of R only to two: for each state q and each symbol a, $\bar{\delta}(q,a)$ contains at most two elements.[3]

Subsequently, let A be a Büchi automaton and let R be constructed from A as described above.

Proof that L(A) = L(R)

Lemma 1. $L_A \subseteq L_R$.

The newly constructed automaton R accepts at least all the words the initial automaton A does.

Proof. Let $x = x_0x_1x_2 \ldots \in L_A$. Let then $q_0q_1q_2 \ldots$ be an accepting run of A on x and let r_0 be $\{q_0\}$. Let, for $i \geqslant 0$, r_{i+1} be the one of the two sets $\delta^*(r_i, x_i) \cap F$ and $\delta^*(r_i, x_i) \cap (Q - F)$ which contains q_{i+1}. By construction, $r_0r_1r_2 \ldots$ is a run of R on x. Because infinitely many different q_i are accepting states of A, the infinitely many "matching" states r_i (the ones which contain an accepting q_i) are accepting states of R, and hence $r_0r_1r_2 \ldots$ is an accepting run of R on x. □

Lemma 1 is immediate as the power-set construction always yields an automaton that Büchi-accepts a superset of the original automaton. The next lemma, Lemma 2, is the interesting one as it states that in the power-set construction used to reduce the degree of non-determinism to two, this superset is always the trivial one, leading to the result of Corollary 3 that A and R Büchi-accept the same ω-language.

The new automaton R doesn't accept any new words.

Lemma 2. $L_R \subseteq L_A$.

The initial automaton A accepts all the words the newly constructed automaton R does.

Proof. Let $x = x_0x_1x_2 \ldots \in L_R$. Let $r_0r_1r_2 \ldots$ be an accepting run of R on x. We show that there exists an accepting run $q_0q_1q_2 \ldots$ of A on x. Let $s_i = \{p_{(i)} \mid p \in r_i\}$, i.e., all states in r_i are simply labeled with "(i)". Then the s_i are finite sets for all $i \geqslant 0$, because the r_i are finite, and $\bigcup_{i \geqslant 0} s_i$ has infinite cardinality.[4] For two elements $p_{(i)}$ and $p'_{(j)}$ in $\bigcup_{i \geqslant 0} s_i$ let relation *succ* satisfy $(p_{(i)}, p'_{(j)}) \in succ$ if and only if $j = i + 1$ and $p' \in \delta(p, x_i)$. By definition of $\bar{\delta}$ of R, there exists a $p_{(i)} \in s_i$ to each $p'_{(i+1)} \in s_{i+1}$ such that $(p_{(i)}, p'_{(i+1)}) \in succ$. Application of König's lemma establishes that there exists a sequence $q_{0_{(0)}} q_{1_{(1)}} q_{2_{(2)}} \ldots$ with $q_{0_{(0)}} \in s_0$, $q_{1_{(1)}} \in s_1$, $q_{2_{(2)}} \in s_2$, etc., such that, for all $i \geqslant 0$, $(q_{i_{(i)}}, q_{i+1_{(i+1)}}) \in succ$. Hence, by definition of *succ* and removing of the labels "(i)", $q_0q_1q_2 \ldots$ is a run of A on x such that $q_i \in r_i$. Because $r_0r_1r_2 \ldots$ is an accepting run of R on x, infinitely many different r_i are subsets of F, and thus infinitely many of the q_i are accepting. Hence $q_0q_1q_2 \ldots$ is an accepting run of A on x. □

Note that the above construction fails in the case where the r_i are not either subsets of F or disjoint to F as it were, for instance, the case if one determined A completely. The reason for that observation is that then the run $q_0q_1q_2 \ldots$ constructed in the proof is not guaranteed to be accepting.

The main result of this paper is an immediate consequence of the two lemmas:

Corollary 3. $L_R = L_A$.

4. A small example

Fig. 3 shows a Büchi automaton accepting an ω-language similar to the one accepted by the Büchi automaton of Fig. 1. The Büchi automaton in Fig. 3 accepts "finitely many a's but at least one a": $(a + b)^* \cdot a \cdot b^\omega$ (the only difference is that the automaton in Fig. 1 can accept the infinite string consisting of b's only, whereas the automaton in Fig. 3 cannot accept that ω-string).

[2] One removes all states not reachable from the initial state and one removes all states from which no accepting cycle can be reached.

[3] It contains *at most* two elements because the empty set will be removed from $\bar{\delta}(q,a)$ whenever $\bar{\delta}(q,a)$ contains it.

[4] The union of the s_i is infinite because of the introduced labeling: same elements in r_i and r_j, $i \neq j$, become different in s_i and s_j by labeling them with "(i)" and "(j)", respectively. Making the union of the s_i infinite is the only purpose of the introduced labeling.

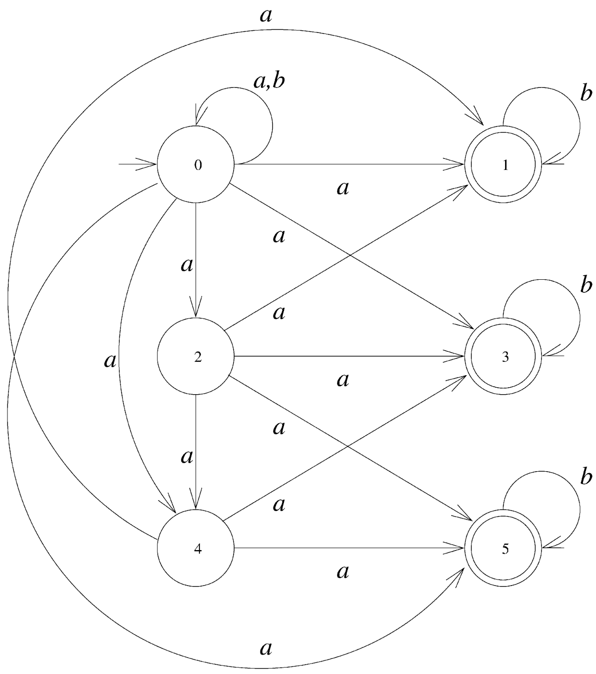


Fig. 3. Büchi automaton accepting $(a+b)^* \cdot a \cdot b^\omega$.

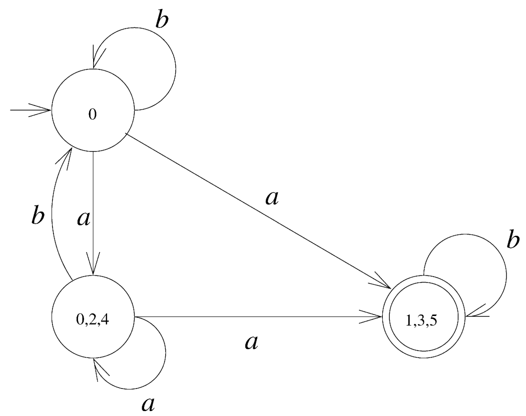


Fig. 4. Reduced Büchi automaton to Fig. 3 accepting the same ω-language.

Fig. 4 shows the result of reducing the degree of non-determinism of the automaton in Fig. 3 to two along the construction given in this paper. The states in Figs. 3 and 4 are numbered to see which states of Fig. 3 have contributed to which states of Fig. 4.

5. Conclusions

The construction and related proofs in this paper showed that the degree of non-determinism of a Büchi automaton can be reduced to two by a simple subset construction without limiting the acceptance capabilities of the automaton. The presented result is a side result of work in direction of constructing a reasonably efficient algorithm for a satisfaction relation of linear-time temporal properties [2] with an inherent fairness condition [8,9,11]. The construction of the reduced Büchi automaton given in this paper will work in parallel with the computation of the synchronous product of a behavior and a property automaton—the synchronous product cannot be computed in parallel with Karpínski's construction [5] of a Büchi automaton of non-determinism degree two. The ultimate goal of future work, using the result of this paper, will be the development of an algorithm for the inherently fair satisfaction of linear-time properties [8,9,11] which performs better than the currently known naïve algorithm using Boolean operations on Büchi and finite-string-accepting automata in the most straightforward fashion.

Acknowledgements

I would like to thank Thierry Nicola and Frank Nießner for many fruitful, interesting discussions through which they have contributed to this paper.

References

[1] J.R. Büchi, On a decision method in restricted second order arithmetic, in: E. Nagel, et al. (Eds.), Proceedings of the International Congress on Logic, Methodology and Philosophy of Science 1960, Stanford University Press, 1962, pp. 1–11.
[2] E.A. Emerson, Temporal and modal logic, in: van Leeuwen [12], pp. 995–1072.
[3] H. Hoogeboom, G. Rozenberg, Infinitary languages: Basic theory and applications to concurrent systems, in: J. de Bakker, W.-P. de Roever, G. Rozenberg (Eds.), Current Trends in Concurrency, in: Lecture Notes in Computer Science, vol. 224, Springer-Verlag, Berlin, 1986, pp. 266–342.
[4] J.E. Hopcroft, R. Motwani, J.D. Ullman, Introduction to Automata Theory, Languages and Computation, Addison–Wesley/Longman, 2001.
[5] M. Karpínski, Almost deterministic ω-automata with existential output condition, Proceedings of the American Mathematical Society 53 (2) (December 1975) 449–452.
[6] D. König, Über eine Schlußweise aus dem Endlichen ins Unendliche (Punktmengen. Kartenfärben. Verwandtschaftsbeziehungen. Schachspiel), Acta Litterarum ac Scientiarum Regiae Universitatis Hungaricae Francisco-Josephinae. Sectio Scientiarum Mathematicarum 3 (1927) 121–130.
[7] D.E. Muller, Infinite sequences and infinite machines, in: AIEE Proceedings of the 4th Annual Symposium on Switching Theory and Logical Design, 1963, pp. 3–16.
[8] U. Nitsche, P. Ochsenschläger, Approximately satisfied properties of systems and simple language homomorphisms, Information Processing Letters 60 (1996) 201–206.

[9] U. Nitsche, P. Wolper, Relative liveness and behavior abstraction (extended abstract), in: Proceedings of the 16th ACM Symposium on Principles of Distributed Computing (PODC'97), Santa Barbara, CA, 1997, pp. 45–52.
[10] W. Thomas, Automata on infinite objects, in: van Leeuwen [12], pp. 133–191.
[11] U. Ultes-Nitsche, S. St James, Improved verification of linear-time properties within fairness—weakly continuation-closed behaviour abstractions computed from trace reductions, Software Testing, Verification and Reliability (STVR) 13 (4) (2003) 241–255.
[12] J. van Leeuwen (Ed.), Formal Models and Semantics, Handbook of Theoretical Computer Science, vol. B, Elsevier, Amsterdam, 1990.