Automata and Reactive Systems

Lecture No. 9

Prof. Dr. Wolfgang Thomas

thomas@informatik.rwth-aachen.de

Lehrstuhl für Informatik VII

RWTH Aachen

McNaughton's Theorem (1966)

4.4 Theorem: If L is Büchi recognizable then L is recognizable by a deterministic Muller automaton.

For the given Büchi automaton $\mathcal{B} = (Q, \Sigma, q_0, \Delta, F)$

define the Muller automaton $\mathcal{M} = (Q', \Sigma, q_0', \delta, \mathcal{F})$ by

- Q' := set of Safra trees over Q.
- q_0' := Safra tree consisting just of root labelled $\{q_0\}$
- $S \in \mathcal{F} \iff$ if some node name appears in each tree $s \in S$, and in some tree $s \in S$ the label of this node name carries the marker "!"

Definition of δ

Define $\delta(s, a)$ (for Safra tree s, input letter a) in four stages as follows:

1. For each node whose label contains final states, branch off a new son containing these final states. (Take as node name a free number $\leq 2|Q|$)
2. To each node label apply the powerset construction via input letter a: $R \rightarrow \{r' \mid \exists r \in R : (r, a, r') \in \Delta\}$
3. Cancel state q if it occurs also in an older brother node. Cancel a node if it carries label $\emptyset$ (unless it is the root)
4. Cancel all sons (and their descendants) if the union of their labels is the parent label, and in this case mark the parent by "!"

Towards the correctness proof

Notation: $P \overset{w}{\rightsquigarrow} R$ for $P, R \subseteq Q$ means

$\forall r \in R\ \exists p \in P$ s.t. $\mathcal{B} : p \xrightarrow{w} r$.

We analyze the case where a node name stays alive and has "!" again and again:

$$\begin{array}{ccccccccccccc}
R_0 & \overset{u_1}{\rightsquigarrow} & P_1 & \overset{v_1}{\rightsquigarrow} & R_1! & \overset{u_2}{\rightsquigarrow} & P_2 & \overset{v_2}{\rightsquigarrow} & R_2! & \dots & P_i & \overset{v_i}{\rightsquigarrow} & R_i! \\
 & & \cup\!\!| & & \| & & \cup\!\!| & & \| & & \cup\!\!| & & \| \\
 & & F_1 & \overset{v_1}{\rightsquigarrow} & Q_1 & & F_2 & \overset{v_2}{\rightsquigarrow} & Q_2 & & F_i & \overset{v_i}{\rightsquigarrow} & Q_i
\end{array}$$

where F_i = set of final states from P_i.

Then $\forall r \in R_i\ \exists p \in R_0$:

$\mathcal{A}$ reaches from p via input $u_1 v_1 u_2 v_2 \dots u_i v_i$ the state r with $\geq i$ visits of final states.

König’s Lemma

4.5 Lemma: An infinite finitely branching tree has an infinite path.

Proof:

Let t be an infinite finitely branching tree.

Define a path π such that each node v on π has the following property: the subtree at v is infinite.

The root has the property by assumption.

If v has the property then we can pick a son v' with the same property (because the tree is finitely branching!)

Iterating this we obtain an infinite path.

Run Lemma

4.6 Lemma: Let $R_0 \overset{u_1v_1}{\rightsquigarrow} R_1! \overset{u_2v_2}{\rightsquigarrow} R_2! \ldots R_i! \overset{u_{i+1}v_{i+1}}{\rightsquigarrow} \ldots$ be as before.

Then on the input $u_1v_1u_2v_2\ldots$ there is a successful run of the Büchi automaton $\mathcal{B}$, starting in a state of R_0.

Proof:

Consider for each state from $r \in R_i$ a run from R_0 to r via $u_1v_1\ldots u_iv_i$.

These runs form a run tree which is infinite and finitely branching.

By König's Lemma there is an infinite run in this tree.

By construction, a final state is visited after each prefix $u_1v_1\ldots u_iv_i$.

Correctness of Safra construction

Claim: $L(\mathcal{B}) = L(\mathcal{M})$

Show first $L(\mathcal{M}) \subseteq L(\mathcal{B})$:

Let $\alpha \in L(\mathcal{M})$

Consider the successful run of Safra trees on α.

Pick a node k which from some point onwards occurs in each Safra tree and has marker "!" infinitely often.

Consider the labels where "!" occurs at k, call them $R_1, R_2, \ldots$.

The Run Lemma applies and yields an infinite run of $\mathcal{B}$ on α.

Show $L(\mathcal{B}) \subseteq L(\mathcal{M})$:

Let $\alpha \in L(\mathcal{B})$, consider a successful run of $\mathcal{B}$ on α, visiting say the final state q again and again.

Consider the $\mathcal{M}$-run of Safra trees on α

If root is marked "!" infinitely often, $\mathcal{M}$ accepts.

Otherwise look at first occurrence of q afterwards:

Here q is put into a son of the root, and the Büchi run finally stays in a fixed son k_1 of the root.

If k_1 is marked "!" infinitely often, $\mathcal{M}$ accepts.

Otherwise we continue analogously and get a son k_2 of k_1 where the Büchi run finally stays.

At some stage the marker "!" occurs infinitely often, otherwise height of the Safra trees would be unbounded.

Rabin automata: Motivation

We may define the sets S which form the acceptance component $\mathcal{F}$ by two conditions:

For some node name j

- S should not contain a tree without node name j
- S should contain a tree where node name j appears with marker "!"

Define

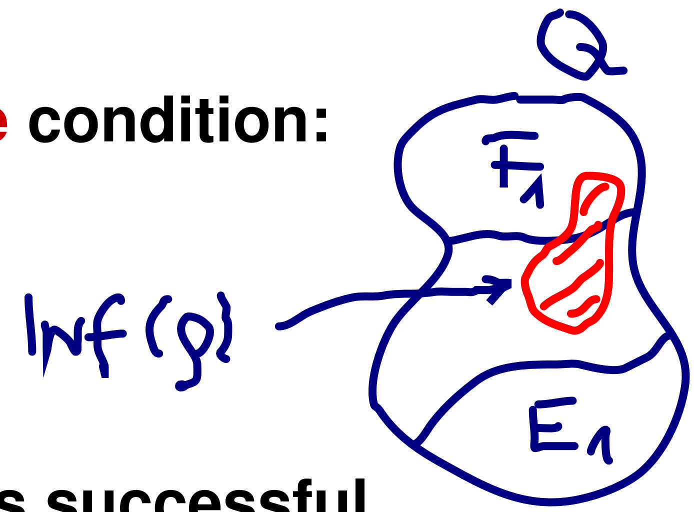
- E_j = set of Safra trees without node name j
- F_j = tree where node name j appears with marker "!"

Then: ρ is successful if for some j,
$\mathrm{Inf}(\rho) \cap E_j = \emptyset$ and $\mathrm{Inf}(\rho) \cap F_j \neq \emptyset$

A (deterministic) Rabin automaton is of the form

$\mathcal{A} = (Q, \Sigma, q_0, \delta, \Omega)$ **where** $\Omega = ((E_1, F_1), \ldots, (E_k, F_k))$ **is a list of "accepting pairs" with** $E_i, F_i \subseteq Q$,

used with the following Rabin acceptance condition:



A run ρ **is successful if for some** $j \leq k$

$$\mathrm{Inf}(\rho) \cap E_j = \emptyset \text{ and } \mathrm{Inf}(\rho) \cap F_j \neq \emptyset$$

$\mathcal{A}$ **accepts** α **if the unique run of** $\mathcal{A}$ **on** α **is successful.**

$$L(\mathcal{A}) = \{\alpha \mid \mathcal{A} \text{ accepts } \alpha\}$$

L **is called Rabin recognizable if** $L = L(\mathcal{A})$ **for a Rabin automaton** $\mathcal{A}$

Equivalence Theorem

The Safra construction transforms a Büchi automaton into a deterministic Rabin automaton

A Rabin automaton $\mathcal{R} = (Q, \Sigma, q_0, \delta, \Omega)$ with $\Omega = ((E_1, F_1), \ldots, (E_k, F_k))$ is equivalent to the Muller automaton $\mathcal{M} = (Q, \Sigma, q_0, \delta, \mathcal{F})$ with

$$P \in \mathcal{F} \iff \bigvee_{j=1}^{k} (P \cap E_j = \emptyset \wedge P \cap F_j \neq \emptyset)$$

So we have proved

4.7 Theorem: For an ω-language, the following are equivalent:

1. L is recognized by a nondeterministic Büchi automaton.
2. L is recognized by a deterministic Rabin automaton.
3. L is recognized by a determinsitic Muller automaton.

Complexity of the Safra construction

Question:

Given a Büchi automaton with n states, how many states do we need for an equivalent deterministic Rabin automaton?

Plan:

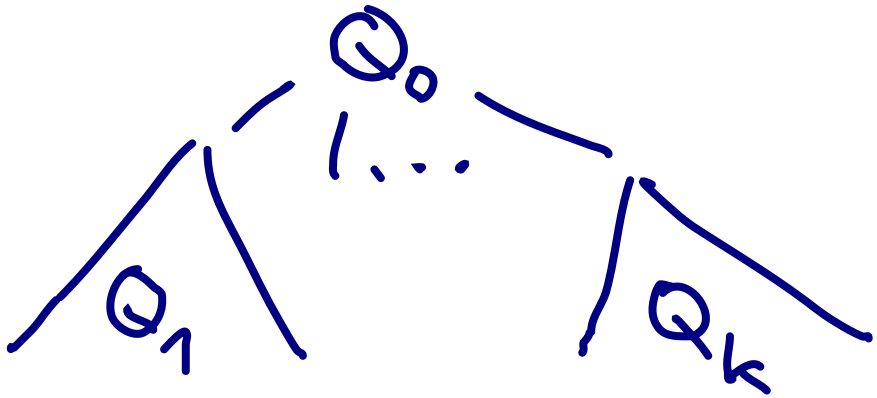
- First we analyze the Safra construction to obtain an upper bound.
- We obtain $2^{O(n \log n)}$ states, the growth rate of $n!$
- We show that this growth rate is necessary.

4.8 Lemma: For $|Q| = n$, a Safra tree over Q has $\leq n$ nodes.

Proof: by induction on the height h of Safra trees.

$h = 0$: Safra tree has one node, hence $\leq n$ nodes.

$h + 1$:



$Q_1, \ldots Q_k$ are disjoint, $\subsetneq Q_0 \subseteq Q$.

Subtrees are Safra trees over $Q_1, \ldots Q_k$,

by induction hypothesis with $\leq |Q_1|, \ldots, \leq |Q_k|$ nodes.

Total number of nodes $\leq 1 + |Q_1| + \ldots + |Q_k| \leq |Q|$

Number of Safra trees

If q occurs in Safra tree, there is a unique node ("characteristic node") which contains q but such that no son contains q.

We need four functions to descibe a Safra tree: $(2n+1)^n$

1. function $Q \to \{0, \ldots, 2n\}$ giving the characteristic node
2. "!"-label function: $\{1, \ldots, 2n\} \to \{0, 1\}$ (Value 1 indicates presence of "!") 2^{2n}
3. parent function: $\{1, \ldots, 2n\} \to \{0, \ldots, 2n\}$ $(2n+1)^{2n}$
4. older-brother function: $\{1, \ldots, 2n\} \to \{0, \ldots, 2n\}$ — " —

Number of Safra trees $\leq$ number of such functions

$$\leq (2n+1)^n \cdot 2^{2n} \cdot (2n+1)^{2n} \cdot (2n+1)^{2n}$$
$$\leq (2n+1)^{7n} \in 2^{O(n \log n)}$$

$n^n = (2^{\log n})^n$

4.9 Theorem (Safra):

A Büchi automaton with n states can be transformed into a deterministic Rabin automaton with $2^{O(n \log n)}$ states and $O(n)$ accepting pairs.

4.10 Theorem (M. Michel 1988, C. Löding 1998):

There is no translation from Büchi automata with $O(n)$ states to deterministic Rabin automata with $2^{O(n)}$ states.

Proof strategy:

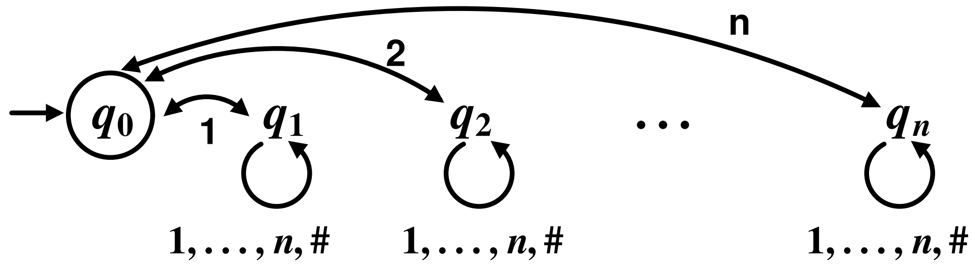
1. Definition of ω-language $L_n \subseteq \{1, \ldots, n, \#\}^\omega$, recognizable by a Büchi automaton with $O(n)$ states.
2. Proof that L_n is not recogizable by a deterministic Rabin automaton with $2^{O(n)}$ states. $\geq n!$

Definition of L_n by a Büchi automaton $\mathcal{B}_n$ over the alphabet $\{1, \ldots, n, \#\}$

BSP.:

2 22

1 13 31



Remark: $\alpha \in L_n \iff$

$(*)$ α **starts with some i_1 and there are letters $i_1, \ldots, i_k \in \{1, \ldots, n\}$ (pairwise distinct) such that the segments of letter pairs $i_1 i_2, i_2 i_3, \ldots, i_{k-1} i_k, i_k i_1$ occur infinitely often in α**

Characterization of L_n

Remark: $\alpha \in L_n \iff$

$(*)$ α starts with some i_1 and there are letters $i_1, \ldots, i_k \in \{1, \ldots, n\}$ (pairwise distinct) such that the segments of letter pairs $i_1 i_2, i_2 i_3, \ldots, i_{k-1} i_k, i_k i_1$ occur infinitely often in α

Proof of $\Leftarrow$:

Assume $(*)$ with $i_1, \ldots, i_k$. Define a successful run of the Büchi automaton:

Go to q_{i_1} and stay there. By first input pair $i_1 i_2$ do

$q_{i_1} \xrightarrow{i_1} q_0 \xrightarrow{i_2} q_{i_2}$.

Similarly with $i_2 i_3, i_3 i_4, \ldots$ in cycles.

This ensures infinitely many visits to q_0, i.e. acceptance.

Proof of $\Rightarrow$:

Assume the Büchi automaton accepts α but $(*)$ fails.

Pick position p in α such that the letter pairs $i_1 i_2$ occurring later will occur in fact infinitely often.

If state $q_i \neq q_0$ is visited after p and q_0 later than that, then no return to q_i is possible.

(Otherwise we would get a cycle as in $(*)$)

Since $q_i \neq q_0$ was arbitrary, the run would eventually stay in q_0.

Contradiction.