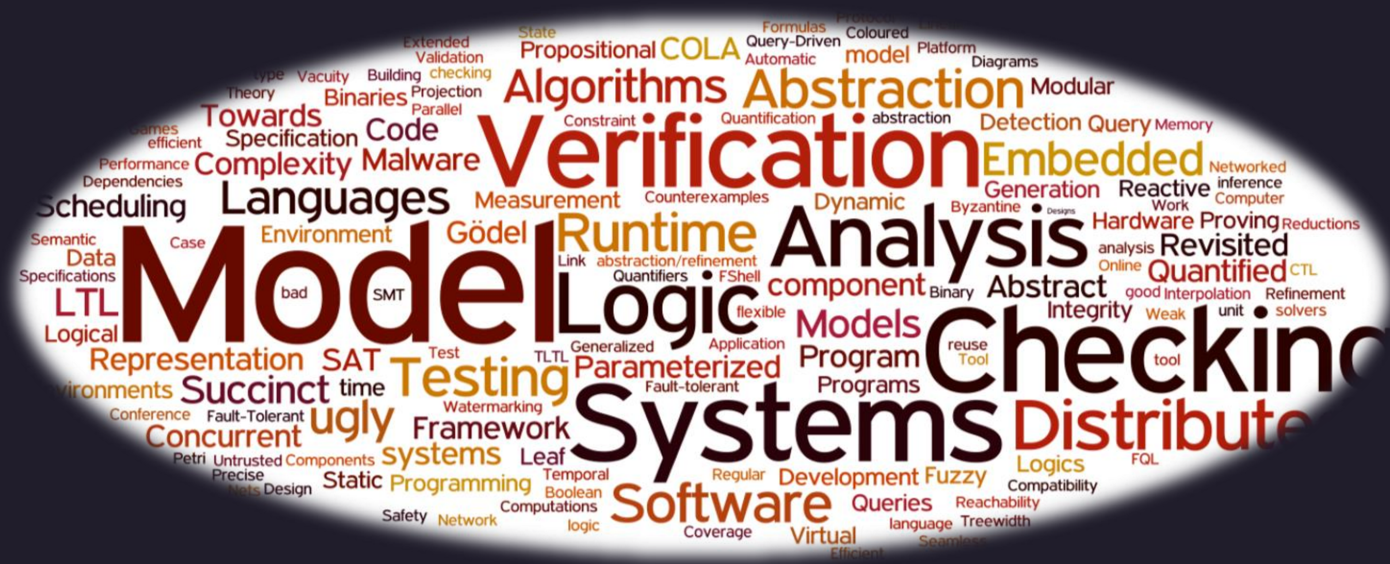
Omega Regular Languages and Büchi Automata

CS60030 FORMAL SYSTEMS

PALLAB DASGUPTA,
FNAE, FASc,
A K Singh Distinguished Professor in AI,
Dept of Computer Science & Engineering
Indian Institute of Technology Kharagpur
Email: pallab@cse.iitkgp.ac.in
Web: http://cse.iitkgp.ac.in/~pallab



INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

FMSAFE

FORMAL METHODS FOR SAFETY CRITICAL SYSTEMS

Regular expressions

- Let Σ be an alphabet with $\mathbf{A} \in \Sigma$

- Regular expressions over Σ have ***syntax:***

$$E ::= \underline{\phi} \mid \underline{\varepsilon} \mid \underline{A} \mid E + E' \mid E \,.\, E' \mid E^*$$

- The **semantics** of regular expression E is a language $L(E) \subseteq \Sigma^*$:

$$L(\underline{\phi}) = \phi^* \qquad L(\underline{\varepsilon}) = \{\varepsilon\} \qquad L(\underline{A}) = \{A\}$$

$$L(E + E') = L(E) \cup L(E') \qquad L(E.E') = L(E).L(E') \qquad L(E^*) = L(E)^*$$

Syntax of ω-regular expressions

- *Regular expressions* denote languages of finite words
- *ω-Regular expressions* denote languages of infinite words
- An *ω-regular expression* G over Σ has the form:

$$\boxed{G = E_1 . F_1^{\omega} + \ldots + E_n . F_n^{\omega} \ \text{ for } n>0}$$

  - where E_i, Fi are regular expressions over Σ with $\varepsilon \notin L(F_i)$

- Some examples:
  - $(A + B)^* . B^{\omega}$
  - $(B^* . A)^{\omega}$
  - $A^* . B^{\omega} + A^{\omega}$

Semantics of ω-regular expressions

- *For* $L \subseteq \Sigma^*$ *let* $L^\omega = \{w_1 w_2 w_3 \ldots \mid \forall i \geq 0.\, wi \in L\}$

- *Let ω-regular expression* $G = E_1.F_1^\omega + \ldots + E_n.F_n^\omega$

- The ***semantics*** of G is the language $L_\omega(G) \subseteq \Sigma^\omega$:

$$\boxed{\boldsymbol{L_\omega(G) = L(E_1).L(F_1)^\omega \cup \ldots \cup L(E_n).L(F_n)^\omega}}$$

- G_1 and G_2 are ***equivalent***, denoted $G_1 \equiv G_2$, if $L_\omega(G_1) = L_\omega(G_2)$

ω-Regular languages

- L is ω-regular if $L = L_\omega(G)$ for some ω-regular expression G

- Examples over $\Sigma = \{A, B\}$:
  - Language of all words with infinitely many As: $(B^*.A)^\omega$
  - Language of all words with finitely many As: $(A + B)^*.B^\omega$
  - The empty language: $\emptyset^\omega$

- ω-Regular languages are closed under $\cup$, $\cap$ and complementation

ω-Regular safety properties

- **Definition:**
  - **LT property P over AP is ω-Regular if P is an ω-regular language over the alphabet 2^{AP}**
- **Or, equivalently:**
  - **LT property P over AP is ω-Regular if P is a language accepted by a nondeterministic Büchi automaton over 2^{AP}**

Nondeterministic Büchi automata

- NFA (and DFA) are incapable of accepting infinite words
- Automata on infinite words
  - Suited for accepting ω-regular languages
  - We consider nondeterministic Büchi automata (NBA)
- Accepting runs have to “check” the entire input word ⇒ are infinite
  - acceptance criteria for infinite runs are needed
- NBA are like NFA, but have a distinct *acceptance criterion*
  - *one of the accept states must be visited infinitely often*

Büchi Automata

A nondeterministic Büchi automaton (NBA) $\mathcal{A}$ is a tuple $(Q, \Sigma, \delta, Q_0, F)$ where:

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

Language of an NBA

- NBA $\mathcal{A}$ = (Q, Σ, δ, Q_0, F) and word σ = $A_0 A_1 A_2 \ldots \in \Sigma^\omega$

- A *run* for σ in $\mathcal{A}$ is an infinite sequence $q_0\, q_1\, q_2 \ldots$ such that:
  - $q_0 \in Q_0$ and $q_i \xrightarrow{A_{i+1}} q_{i+1}$ for all $i \geq 0$

- Run $q_0\, q_1\, q_2 \ldots$ is accepting if $q_i \in F$ for infinitely many i.

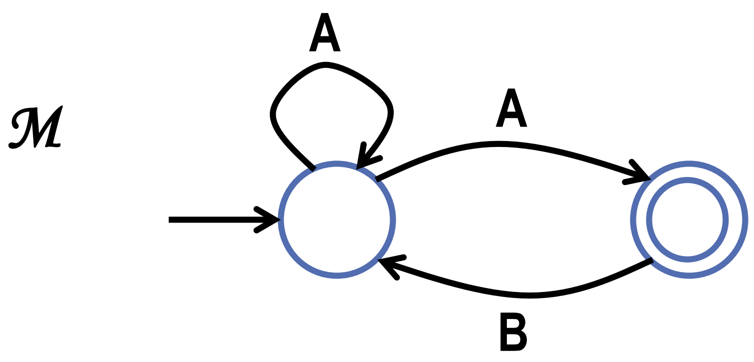
- $\sigma \in \Sigma^\omega$ is accepted by A if there exists an accepting run for σ

- The accepted language of A :

$$L_\omega(A) = \{\sigma \in \Sigma^\omega \mid there\ exists\ an\ accepting\ run\ for\ \sigma\ in\ A\}$$

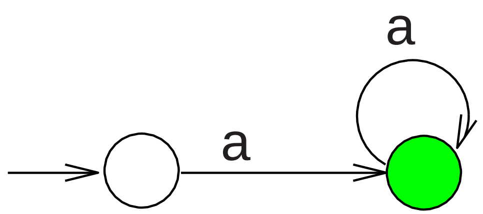
- NBA A and A' are equivalent if $L_\omega(A) = L_\omega(A')$

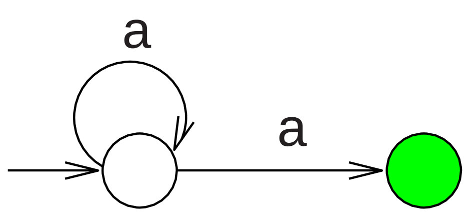
An Example NBA

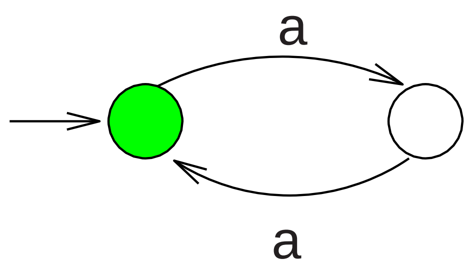


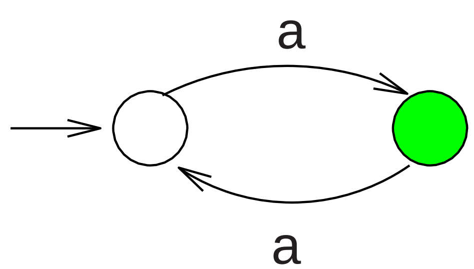
- If we treat $\mathcal{M}$ as a NFA, then $\mathcal{L}(\mathcal{M})$ = (A + AB)*A
- If we treat $\mathcal{M}$ as a NBA, then $\mathcal{L}(\mathcal{M})$ = $(A^*AB)^\omega$
  - Can you write some words which are accepted and some words which are not accepted?

NBA versus NFA









finite equivalence $\not\Rightarrow$ ω-equivalence

$L(A) = L(A')$, but $L_\omega(A) \neq L_\omega(A')$

ω-equivalence $\not\Rightarrow$ finite equivalence

$L_\omega(A) = L_\omega(A')$,but $L(A) \neq L(A')$

NBA and ω-regular languages

The class of languages accepted by NBA agrees with the class of ω-regular languages.

This means:

(1) any ω-regular language is recognized by an NBA

(2) for any NBA A, the language $L_\omega(A)$ is ω-regular

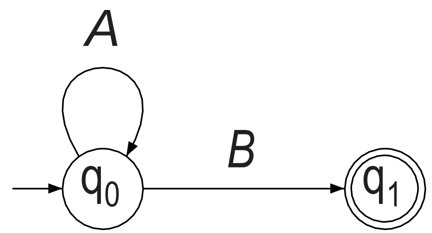
For any ω-regular language there is an NBA

- How to construct an NBA for the ω-regular expression: $G = E_1.F_1^{\omega} + \ldots + E_n.F_n^{\omega}$?
  - where $\boldsymbol{E_i}$, $\boldsymbol{Fi}$ are regular expressions over Σ with $\boldsymbol{\varepsilon \notin L(F_i)}$
- Use operators on NBA, mimicking operators on ω-regular expressions:
  - for NBA A_1 and A_2 there is an NBA accepting $L_{\omega}(A_1) \cup L_{\omega}(A_2)$
  - for any regular language L with $\varepsilon \notin L$ there is an NBA accepting L^{ω}
  - for regular language L and NBA A' there is an NBA accepting $L.L_{\omega}(A')$
- We will discuss these three operators in detail

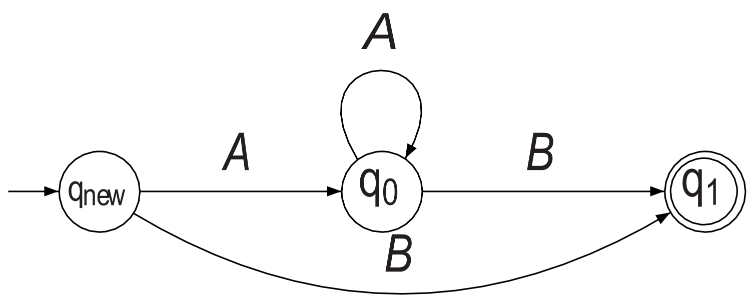
Union of NBAs

For NBA A_1 and A_2 (both over the alphabet Σ)
there exists an NBA A such that:
$L_\omega(A) = L_\omega(A_1) \cup L_\omega(A_2)$ **and** $|\mathrm{A}| = \mathrm{O}(|A_1| + |A_2|)$

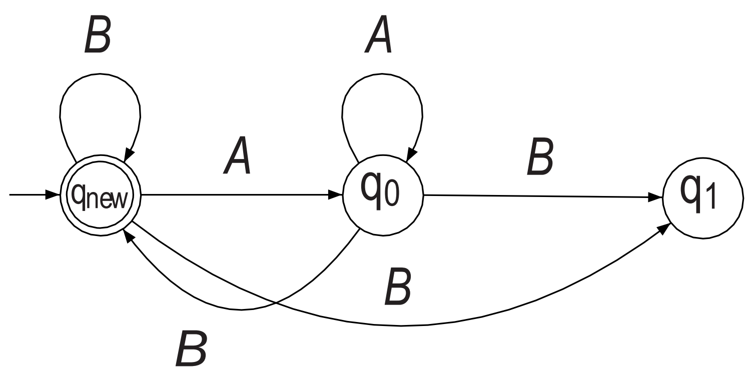
Definition of ω-operator for NFA



- Let $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ be an NFA with $\varepsilon \notin L(\mathcal{A})$.
- Assume no initial state in $\mathcal{A}$ has incoming transitions and $Q_0 \cap F = \emptyset$
  - Otherwise introduce a new initial state $q_{new} \notin F$
  - Let $q_{new} \xrightarrow{A} q$ iff $q_0 \xrightarrow{A} q$ for some $q_0 \in Q_0$
  - Keep all transitions in $\mathcal{A}$



- Construct an NBA $\mathcal{A}' = (Q, \Sigma, \delta', Q'_0, F')$ as follows
  - If $q \xrightarrow{A} q' \in F$ then add $q \xrightarrow{A} q_0$ for any $q_0 \in Q_0$
  - Keep all transitions in $\mathcal{A}$
  - $Q'_0 = Q_0$ and $\mathrm{F}' = Q_0$



From A*B to $(A^*B)^\omega$

Proof of $L_\omega(\mathcal{A}') \subseteq L(\mathcal{A})^\omega$

- Let $\sigma \in L_\omega(\mathcal{A}')$ and $q_0\, q_1\, q_2 \ldots$ be an accepting run for σ in $\mathcal{A}'$
  - Hence, $q_i \in F' = Q_0$ for infinitely many indices i
  - Let $i_0 = 0 < i_1 < i_2 < \ldots$ such that $q_{i_k} \in F'$ and $q_j \notin F'$ for $j \neq i_k$
- Divide σ into infinitely many nonempty finite sub-words $w_i \in \Sigma^*$:

  $\sigma = w_1 w_2 w_3 \ldots$ such that $q_{i_k} \in \delta'^*(q_{i_{k-1}}, w_k)$ for all $k > 0$
- It follows $\delta^*(q_{i_{k-1}}, w_k) \cap F \neq \phi$
  - $q_{i_k} \in Q_0$ and $q_{i_k} \in Q_0$ has no incoming transitions, thus $q_{i_k} \in F$
- Thus: $w_k \in L(\mathcal{A})$ for any $k > 0$, and hence $\sigma \in L(\mathcal{A})^\omega$

Proof of $L_\omega(\mathcal{A}') \supseteq L(\mathcal{A})^\omega$

- Let $\sigma = \boldsymbol{w_1 w_2 w_3} \ldots$ such that $\boldsymbol{w_k} \in \boldsymbol{L}(\mathcal{A})$ for all $k > 0$
  - That is, $\sigma \in \boldsymbol{L}(\mathcal{A})^{\omega}$
- Let $\boldsymbol{q_0^k\, q_1^k\, q_2^k \ldots q_{n_k}^k}$ be an accepting run for $\boldsymbol{w_k}$ in $\mathcal{A}$
- By definition of $\mathcal{A}'$, we have $\boldsymbol{q_0^{k+1} \in \delta'^*(q_0^k, w_k)}$ for all $\boldsymbol{k > 0}$

  $\boldsymbol{q_0^1 \ldots q_{n_1-1}^1\ \ q_0^2 \ldots q_{n_2-1}^2\ \ q_0^3 \ldots q_{n_3-1}^3} \ldots$ is an accepting run for σ in $\mathcal{A}'$
- Hence $\sigma \in \boldsymbol{L_\omega}(\mathcal{A})$

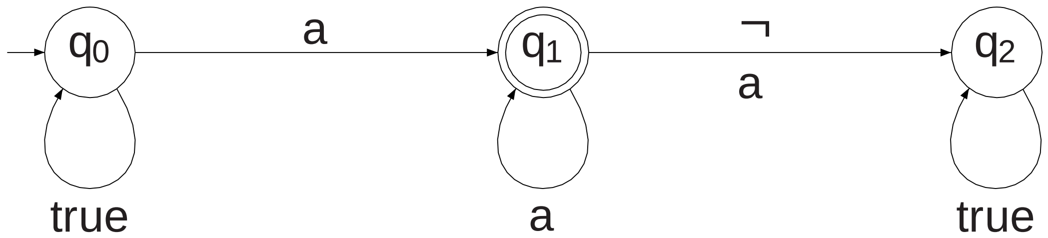
Concatenating an NFA and an NBA

For NFA A and NBA A' (both over the alphabet Σ)
there exists an NBA A'' with:
$L_\omega(A'') = L(A).L_\omega(A')$ and $|A''| = \mathbf{O}(|A| + |A'|)$

Interesting questions for NBA

- How to determine whether a NBA is empty?
- What is a deterministic BA?

- NBAs are more powerful than DBAs. The language $(A+B)^*B^\omega$ is accepted by a NBA but not accepted by any DBA (why?)
- Non-determinism is useful:



Let AP = { a }, i.e., 2^{AP} = $\{A, B\}$ where A = { } and B = {a}

The language: *eventually forever a* may be represented as $(A + B)^* B^\omega$ = $(\{\} + \{a\})^* \{a\}^\omega$

Generalized Büchi automata

- NBA are as expressive as ω-regular languages
- Variants of NBA exist that are equally expressive
  - Muller, Rabin, and Streett automata
  - *generalized Büchi automata* (GNBA)
- GNBA are like NBA, but have a distinct *acceptance criterion*
  - a GNBA requires to visit several sets $F_1, \ldots, F_k$ $(k \geq 0)$ infinitely often
  - for $k = 0$, all runs are accepting
  - for $k = 1$ this boils down to an NBA
- GNBA are useful to relate temporal logic and automata
  - but they are equally expressive as NBA

Generalized Büchi automata

A generalized NBA (GNBA) G is a tuple $(Q, \Sigma, \delta, Q_0, F)$ where:

- Q is a finite set of states with $Q_0 \subseteq Q$ a set of initial states
- Σ is an alphabet
- $\delta : Q \times \Sigma \rightarrow 2^Q$ is a transition function
- $F = \{ F_1, \ldots, F_k \}$ is a (possibly empty) subset of 2^{2^Q}

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

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

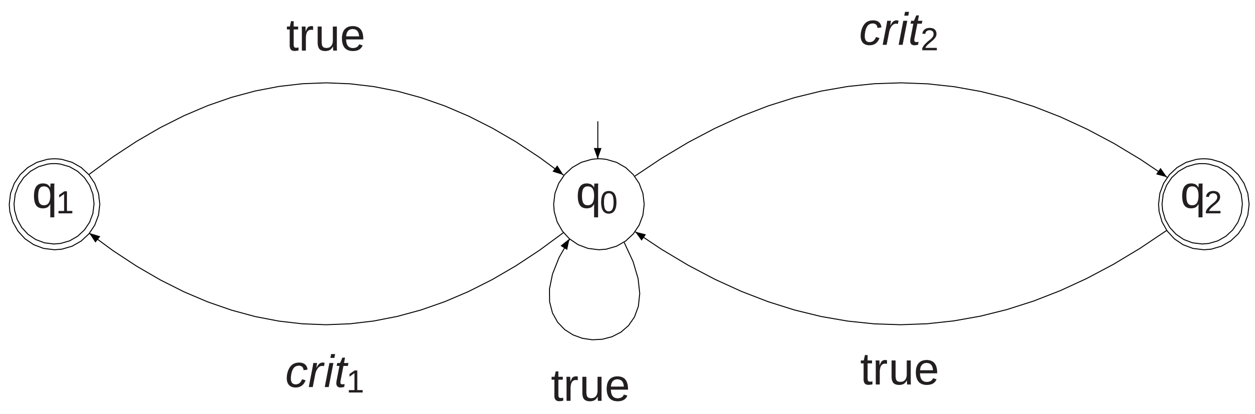
Language of a GNBA

- GNBA $\mathcal{G} = (Q, \Sigma, \delta, Q_0, \mathcal{F})$ and word $\sigma = A_0 A_1 A_2 \ldots \in \Sigma^\omega$
- A *run* for σ in $\mathcal{G}$ is an infinite sequence $q_0\, q_1\, q_2 \ldots$ such that:
  - $q_0 \in Q_0$ and $q_i \xrightarrow{A_i} q_{i+1}$ for all $i \geq 0$
- Run $q_0\, q_1\, q_2 \ldots$ is accepting if for all $F \in \mathcal{F}$ for infinitely many i.
- $\sigma \in \Sigma^\omega$ is accepted by $\mathcal{G}$ if there exists an accepting run for σ
- The accepted language of $\mathcal{G}$:

$$L_\omega(\mathcal{G}) = \{\sigma \in \Sigma^\omega \mid there\ exists\ an\ accepting\ run\ for\ \sigma\ in\ \mathcal{G}\}$$

- GNBA $\mathcal{G}$ and $\mathcal{G}'$ are equivalent if $L_\omega(\mathcal{G}) = L_\omega(\mathcal{G}')$

Example



A GNBA for the property *"both processes are infinitely often in their critical section"*

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

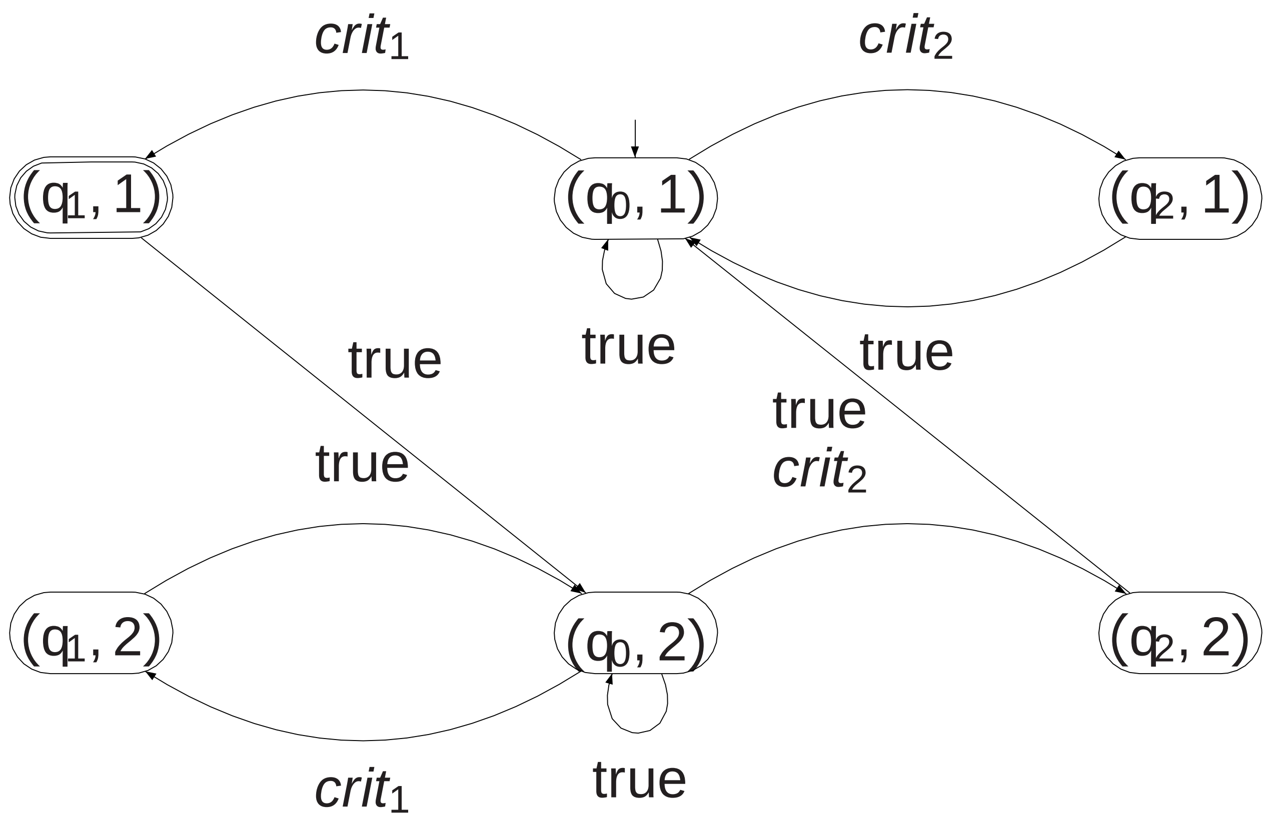
From GNBA to NBA

For any GNBA G there exists an NBA A with:

$L_\omega(G) = L_\omega(A)$ and $|A| = O(|G| \cdot |F|)$

where F denotes the set of acceptance sets in G

Example



$crit_1$
$crit_2$
$(q_1, 1)$
$(q_0, 1)$
$(q_2, 1)$
true
true
true
true
true
$crit_2$
true
$(q_1, 2)$
$(q_0, 2)$
$(q_2, 2)$
$crit_1$
true

Facts about Büchi automata

- They are as expressive as ω-regular languages
- They are closed under various operations and also under ∩
- Nondeterministic BA are more expressive than deterministic BA
- Emptiness check = check for reachable recurrent accept state
  - this can be done in $O(|A|)$