Model Checking

Büchi Automata

Joost-Pieter Katoen and Tim Quatmann

Software Modeling and Verification Group

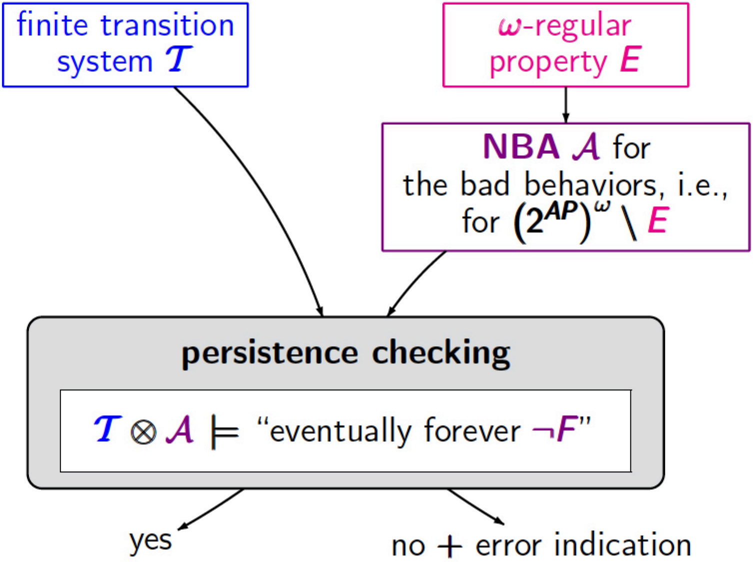
RWTH Aachen, SoSe 2022

Overview

1. Büchi Automata
2. NBA and ω-Regular Languages
3. Checking Non-emptiness of NBA $\mathcal{L}_\omega(\mathcal{A}) = \emptyset ?$
4. Deterministic Büchi Automata $DBA \subsetneq NBA$
5. Generalised Nondeterministic Büchi Automata

Overview

Verifying ω-Regular Properties



ω-Regular Properties

Definition: ω-regular language

The set $\mathfrak{L}$ of infinite words over the alphabet Σ is ω-regular if $\mathfrak{L} = \mathfrak{L}_\omega(G)$ for some ω-regular expression G over Σ.

$$E_1 \cdot F_1^{\omega} + \ldots + E_n \cdot F_n^{\omega}$$

Definition: ω-regular properties

LT property E over AP is ω-regular if E is an ω-regular language over 2^{AP}.

ω-Regular Properties

Definition: ω-regular language

The set $\mathfrak{L}$ of infinite words over the alphabet Σ is ω-regular if $\mathfrak{L} = \mathfrak{L}_\omega(G)$ for some ω-regular expression G over Σ.

Definition: ω-regular properties

LT property E over AP is ω-regular if E is an ω-regular language over 2^{AP}.

We will see that this is equivalent to:

LT property E over AP is ω-regular if E is accepted by a non-deterministic Büchi automaton (over the alphabet 2^{AP}).

But not by a deterministic Büchi automaton.

Example ω-Regular Properties

- Any invariant E is an ω-regular property
  - Φ^ω describes E with invariant condition Φ

- Any regular safety property E is an ω-regular property
  - $\overline{E} = \mathit{BadPref}(E) . \left(2^{AP}\right)^\omega$ is ω-regular
  - and ω-regular languages are closed under complement

Example ω-Regular Properties

- Any invariant E is an ω-regular property
  - Φ^ω describes E with invariant condition Φ
- Any regular safety property E is an ω-regular property
  - $\overline{E} = BadPref(E).\left(2^{AP}\right)^\omega$ is ω-regular
  - and ω-regular languages are closed under complement
- Let $\Sigma = \{ a, b \}$ Then:
  - Infinitely often a:

$$((\underbrace{\varnothing + \{ b \}}_{\neg a})^*.(\underbrace{\{ a \} + \{ a, b \}}_{a}))^\omega \equiv (\neg a^* \, a)^\omega$$

  - eventually a:

$$\left(2^{AP}\right)^*.(\{ a \} + \{ a, b \}).\left(2^{AP}\right)^\omega \equiv true^* \; a \; true^\omega$$

Shorthand Notation

Examples for $AP = \{a, b\}$

- invariant with invariant condition $a \lor \neg b$

$$(a \lor \neg b)^\omega \quad \widehat{=} \quad (\emptyset + \{a\} + \{a, b\})^\omega$$

- "infinitely often a"

$$((\neg a)^*.a)^\omega \quad \widehat{=} \quad ((\emptyset + \{b\})^*.(\{a\} + \{a, b\}))^\omega$$

- "from some moment on a":

$$true^*.a^\omega$$

- "whenever a then b will hold somewhen later"

$$(\underline{(\neg a)^*.a.true^*.b})^*.(\neg a)^\omega + (\underline{(\neg a)^*.a.true^*.b})^\omega$$

Julius Richard Büchi

Julius Richard Büchi (1924 – †1984)

Nondeterministic Büchi automata

Definition: Nondeterministic Büchi automaton

A nondeterministic Büchi automaton (NBA) $\mathfrak{A} = (Q, \Sigma, \delta, Q_0, F)$ with:

- Q is a finite set of states
- Σ is an alphabet
- $\delta : Q \times \Sigma \to 2^Q$ is a transition function
- $Q_0 \subseteq Q$ a set of initial states
- $F \subseteq Q$ is a set of accept (or: final) states.

This definition is the same as for NFA.

The acceptance condition of NBA is different though.

Language of a Büchi Automaton

- NBA $\mathfrak{A} = (Q, \Sigma, \delta, Q_0, F)$ and infinite word $w = A_1 A_2 \ldots \in \Sigma^\omega$
- A run for w in $\mathfrak{A}$ is an infinite sequence $q_0 \, q_1 \, \ldots \in Q^\omega$ such that:
  - $q_0 \in Q_0$ and $q_i \xrightarrow{A_{i+1}} q_{i+1}$ for all $0 \leq i$
- Run $q_0 \, q_1 \, \ldots$ is accepting if $q_i \in F$ for infinitely many i

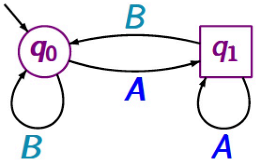
Language of a Büchi Automaton

- NBA $\mathfrak{A} = (Q, \Sigma, \delta, Q_0, F)$ and infinite word $w = A_1 A_2 \ldots \in \Sigma^\omega$
- A run for w in $\mathfrak{A}$ is an infinite sequence $q_0 \, q_1 \, \ldots \in Q^\omega$ such that:
  - $q_0 \in Q_0$ and $q_i \xrightarrow{A_{i+1}} q_{i+1}$ for all $0 \leq i$
- Run $q_0 \, q_1 \, \ldots$ is accepting if $q_i \in F$ for infinitely many i
- The accepted language of $\mathfrak{A}$:

$$\mathfrak{L}_\omega(\mathfrak{A}) = \left\{ w \in \Sigma^\omega \mid \mathfrak{A} \text{ has an accepting run for } w \right\}$$

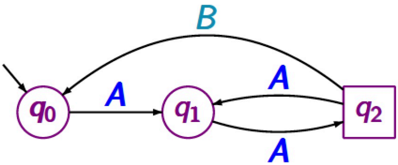
- NBA $\mathfrak{A}$ and $\mathfrak{A}'$ are equivalent if $\mathfrak{L}_\omega(\mathfrak{A}) = \mathfrak{L}_\omega(\mathfrak{A}')$

Examples



accepted language:
set of all infinite words that contain infinitely many A's

$$(B^*.A)^\omega$$

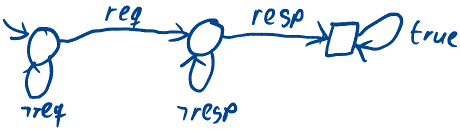


accepted language:
"every B is preceded by a positive even number of A's"

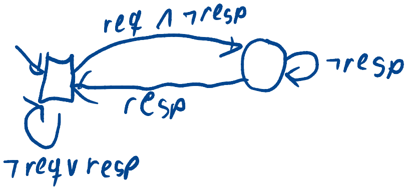
$$((A.A)^+.B)^\omega + ((A.A)^+.B)^*.A^\omega$$

NBA for LT Properties

- "a request eventually leads to a response"

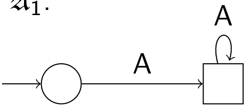


- "on each request eventually a response is provided"

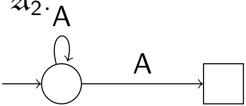


NBA versus NFA

$\mathfrak{A}_1$:



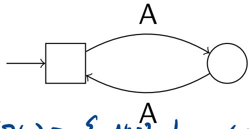
$\mathfrak{A}_2$:



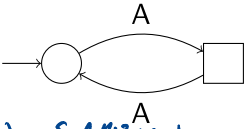
NFA: $\mathcal{L}(\mathfrak{A}_1) = \{ A^{n+1} \mid n \in \mathbb{N} \} = \mathcal{L}(\mathfrak{A}_2)$ $\quad \mathfrak{A}_1 \equiv_{NFA} \mathfrak{A}_2$

NBA: $\mathcal{L}_\omega(\mathfrak{A}_1) = \{ A^\omega \} \neq \mathcal{L}_\omega(\mathfrak{A}_2) = \emptyset$ $\quad \mathfrak{A}_1 \not\equiv_{NBA} \mathfrak{A}_2$

$\mathfrak{A}_3$:



$\mathfrak{A}_4$:



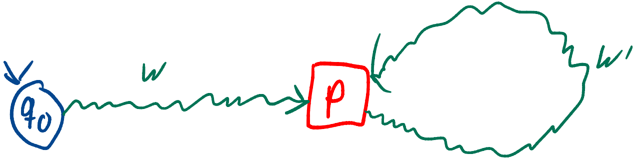
NFA: $\mathcal{L}(\mathfrak{A}_3) = \{ A^{n\cdot 2} \mid n \in \mathbb{N} \}$ $\qquad \mathcal{L}(\mathfrak{A}_4) = \{ A^{n\cdot 2+1} \mid n \in \mathbb{N} \}$

NBA: $\mathcal{L}_\omega(\mathfrak{A}_3) = \{ A^\omega \} = \mathcal{L}_\omega(\mathfrak{A}_4)$

Accepting Runs

- Run $q_0\, q_1 \,\ldots$ in $\mathfrak{A}$ is accepting if $q_i \in F$ for infinitely many i
- Since F is finite we get:

For every accepting run $q_0\, q_1 \,\ldots$ in NBA $\mathfrak{A} = (Q, \Sigma, \delta, Q_0, F)$ there is $p \in F$ with $p = q_i$ for infinitely many i.



Overview

NBA and ω-Regular Languages

Theorem

1. For every NBA $\mathfrak{A}$, the language $\mathfrak{L}_\omega(\mathfrak{A})$ is ω-regular.
2. For every ω-regular language L, there is an NBA $\mathfrak{A}$ with $L = \mathfrak{L}_\omega(\mathfrak{A})$.

Proof.

The next couple of slides. First consider the first part. □

From NBA to ω-Regular Expressions

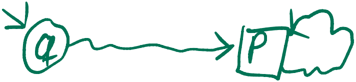
For every NBA $\mathfrak{A}$, the language $\mathfrak{L}_\omega(\mathfrak{A})$ is ω-regular.

Proof.

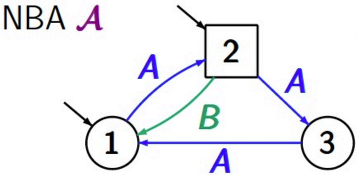
Let NBA $\mathfrak{A} = (Q, \Sigma, \delta, Q_0, F)$ and let $p, q \in Q$ be states in $\mathfrak{A}$.
Define $\mathfrak{A}_{q,p}$ as the NFA $(Q, \Sigma, \delta, q, \{ p \})$.
Then:

$$\mathfrak{L}_\omega(\mathfrak{A}) = \bigcup_{q \in Q_0} \bigcup_{p \in F} \mathfrak{L}(\mathfrak{A}_{q,p}). \left(\mathfrak{L}(\mathfrak{A}_{p,p}) \setminus \{ \epsilon \} \right)^\omega$$

is ω-regular as $\mathfrak{L}(\mathfrak{A}_{q,p})$ and $\mathfrak{L}(\mathfrak{A}_{p,p})$ are regular. □



Example NBA to ω-Regular Expression (1)



$$\mathcal{L}_\omega(\mathcal{A}) = L_{12}(L'_{22})^\omega \cup L_{22}(L'_{22})^\omega$$

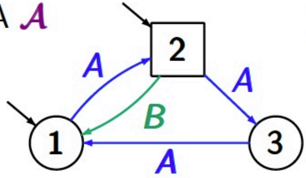
$$L_{12} = \mathcal{L}(\mathcal{A}_{12})$$
$$L_{22} = \mathcal{L}(\mathcal{A}_{22})$$
$$L'_{22} = L_{22} \setminus \{\varepsilon\}$$

$$\mathfrak{L}_\omega(\mathfrak{A}) = \bigcup_{q \in \underbrace{Q_0}_{\{1,2\}}} \bigcup_{p \in \underbrace{F}_{\{2\}}} \mathfrak{L}(\mathfrak{A}_{q,p}).\left(\mathfrak{L}(\mathfrak{A}_{p,p}) \setminus \{\epsilon\}\right)^\omega$$

$\rightsquigarrow$ $2 \cdot 1 = 2$ terms

Example NBA to ω-Regular Expression (2)

NBA $\mathcal{A}$



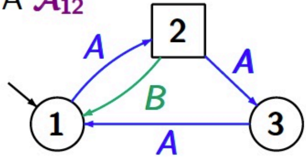
$$\mathcal{L}_\omega(\mathcal{A}) = L_{12}(L'_{22})^\omega \cup L_{22}(L'_{22})^\omega$$

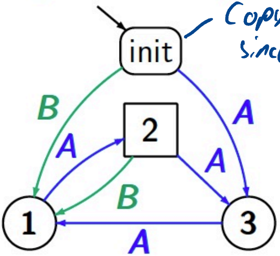
$$L_{12} = \mathcal{L}(\mathcal{A}_{12})$$
$$L_{22} = \mathcal{L}(\mathcal{A}_{22})$$
$$L'_{22} = L_{22} \setminus \{\varepsilon\}$$

$L_{12} \mathrel{\widehat{=}} A.(B.A + A.A.A)^*$

$L'_{22} \mathrel{\widehat{=}} (B.A + A.A.A)^+$

NFA $\mathcal{A}_{12}$

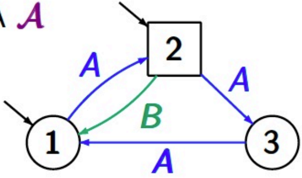




Copy of 2 since $2 \in Q_0 \cap F$

Example NBA to ω-Regular Expression (3)

NBA $\mathcal{A}$

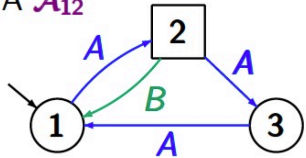


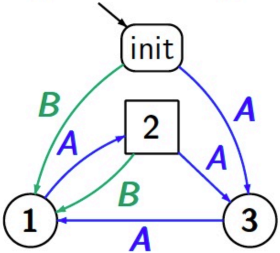
language of $\mathcal{A}$:

$$
\begin{aligned}
& A.(B.A + A.A.A)^\omega \\
+ \; & (B.A + A.A.A)^\omega \\
\equiv \; & (A + \varepsilon).(B.A + A.A.A)^\omega
\end{aligned}
$$

$L_{12} \mathrel{\widehat{=}} A.(B.A + A.A.A)^*$ $\qquad$ $L'_{22} \mathrel{\widehat{=}} (B.A + A.A.A)^+$

NFA $\mathcal{A}_{12}$





NBA and ω-Regular Languages

1 => 2 ✓

2 => 1

Theorem

1. For every NBA $\mathfrak{A}$, the language $\mathfrak{L}_\omega(\mathfrak{A})$ is ω-regular.
2. For every ω-regular language L, there is an NBA $\mathfrak{A}$ with $L = \mathfrak{L}_\omega(\mathfrak{A})$.

Proof.

The next couple of slides. Now consider the second part. □

From ω-Regular Expression to NBA

- How to construct an NBA for the ω-regular expression:

$$G = E_1.F_1^\omega + \ldots + E_n.F_n^\omega \text{ ?}$$

where E_i and F_i are regular expressions over alphabet Σ with $\varepsilon \notin F_i$

From ω-Regular Expression to NBA

- How to construct an NBA for the ω-regular expression:

$$G = \underline{E_1.F_1^\omega} + \ldots + \underline{E_n.F_n^\omega}\ ?$$

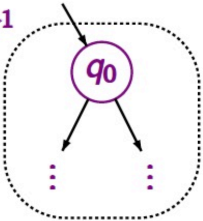
where E_i and F_i are regular expressions over alphabet Σ with $\varepsilon \notin F_i$

- Use operators on NBA mimicking operators on ω-regular expressions. Let NFA $\mathfrak{A}_i$ for E_i and $\mathfrak{B}_i$ for F_i.

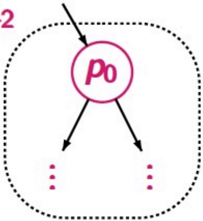
  1. construct NBA $\mathfrak{B}_i^\omega$ for expression F_i^ω — omega on NFA
  2. construct NBA $\mathfrak{C}_i = \mathfrak{A}_i.\mathfrak{B}_i^\omega$ for expression $E_i.F_i^\omega$ — concatenating NFA and NBA
  3. construct NBA for $\bigcup_{0<i\leq n} \mathfrak{L}_\omega(\mathfrak{C}_i)$ — union

Union

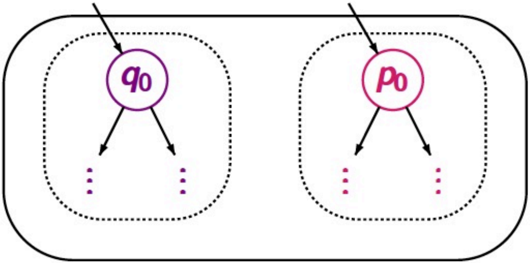
NBA $\mathcal{A}_1$



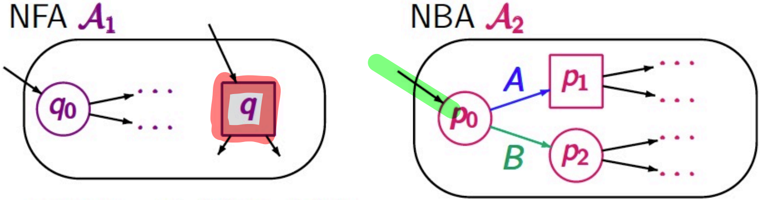
NBA $\mathcal{A}_2$



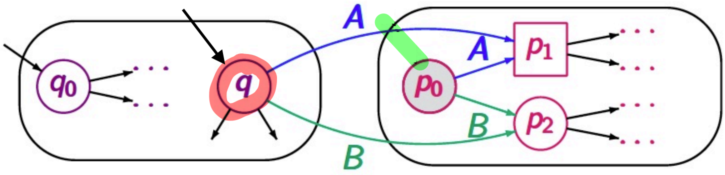
NBA for $\mathcal{L}_\omega(\mathcal{A}_1) \cup \mathcal{L}_\omega(\mathcal{A}_2)$



Concatenating an NFA and NBA

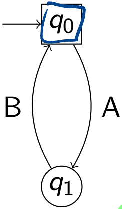


NBA for $\mathcal{L}(\mathcal{A}_1).\mathcal{L}_\omega(\mathcal{A}_2)$:

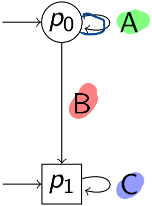


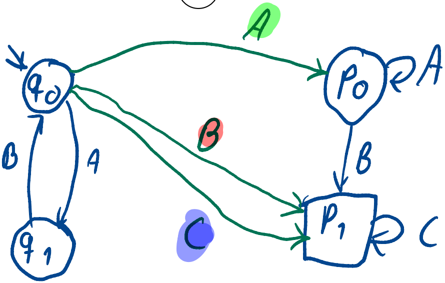
Example: Concatenating NFA and NBA

NFA $\mathfrak{A}_1$:



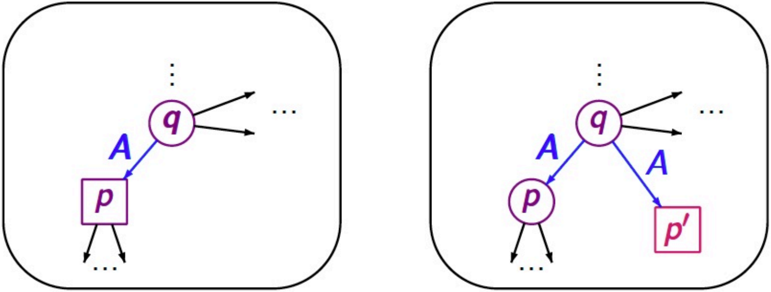
NBA $\mathfrak{A}_2$:





The Omega-Operator on NFA (1)

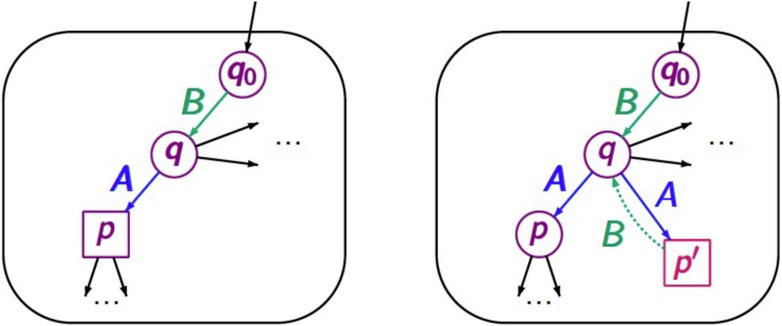
Step 1: make sure all final states in the NFA are terminal



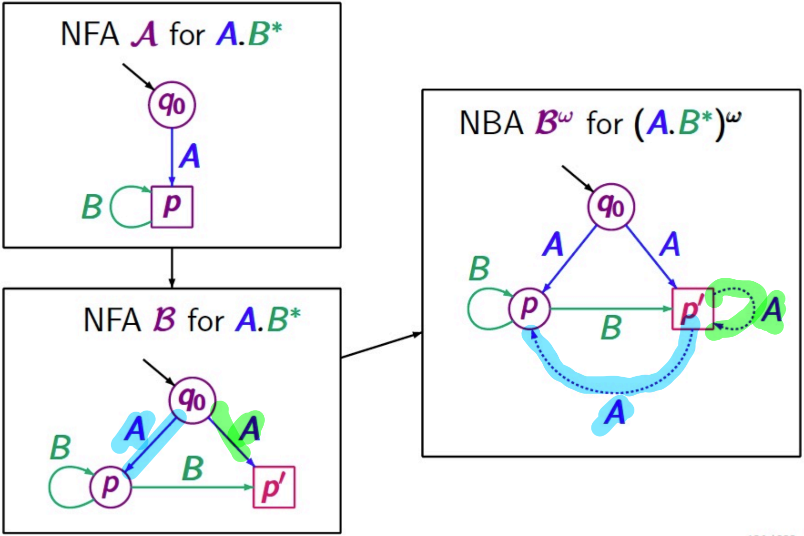
... add a new final state p' ...

The Omega-Operator on NFA (2)

Step 2: for every successor q of the initial state q_0,
add $p' \xrightarrow{B} q$ for each B-transition from q_0 to q



Example: The Omega-Operator on NFA



Overview

Extended Transition Function

Extend the transition function δ to $\delta^* : Q \times \Sigma^* \rightarrow 2^Q$ by:

$$\delta^*(q, \varepsilon) = \{ q \} \quad \text{and} \quad \delta^*(q, A) = \delta(q, A)$$

$$\delta^*(q, A_1 A_2 \ldots A_n) = \bigcup_{p \in \delta(q, A_1)} \delta^*(p, A_2 \ldots A_n)$$

$\delta^*(q, w) =$ set of states reachable from q for the word w

Checking Non-emptiness

For every NBA $\mathfrak{A}$,

$$\mathfrak{L}_\omega(\mathfrak{A}) \neq \varnothing$$

if and only if

$$\underbrace{\exists q_0 \in Q_0.\ \exists q \in F.\ \Big(\exists w \in \Sigma^*.\ q \in \delta^*(q_0, w)\Big) \wedge \Big(\exists v \in \Sigma^+.\ q \in \delta^*(q, v)\Big)}_{\text{there is a reachable accept state on a cycle}}$$

Checking Non-emptiness

For every NBA $\mathfrak{A}$,

$$\mathfrak{L}_\omega(\mathfrak{A}) \neq \varnothing$$

if and only if

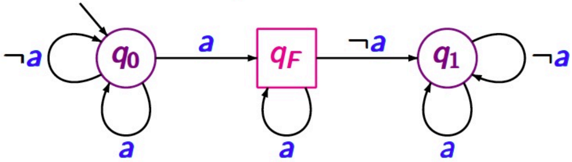
$$\underbrace{\exists q_0 \in Q_0.\ \exists q \in F.\ \Big(\exists w \in \Sigma^*.\ q \in \delta^*(q_0, w)\Big) \wedge \Big(\exists v \in \Sigma^+.\ q \in \delta^*(q, v)\Big)}_{\text{there is a reachable accept state on a cycle}}$$

The emptiness problem for NBA $\mathfrak{A}$
can be solved by graph algorithms in time $O(|\mathfrak{A}|)$

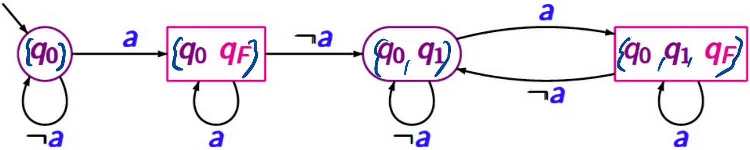
Overview

The Powerset Construction for NBA Fails

NBA for "eventually forever a"



powerset construction



DBA for "infinitely often a"

Deterministic Büchi Automata

Definition: Deterministic Büchi automaton

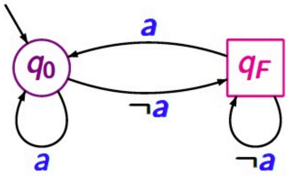
Büchi automaton $\mathfrak{A}$ is deterministic if

$$|Q_0| \leq 1 \quad \text{and} \quad |\delta(q, A)| \leq 1 \quad \text{for all } q \in Q \text{ and } A \in \Sigma.$$

A DBA is total if both inequalities are equalities.

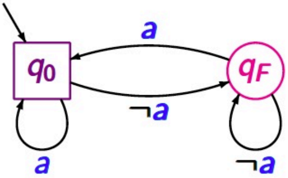
A total DBA has a unique run for each input word.

Complementation on DBA Fails



DBA for
"infinitely often $\neg a$"

complement automaton



DBA for
"infinitely often a"

> There is **no DBA** for the LT-property
> "eventually forever a"

DBA Are Less Expressive Than NBA

There is no DBA that accepts $\mathfrak{L}_\omega((A+B)^* B^\omega)$.

NFA and DFA are equally expressive but NBA and DBA are not!

There is no DBA that accepts $\underbrace{\mathfrak{L}_\omega((A+B)^* B^\omega)}_{\text{"finitely many } A\text{"}}. =: L$

Proof (sketch).

By contraposition. Assume that
$L = L_\omega(\mathcal{A})$ for some DBA $\mathcal{A} = (Q, \Sigma, \{q_0\}, F)$

- Let $\sigma_1 = B^\omega \in L \rightsquigarrow \exists n_1 \in \mathbb{N} : \delta^*(q_0, B^{n_1}) = q_1 \in F$
  since $\mathcal{A}$ is DBA, q_1 is unique
- Let $\sigma_2 = B^{n_1} A B^\omega \in L \rightsquigarrow \exists n_2 \in \mathbb{N} : \delta^*(q_0, B^{n_1} A B^{n_2})$
  $= q_2 \in F$
- Let $\sigma_3 = B^{n_1} A B^{n_2} A B^\omega \in L \rightsquigarrow \exists n_3 \in \mathbb{N} : \delta^*(q_0, B^{n_1} A B^{n_2} A B^{n_3})$
  $= q_3 \in F$

$\vdots$

This reasoning yields a infinite sequence $n_1\ n_2\ n_3 \ldots$
and a sequence of accepting states $q_1\ q_2\ q_3 \ldots$
s.t. $\delta^*(q_0, B^{n_1} A B^{n_2} A \ldots B^{n_{i-1}} A B^{n_i}) = q_i \in F$ for all $i > 0$

Since F is finite there is $i < j$ s.t.

$$\delta^*(q_0, B^{n_1} A \ldots A B^{n_i}) = q_i = q_j = \delta^*(q_0, B^{n_1} A \ldots A B^{n_j})$$

thus, DBA $\mathcal{A}$ has an acc. run on

$$B^{n_1} A \dots A \underset{\substack{\uparrow \\ q_i \in F}}{B}^{n_i} \left(A B^{n_{i+1}} A \dots A \underset{\substack{\uparrow \\ q_i = q_j \in F}}{B}^{n_j} \right)^{\omega} \notin L$$

↯ contradiction

Overview

Generalized Büchi Automata

- NBA are as expressive as ω-regular languages
- Variants of NBA do exist that are equally expressive
  - Muller, Rabin, and Streett automata
  - generalized Büchi automata (GNBA)

Generalized Büchi Automata

- NBA are as expressive as ω-regular languages

- Variants of NBA do exist that are equally expressive
  - Muller, Rabin, and Streett automata
  - generalized Büchi automata (GNBA)

- GNBA have multiple accept sets $F_1, \ldots, F_k$, with $k \in \mathbb{N}$ and $F_i \subseteq Q$
  - a run is accepting if all F_i are visited infinitely often
  - for k=0, all runs are accepting
  - for k=1, this is the same as for NBA

- Why considering GNBA?
  - they ease relating temporal logic and automata
  - they allow to define intersection of NBA

Generalized Büchi Automata

Definition: Generalized Büchi automata

A generalized NBA (GNBA) $\mathfrak{G}$ is a tuple $(Q, \Sigma, \delta, Q_0, \mathfrak{F})$ where Q, Σ, δ, Q_0 are as before and

$$\mathfrak{F} = \{ F_1, \ldots, F_k \} \quad \text{with} \quad F_i \subseteq Q$$

for some natural $k \in \mathbb{N}$.

Run $q_0\, q_1 \ldots \in Q^\omega$ is accepting if $\forall F_j \in \mathfrak{F}$: $q_i \in F_j$ for infinitely many i

Generalized Büchi Automata

Definition: Generalized Büchi automata

A generalized NBA (GNBA) $\mathfrak{G}$ is a tuple $(Q, \Sigma, \delta, Q_0, \mathfrak{F})$ where Q, Σ, δ, Q_0 are as before and

$$\mathfrak{F} = \{ F_1, \ldots, F_k \} \quad \text{with} \quad F_i \subseteq Q$$

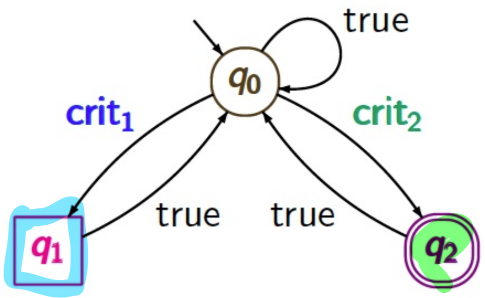
for some natural $k \in \mathbb{N}$.

Run $q_0 \, q_1 \ldots \in Q^\omega$ is accepting if $\forall F_j \in \mathfrak{F}$: $q_i \in F_j$ for infinitely many i

The size of $\mathfrak{G}$, denoted $|\mathfrak{G}|$, is the number of states and transitions in $\mathfrak{G}$:

$$|\mathfrak{G}| = |Q| + \sum_{q \in Q} \sum_{A \in \Sigma} |\, \delta(q, A) \,|$$

Example



where $\mathfrak{F} = \{ \{ q_1 \}, \{ q_2 \} \}$

Specifies the LT property "infinitely often $crit_1$ and infinitely often $crit_2$"

GNBA and NBA are Equally Expressive

For every GNBA $\mathfrak{G}$ there exists an NBA $\mathfrak{A}$ with

$$\mathfrak{L}_\omega(\mathfrak{G}) \;=\; \mathfrak{L}_\omega(\mathfrak{A}) \quad \text{with} \quad |\mathfrak{A}| = O(|\mathfrak{G}| \cdot |\mathfrak{F}|)$$

where $\mathfrak{F} = \{ F_1, \ldots, F_k \}$ denotes the set of acceptance sets in $\mathfrak{G}$.

GNBA and NBA are Equally Expressive

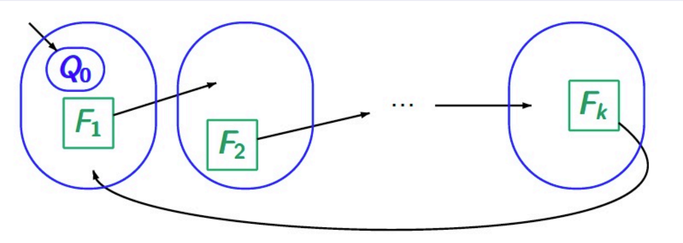
For every GNBA $\mathfrak{G}$ there exists an NBA $\mathfrak{A}$ with

$$\mathfrak{L}_\omega(\mathfrak{G}) = \mathfrak{L}_\omega(\mathfrak{A}) \quad \text{with} \quad |\mathfrak{A}| = O(|\mathfrak{G}| \cdot |\mathfrak{F}|)$$

where $\mathfrak{F} = \{ F_1, \ldots, F_k \}$ denotes the set of acceptance sets in $\mathfrak{G}$.

Proof (sketch).

For k=0, 1, this result follows directly. For $k > 1$, make k copies:



Constructing an NBA from a GNBA

- Let $\mathfrak{G} = (Q, \Sigma, \delta, Q_0, \mathfrak{F})$ with $\mathfrak{F} = \{F_1, \ldots, F_k\}$.

  // if $\mathfrak{F} = \emptyset$, then take $\mathfrak{F} = \{Q\}$

- Assume $\mathfrak{F} \neq \emptyset$

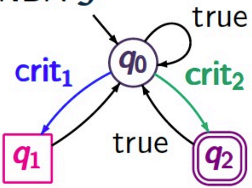
- Define NBA $\mathfrak{A}_{\mathfrak{G}} = (Q', \Sigma, \delta', Q'_0, F)$ with
  - $Q' = Q \times \{1, \ldots, k\}$ "k copies of Q"
  - $Q'_0 = Q_0 \times \{1\}$ "take init states from first copy"
  - $F = F_1 \times \{1\}$ "take acc. " " "
  - $\delta'(\langle q, i\rangle, A) = \begin{cases} \{\langle q', i\rangle \mid q' \in \delta(q, A)\} & \text{if } q \notin F_i \\ \{\langle q', \underline{i \oplus_k 1}\rangle \mid q' \in \delta(q, A)\} & \text{if } q \in F_i \end{cases}$

    $i \oplus_k 1 = \begin{cases} i+1 & \text{if } i < k \\ 1 & \text{if } i = k \end{cases}$

- Then: $\mathfrak{L}_\omega(\mathfrak{G}) = \mathfrak{L}_\omega(\mathfrak{A}_{\mathfrak{G}})$

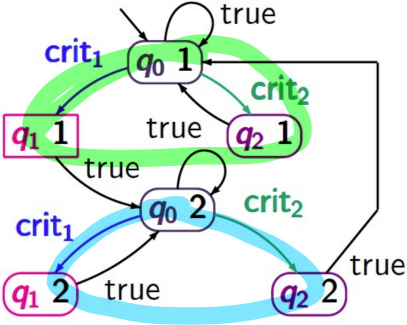
Example

GNBA $\mathcal{G}$



alphabet $\Sigma = 2^{AP}$ where
$AP = \{crit_1, crit_2\}$

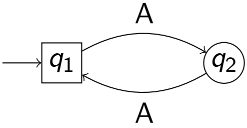
infinitely often $crit_1$ and
infinitely often $crit_2$



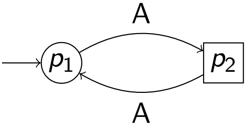
NBA $\mathcal{A}$

The Product Construction on NBA Fails

NBA $\mathfrak{A}_1$:

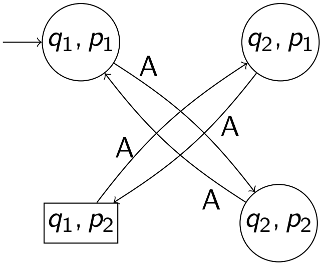


NBA $\mathfrak{A}_2$:



$L_\omega(\mathfrak{A}_1) = L_\omega(\mathfrak{A}_2) = \{A^\omega\} = L_\omega(\mathfrak{A}_1) \cap L_\omega(\mathfrak{A}_2)$

NBA $\mathfrak{A}_1 \otimes \mathfrak{A}_2$:



$L_\omega(\mathfrak{A}_1 \otimes \mathfrak{A}_2) = \emptyset$

The Product of NBA is a GNBA

For NBA $\mathfrak{A}_1$ and $\mathfrak{A}_1$, there is a GNBA $\mathfrak{G}$ such that $\mathfrak{L}_\omega(\mathfrak{G}) = \mathfrak{L}_\omega(\mathfrak{A}_1) \cap \mathfrak{L}_\omega(\mathfrak{A}_2)$.

Proof.

Let $\mathfrak{A}_i = (Q_i, \Sigma, \delta_i, Q_{0,i}, F_i)$ for $i = 1, 2$ be NBA. Define GNBA $\mathfrak{G} = (Q_1 \times Q_2, \Sigma, \delta, Q_{0,1} \times Q_{0,2}, \mathfrak{F})$ with:

- $\delta(\langle q_1, q_2 \rangle, A) = \{ \langle p_1, p_2 \rangle \mid p_1 \in \delta_1(q_1, A) \text{ and } p_2 \in \delta_2(q_2, A) \}$

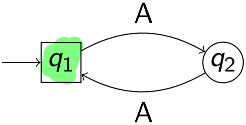
  as in product constr.

- $\mathfrak{F} = \{ F_1 \times Q_2, Q_1 \times F_2 \}$.

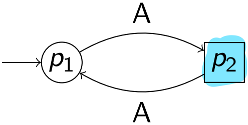
We can show that $\mathfrak{L}_\omega(\mathfrak{G}) = \mathfrak{L}_\omega(\mathfrak{A}_1) \cap \mathfrak{L}_\omega(\mathfrak{A}_2)$. □

Example: NBA Product as a GNBA

NBA $\mathfrak{A}_1$:

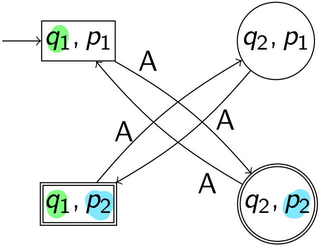


NBA $\mathfrak{A}_2$:



GNBA $\mathfrak{G}_{\mathfrak{A}_1 \otimes \mathfrak{A}_2}$:

$L_\omega(\mathcal{G}_{\mathfrak{A}_1 \otimes \mathfrak{A}_2}) = \{A^\omega\}$



Summary

The class of ω-regular languages coincides with

1. the class of languages described by ω-regular expressions
2. the class of languages recognised by nondeterministic Büchi automata
3. the class of languages recognised by generalised Büchi automata

But deterministic Büchi automata are strictly less expressive

The class of ω-regular languages is closed under $\cap$, $\cup$ and complement[1]

[1] Without further details.

	NFA	NBA
Expressiveness	regular	ω-regular
Closure under:		
$\cup$	union operator	union operator
$\cap$	product construction	via a GNBA product
complement	construct DFA + swap normal $\leftrightarrow$ final	complex procedure (eg. Safra construction)
Determinism	NFA $\equiv$ DFA (powerset construction)	DBA $\subsetneq$ NBA
$L(A) = \emptyset$?	DFS / BFS	nested DFS
Minimal automaton	unique minimal DFA	—

Next Lecture

Monday May 2, 10:30